



Panduan Pengguna untuk Versi 2

# AWS Command Line Interface



# AWS Command Line Interface: Panduan Pengguna untuk Versi 2

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan properti dari masing-masing pemilik, yang mungkin berafiliasi, terkait dengan, atau disponsori oleh Amazon, atau tidak.

# Table of Contents

.....	ix
Tentang AWS CLI .....	1
Tentang AWS CLI versi 2 .....	1
Pemeliharaan dan dukungan untuk versi utama SDK .....	2
Tentang Amazon Web Services .....	2
Tentang contoh .....	2
Dokumentasi dan sumber daya tambahan .....	4
AWS CLIdokumentasi dan sumber daya .....	4
AWSSDK lainnya .....	4
Memulai .....	6
Prasyarat .....	7
Buat akun administratif IAM atau IAM Identity Center .....	7
Langkah selanjutnya .....	8
Instal/Perbarui .....	9
AWS CLI menginstal dan memperbarui instruksi .....	9
Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi .....	23
Langkah selanjutnya .....	23
Rilis sebelumnya .....	23
Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi .....	41
Langkah selanjutnya .....	41
Membangun dan menginstal dari sumber .....	41
Mengapa membangun dari sumber? .....	42
Langkah cepat .....	43
Langkah 1: Siapkan semua persyaratan .....	45
Langkah 2: Mengkonfigurasi instalasi AWS CLI sumber .....	50
Langkah 3: Membangun AWS CLI .....	56
Langkah 4: Memasang AWS CLI .....	57
Langkah 5: Memverifikasi AWS CLI instalasi .....	59
Contoh alur kerja .....	59
Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi .....	62
Langkah selanjutnya .....	62
Amazon ECR Publik/Docker .....	62
Prasyarat .....	63
Memutuskan antara Amazon ECR Public dan Docker Hub .....	63

Jalankan gambar resmi .....	64
Catatan tentang antarmuka dan kompatibilitas mundur dari gambar resmi .....	65
Gunakan versi dan tag tertentu .....	65
Perbarui ke gambar resmi terbaru .....	66
Bagikan file host, kredensyal, variabel lingkungan, dan konfigurasi .....	67
Mempersingkat perintah docker run .....	73
Pengaturan .....	76
Kumpulkan informasi kredensi Anda untuk akses terprogram .....	76
Menyiapkan konfigurasi dan kredensial baru .....	78
Menggunakan file konfigurasi dan kredensial yang ada .....	86
Konfigurasi AWS CLI .....	87
Konfigurasi dan kredensialnya diutamakan .....	87
Topik tambahan di bagian ini .....	88
Konfigurasi dan pengaturan file kredensi .....	89
Format konfigurasi dan file kredensyal .....	89
Di mana pengaturan konfigurasi disimpan? .....	98
Menggunakan profil bernama .....	99
Mengatur dan melihat pengaturan konfigurasi menggunakan perintah .....	99
Menyetel konfigurasi baru dan contoh perintah kredensyal .....	102
Pengaturan config file yang didukung .....	105
Variabel lingkungan .....	124
Cara mengatur variabel lingkungan .....	124
AWS CLI variabel lingkungan yang didukung .....	126
Opsi baris perintah .....	136
Cara menggunakan opsi baris perintah .....	136
AWS CLI mendukung opsi baris perintah global .....	137
Penggunaan umum opsi baris perintah .....	142
Penyelesaian perintah .....	142
Cara kerjanya .....	143
Mengkonfigurasi penyelesaian perintah di Linux atau macOS .....	144
Mengkonfigurasi penyelesaian perintah pada Windows .....	147
Percobaan ulang .....	148
Mode coba lagi yang tersedia .....	149
Mengkonfigurasi mode coba lagi .....	151
Melihat log upaya coba lagi .....	153
Gunakan proxy HTTP .....	154

Menggunakan contoh .....	154
Mengautentikasi ke proxy .....	155
Menggunakan proxy di instans Amazon EC2 .....	156
Pemecahan Masalah .....	156
Titik akhir .....	156
Tetapkan titik akhir untuk satu perintah .....	157
Tetapkan titik akhir global untuk semua Layanan AWS .....	157
Setel untuk menggunakan titik akhir FIP untuk semua Layanan AWS .....	159
Setel untuk menggunakan titik akhir dual-stack untuk semua Layanan AWS .....	160
Tetapkan titik akhir khusus layanan .....	161
Konfigurasi titik akhir dan pengaturan diutamakan .....	164
Otentikasi dan akses kredensial .....	166
Konfigurasi dan prioritas kredensial .....	167
Topik tambahan di bagian ini .....	168
Autentikasi Pusat Identitas IAM .....	168
Konfigurasi penyegaran token otomatis .....	169
Konfigurasi lawas yang tidak dapat disegarkan .....	177
Menggunakan profil Pusat Identitas IAM .....	182
Kredensial jangka pendek .....	185
Peran IAM .....	187
Prasyarat .....	187
Ikhtisar penggunaan peran IAM .....	187
Mengkonfigurasi dan menggunakan peran .....	189
Menggunakan MFA .....	191
Peran lintas akun dan ID eksternal .....	192
Menentukan nama sesi peran untuk audit yang lebih mudah .....	193
Asumsikan peran dengan identitas web .....	194
Membersihkan kredensial yang di-cache .....	195
Pengguna IAM .....	196
Langkah 1: Buat pengguna IAM Anda .....	196
Langkah 2: Dapatkan kunci akses Anda .....	197
Konfigurasi AWS CLI .....	197
Menggunakan kredensial untuk metadata instans Amazon EC2 .....	199
Prasyarat .....	200
Mengonfigurasi profil untuk metadata Amazon EC2 .....	200
Kredensial eksternal .....	201

---

Gunakan AWS CLI .....	204
Dapatkan Bantuan .....	204
Perintah AWS CLI bantuan bawaan .....	205
AWS CLI panduan referensi .....	210
Dokumentasi API .....	210
Memecahkan masalah kesalahan .....	211
Bantuan tambahan .....	211
Struktur Perintah .....	211
Struktur perintah .....	211
Tunggu perintah .....	212
Tentukan Nilai Parameter .....	214
Jenis Parameter Umum .....	215
Kutipan dengan Strings .....	220
Parameter dari File .....	224
Hasilkan Template Kerangka CLI .....	227
Sintaks Shorthand .....	239
Prompt otomatis .....	242
Cara kerjanya .....	242
Fitur prompt otomatis .....	242
Mode prompt otomatis .....	246
Konfigurasi prompt otomatis .....	246
Output Perintah Kontrol .....	246
Output sensitif .....	247
Opsi keluaran sisi server vs sisi klien .....	248
Format Output .....	248
Paginasi .....	258
Filter keluaran .....	264
Kode Pengembalian .....	288
Penyihir .....	290
Cara kerjanya .....	290
Alias .....	291
Prasyarat .....	291
Langkah 1: Membuat file alias .....	292
Langkah 2: Membuat alias .....	293
Langkah 3: Memanggil alias .....	296
Contoh repositori alias .....	298

Sumber daya .....	299
Contoh kode .....	301
Contoh perintah terpandu .....	301
DynamoDB .....	302
Amazon EC2 .....	306
S3 Glacier .....	325
IAM .....	332
Amazon S3 .....	337
Amazon SNS .....	356
Contoh perintah .....	358
Tindakan dan skenario .....	359
Contoh skrip Bash .....	6432
Tindakan dan skenario .....	6432
Keamanan .....	6596
Perlindungan Data .....	6597
Enkripsi data .....	6598
Identity and Access Management .....	6598
Audiens .....	6599
Mengautentikasi dengan identitas .....	6599
Mengelola akses menggunakan kebijakan .....	6603
Bagaimana Layanan AWS bekerja dengan IAM .....	6606
Memecahkan masalah AWS identitas dan akses .....	6606
Validasi Kepatuhan .....	6608
Ketangguhan .....	6609
Keamanan Infrastruktur .....	6610
Menegakkan versi TLS minimum .....	6611
Memecahkan masalah kesalahan .....	6612
Pemecahan masalah umum untuk dicoba terlebih dahulu .....	6612
Periksa pemformatan AWS CLI perintah Anda .....	6613
Periksa AWS CLI perintah yang Wilayah AWS Anda gunakan .....	6613
Konfirmasikan bahwa Anda menjalankan versi terbaru AWS CLI .....	6614
Gunakan --debug opsi .....	6614
Aktifkan dan tinjau log riwayat AWS CLI perintah .....	6620
Konfirmasikan bahwa Anda telah AWS CLI dikonfigurasi .....	6621
Perintah tidak ditemukan kesalahan .....	6621
Perintah <code>aws --version ""</code> mengembalikan versi yang berbeda dari yang Anda instal .....	6624

Perintah <code>aws --version</code> mengembalikn versi setelah mencopot pemasangan AWS CLI	6625
AWS CLI Diproses perintah dengan nama parameter yang tidak lengkap	6626
Akses kesalahan ditolak	6628
Kredensyal dan kesalahan kunci tidak valid	6629
Tanda tangan tidak cocok dengan kesalahan	6630
Kesalahan sertifikat SSL	6632
Kesalahan JSON tidak valid	6633
Sumber daya tambahan	6635
Panduan migrasi	6636
Fitur dan perubahan baru	6636
AWS CLI versi 2 fitur baru	6636
Memecahkan perubahan antara AWS CLI versi 1 dan AWS CLI versi 2	6638
Instruksi migrasi	6645
Mengganti versi 1 dengan versi 2	6646
Side-by-side instal	6647
Menghapus instalasi	6648
Memecahkan masalah kesalahan AWS CLI pemasangan dan uninstall	6651
Riwayat Dokumen	6652
AWS Glosarium	6658



Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.

# Apa AWS Command Line Interface?

The AWS Command Line Interface (AWS CLI) adalah alat open source yang memungkinkan Anda berinteraksi dengan AWS layanan menggunakan perintah di shell baris perintah Anda. Dengan konfigurasi minimal, AWS CLI memungkinkan Anda untuk mulai menjalankan perintah yang mengimplementasikan fungsionalitas yang setara dengan yang disediakan oleh browser berbasis AWS Management Console dari prompt perintah di program terminal Anda:

- Shell Linux — Gunakan program shell umum seperti [bash](#), [zsh](#), dan [tcsh](#) untuk menjalankan perintah di Linux atau macOS.
- Baris perintah Windows — Pada Windows, jalankan perintah di prompt perintah Windows atau di PowerShell.
- Jarak Jauh — Jalankan perintah di Amazon Elastic Compute Cloud (Amazon EC2) instans melalui program terminal jarak jauh seperti PuTTY atau SSH, atau dengan. AWS Systems Manager

Semua fungsi AWS administrasi, manajemen, dan akses IaaS (infrastruktur sebagai layanan) AWS Management Console tersedia di AWS API dan. AWS CLI Fitur dan layanan AWS IaaS baru menyediakan AWS Management Console fungsionalitas penuh melalui API dan CLI saat peluncuran atau dalam 180 hari peluncuran.

AWS CLI Menyediakan akses langsung ke API AWS layanan publik. Anda dapat menjelajahi kemampuan layanan dengan AWS CLI, dan mengembangkan skrip shell untuk mengelola sumber daya Anda. Selain perintah tingkat rendah, setara API, beberapa AWS layanan menyediakan penyesuaian untuk. AWS CLI Kustomisasi dapat mencakup perintah tingkat tinggi yang menyederhanakan penggunaan layanan dengan API yang kompleks.

## Tentang AWS CLI versi 2

AWS CLI Versi 2 adalah versi utama terbaru AWS CLI dan mendukung semua fitur terbaru. Beberapa fitur yang diperkenalkan di versi 2 tidak di-backport ke versi 1 dan Anda harus meningkatkan untuk mengakses fitur-fitur tersebut. Ada beberapa perubahan “melanggar” dari versi 1 yang mungkin mengharuskan Anda untuk mengubah skrip Anda. Untuk daftar perubahan yang melanggar di versi 2, lihat [Migrasi dari AWS CLI versi 1 ke versi 2](#).

AWS CLI Versi 2 tersedia untuk diinstal hanya sebagai installer yang dibundel. Meskipun Anda mungkin menemukannya di manajer paket, ini adalah paket tidak didukung dan tidak resmi yang

tidak diproduksi atau dikelola oleh. AWS Kami menyarankan Anda menginstal AWS CLI dari hanya titik AWS distribusi resmi, seperti yang didokumentasikan dalam panduan ini.

Untuk menginstal AWS CLI versi 2, lihat [the section called “Instal/Perbarui”](#).

Untuk memeriksa versi yang saat ini diinstal, gunakan perintah berikut:

```
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Untuk riwayat versi, lihat [Changelog AWS CLI versi 2](#) aktif. GitHub

## Pemeliharaan dan dukungan untuk versi utama SDK

Untuk informasi tentang pemeliharaan dan dukungan untuk versi utama SDK dan dependensi yang mendasarinya, lihat berikut di [Panduan Referensi SDK dan Alat AWS](#):

- [AWS Kebijakan pemeliharaan SDK dan alat](#)
- [AWS Matriks dukungan versi SDK dan alat](#)

## Tentang Amazon Web Services

Amazon Web Services (AWS) adalah kumpulan layanan infrastruktur digital yang dapat dimanfaatkan developer saat mengembangkan aplikasi mereka. Layanan termasuk komputasi, penyimpanan, database, dan sinkronisasi aplikasi (pesan dan antrian). AWS menggunakan model pay-as-you-go layanan. Anda hanya dikenakan biaya untuk layanan yang Anda—atau aplikasi Anda—gunakan. Selain itu, untuk membuat AWS lebih mudah didekati sebagai platform untuk pembuatan prototipe dan eksperimen, AWS menawarkan tingkat penggunaan gratis. Pada tingkat ini, layanan akan gratis di bawah tingkat penggunaan tertentu. Untuk informasi selengkapnya tentang AWS biaya dan Tingkat Gratis, lihat [Tingkat AWS Gratis](#). Untuk mendapatkan AWS akun, buka [AWS halaman](#) beranda dan kemudian pilih Buat AWS Akun.

## Tentang AWS CLI contoh

Contoh AWS Command Line Interface (AWS CLI) dalam panduan ini diformat menggunakan konvensi berikut:

- Prompt — Command prompt menggunakan prompt Linux dan ditampilkan sebagai (\$ ). Untuk perintah yang khusus Windows, C:\> digunakan sebagai prompt. Jangan sertakan prompt saat Anda mengetik perintah.
- Direktori — Ketika perintah harus dijalankan dari direktori tertentu, nama direktori ditampilkan sebelum simbol prompt.
- Input pengguna - Teks perintah yang Anda masukkan pada baris perintah diformat sebagai **user input**.
- Teks yang dapat diganti - Teks variabel, termasuk nama sumber daya yang Anda pilih, atau ID yang dihasilkan oleh AWS layanan yang harus Anda sertakan dalam perintah, diformat sebagai teks yang *dapat diganti*. Dalam perintah atau perintah multi-baris di mana input keyboard tertentu diperlukan, perintah keyboard juga dapat ditampilkan sebagai teks yang dapat diganti.
- Output - Output yang dikembalikan oleh AWS layanan ditampilkan di bawah input pengguna, dan diformat sebagai **computer output**.

Contoh **aws configure** perintah berikut menunjukkan input pengguna, teks yang dapat diganti, dan output:

1. Masukkan **aws configure** di baris perintah, lalu tekan Enter.
2. AWS CLI Output baris teks, meminta Anda untuk memasukkan informasi tambahan.
3. Masukkan setiap tombol akses Anda secara bergantian, lalu tekan Enter.
4. Kemudian, masukkan nama AWS Wilayah dalam format yang ditampilkan, tekan Enter, lalu tekan Memasukkan waktu terakhir untuk melewati pengaturan format output.
5. Perintah Enter terakhir ditampilkan sebagai teks yang dapat diganti karena tidak ada input pengguna untuk baris itu.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: ENTER
```

Contoh berikut menunjukkan perintah sederhana dengan output. Untuk menggunakan contoh ini, masukkan teks lengkap perintah (teks yang disorot setelah prompt), lalu tekan Enter. Nama grup keamanan, *my-sg*, dapat diganti dengan nama grup keamanan yang Anda inginkan. Dokumen JSON, termasuk kurawal kurawal, adalah output. Jika Anda mengonfigurasi CLI Anda ke output

dalam format teks atau tabel, output akan diformat secara berbeda. [JSON](#) adalah format output default.

```
$ aws ec2 create-security-group --group-name my-sg --description "My security group"
{
  "GroupId": "sg-903004f8"
}
```

## Dokumentasi dan sumber daya tambahan

### AWS CLIdokumentasi dan sumber daya

Selain panduan pengguna ini, berikut ini adalah sumber daya online yang berharga untuk AWS CLI.

- [AWS CLI panduan referensi versi 2](#)
- [AWS CLI contoh kode repositori](#)
- [AWS CLI GitHub repositori](#) Anda dapat melihat dan melakukan fork kode sumber untuk on. AWS CLI GitHub Bergabunglah dengan komunitas pengguna GitHub untuk memberikan umpan balik, meminta fitur, dan mengirimkan kontribusi Anda sendiri.
- [AWS CLI alias contoh repositori](#) Anda dapat melihat dan memotong contoh AWS CLI alias pada GitHub
- [AWS CLI versi 2 Changelog](#)

### AWSSDK lainnya

Bergantung pada kasus penggunaan Anda, Anda mungkin ingin memilih salah satu AWS SDK atau: AWS Tools for PowerShell

- [AWS Tools for PowerShell](#)
- [AWS SDK for Java](#)
- [AWS SDK for .NET](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for Ruby](#)
- [AWS SDK for Python \(Boto\)](#)
- [AWS SDK for PHP](#)

- [AWS SDK for Go](#)
- [AWS Mobile SDK for iOS](#)
- [AWS Mobile SDK for Android](#)

# Memulai dengan AWS CLI

Bab ini memberikan langkah-langkah untuk memulai dengan versi 2 dari AWS Command Line Interface (AWS CLI) dan menyediakan tautan ke instruksi yang relevan.

1. [Lengkapi semua prasyarat](#) - Untuk mengakses AWS layanan dengan AWS CLI, Anda memerlukan minimal kredensial dan IAM. Akun AWS Untuk meningkatkan keamanan AWS akun Anda, kami sarankan Anda tidak menggunakan kredensial akun root Anda. Anda harus membuat pengguna dengan hak istimewa paling sedikit untuk memberikan kredensial akses ke tugas yang akan Anda jalankan. AWS
2. Instal atau dapatkan akses ke AWS CLI menggunakan salah satu metode berikut:
  - (Direkomendasikan) [the section called “Instal/Perbarui”](#).
  - [the section called “Rilis sebelumnya”](#). Menginstal versi tertentu terutama digunakan jika tim Anda menyelaraskan alat mereka ke versi tertentu.
  - [the section called “Membangun dan menginstal dari sumber”](#). Membangun AWS CLI from GitHub source adalah metode yang lebih mendalam yang terutama digunakan oleh pelanggan yang bekerja pada platform yang tidak kami dukung secara langsung dengan installer pra-bangun kami.
  - [the section called “Amazon ECR Publik/Docker”](#).
  - Akses AWS CLI versi 2 di AWS konsol dari browser Anda menggunakan AWS CloudShell. Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS CloudShell](#).
3. [Setelah Anda memiliki akses ke AWS CLI, konfigurasi AWS CLI dengan kredensial IAM Anda untuk penggunaan pertama kali](#).

## Mengatasi masalah penginstal atau mengkonfigurasi kesalahan

Jika Anda mengalami masalah setelah menginstal, mencopot pemasangan, atau mengonfigurasi AWS CLI, lihat [Memecahkan masalah kesalahan](#) langkah-langkah pemecahan masalah.

## Topik

- [Prasyarat untuk menggunakan versi 2 AWS CLI](#)
- [Instal atau perbarui ke versi terbaru AWS CLI](#)

- [Instal rilis sebelumnya dari AWS CLI versi 2](#)
- [Membangun dan menginstal AWS CLI dari sumber](#)
- [Jalankan AWS CLI dari gambar Amazon ECR Public atau Docker resmi](#)
- [Mengatur AWS CLI](#)

## Prasyarat untuk menggunakan versi 2 AWS CLI

Untuk mengakses AWS layanan dengan AWS CLI, Anda memerlukan kredensi Akun AWS dan IAM. Saat menjalankan AWS CLI perintah, AWS CLI kebutuhan untuk memiliki akses ke AWS kredensial tersebut. Untuk meningkatkan keamanan AWS akun Anda, kami sarankan Anda tidak menggunakan kredensi akun root Anda. Anda harus membuat pengguna dengan hak istimewa paling sedikit untuk memberikan kredensial akses ke tugas yang akan Anda jalankan. AWS

Topik

- [Buat akun administratif IAM atau IAM Identity Center](#)
- [Langkah selanjutnya](#)

## Buat akun administratif IAM atau IAM Identity Center

Sebelum Anda dapat mengkonfigurasi

Untuk membuat pengguna administrator, pilih salah satu opsi berikut.

Pilih salah satu cara untuk mengelola administrator Anda	Ke	oleh	Anda juga bisa
Di Pusat Identitas IAM	Gunakan kredensi jangka pendek untuk mengakses. AWS	Mengikuti petunjuk di <a href="#">Memulai</a> di Panduan	Konfigurasi akses terprogram dengan <a href="#">Mengonfigurasi AWS CLI yang akan</a>



Pilih salah satu cara untuk mengelola administrator Anda	Ke	oleh	Anda juga bisa
(Direkomendasikan)	Ini sejalan dengan praktik terbaik keamanan. Untuk informasi tentang praktik terbaik, lihat <a href="#">Praktik terbaik keamanan di IAM</a> di Panduan Pengguna IAM.	AWS IAM Identity Center Pengguna.	<a href="#">digunakan AWS IAM Identity Center</a> dalam AWS Command Line Interface Panduan Pengguna.
Di IAM (Tidak disarankan)	Gunakan kredensi jangka panjang untuk mengakses. AWS	Mengikuti petunjuk dalam <a href="#">Membuat pengguna admin IAM pertama Anda dan grup pengguna</a> di Panduan Pengguna IAM.	Konfigurasi akses terprogram dengan <a href="#">Mengelola kunci akses untuk pengguna IAM di Panduan Pengguna IAM</a> .

## Langkah selanjutnya

Setelah membuat kredensi Akun AWS dan IAM, untuk menggunakan AWS CLI Anda dapat melakukan salah satu hal berikut:

- [Instal rilis terbaru](#) AWS CLI versi 2 di komputer Anda.
- [Instal rilis AWS CLI versi 2 sebelumnya](#) di komputer Anda.
- Akses AWS CLI versi 2 dari komputer Anda [menggunakan gambar Docker](#).
- Akses AWS CLI versi 2 di AWS konsol dari browser Anda menggunakan AWS CloudShell. Untuk informasi selengkapnya, lihat [AWS CloudShell Panduan Pengguna](#).

# Instal atau perbarui ke versi terbaru AWS CLI

Topik ini menjelaskan cara menginstal atau memperbarui rilis terbaru dari AWS Command Line Interface (AWS CLI) pada sistem operasi yang didukung. Untuk informasi tentang rilis terbaru AWS CLI, lihat [Changelog AWS CLI versi 2](#) di GitHub

Untuk menginstal rilis sebelumnya AWS CLI, lihat [the section called “Rilis sebelumnya”](#). Untuk petunjuk uninstall, lihat [Menghapus instalasi](#).

## Important

AWS CLI versi 1 dan 2 menggunakan nama aws perintah yang sama. Jika sebelumnya Anda menginstal AWS CLI versi 1, lihat [Migrasi dari AWS CLI versi 1 ke versi 2](#).

## Topik

- [AWS CLI menginstal dan memperbarui instruksi](#)
- [Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi](#)
- [Langkah selanjutnya](#)

## AWS CLI menginstal dan memperbarui instruksi

Untuk instruksi instalasi, perluas bagian untuk sistem operasi Anda.

### Linux

#### Instal dan perbarui persyaratan

- Anda harus dapat mengekstrak atau “unzip” paket yang diunduh. Gunakan ekuivalen jika sistem pengoperasian Anda tidak memiliki perintah unzip.
- AWS CLI Kegunaannya `glibc`, `groff`, dan `less`. Ini disertakan secara default di sebagian besar distribusi utama Linux.
- Kami mendukung versi 64-bit dari distribusi terbaru CentOS, Fedora, Ubuntu, Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, dan Linux ARM. AWS CLI
- Karena AWS tidak memelihara repositori pihak ketiga, kami tidak dapat menjamin bahwa repositori tersebut berisi versi terbaru dari AWS CLI

## Instal atau perbarui AWS CLI

### Warning

Jika ini adalah pertama kalinya Anda memperbarui di Amazon Linux, untuk menginstal versi terbaru dari AWS CLI, Anda harus menghapus yum versi pra-instal menggunakan perintah berikut:

```
$ sudo yum remove awscli
```

Setelah yum instalasi dihapus, ikuti petunjuk instalasi Linux di bawah ini. AWS CLI

Untuk memperbarui instalasi Anda saat ini AWS CLI, unduh penginstal baru setiap kali Anda memperbarui untuk menimpa versi sebelumnya. Ikuti langkah-langkah ini dari baris perintah untuk menginstal AWS CLI di Linux.

Berikut ini adalah langkah-langkah instalasi cepat dalam satu grup salin dan tempel berdasarkan apakah Anda menggunakan Linux 64-bit atau Linux ARM yang menyediakan instalasi dasar. Untuk petunjuk yang dipandu, lihat langkah-langkah berikut.

Linux x86 (64-bit)

### Note

(Opsional) Perintah berikut memblokir unduhan dan menginstal AWS CLI tanpa terlebih dahulu memverifikasi integritas unduhan Anda. Untuk memverifikasi integritas unduhan Anda, gunakan petunjuk langkah demi langkah di bawah ini.

Untuk menginstal AWS CLI, jalankan perintah berikut.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"  
unzip awscliv2.zip  
sudo ./aws/install
```

Untuk memperbarui instalasi Anda saat ini AWS CLI, tambahkan symlink yang ada dan informasi installer untuk membuat `install` perintah menggunakan `--bin-dir`, `--install-dir`, dan

parameter. `--update` *Blok perintah berikut menggunakan contoh symlink dari `/usr/local/bin` dan contoh lokasi installer `/usr/local/aws-cli`.*

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

## Linux ARM

### Note

(Opsional) Perintah berikut memblokir unduhan dan menginstal AWS CLI tanpa terlebih dahulu memverifikasi integritas unduhan Anda. Untuk memverifikasi integritas unduhan Anda, gunakan petunjuk langkah demi langkah di bawah ini.

Untuk menginstal AWS CLI, jalankan perintah berikut.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Untuk memperbarui instalasi Anda saat ini AWS CLI, tambahkan symlink yang ada dan informasi installer untuk membuat `install` perintah menggunakan `--bin-dir`, `--install-dir`, dan parameter. `--update` *Blok perintah berikut menggunakan contoh symlink dari `/usr/local/bin` dan contoh lokasi installer `/usr/local/aws-cli`.*

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

## Langkah-langkah instalasi terpandu

1. Unduh file instalasi dengan salah satu cara berikut:

## Linux x86 (64-bit)

- Gunakan **curl** perintah - -o Opsi menentukan nama file yang paket download ditulis. Opsi pada perintah contoh berikut menulis file yang diunduh ke direktori saat ini dengan nama lokalawscliv2.zip.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o  
"awscliv2.zip"
```

- Mengunduh dari URL — Untuk mengunduh penginstal dengan browser Anda, gunakan URL berikut: [https://awscli.amazonaws.com/awscli-exe-linux-x86\\_64.zip](https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip)

## Linux ARM

- Gunakan **curl** perintah - -o Opsi menentukan nama file yang paket download ditulis. Opsi pada perintah contoh berikut menulis file yang diunduh ke direktori saat ini dengan nama lokalawscliv2.zip.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip" -o  
"awscliv2.zip"
```

- Mengunduh dari URL — Untuk mengunduh penginstal dengan browser Anda, gunakan URL berikut: <https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip>

## 2. (Opsional) Memverifikasi integritas file zip yang Anda unduh

Jika Anda memilih untuk mengunduh paket AWS CLI penginstal secara manual pada langkah-langkah .zip di atas, Anda dapat menggunakan langkah-langkah berikut untuk memverifikasi tanda tangan dengan menggunakan alat ini. GnuPG

.zipFile paket AWS CLI installer ditandatangani secara kriptografis menggunakan tanda tangan PGP. Jika ada kerusakan atau perubahan file, verifikasi ini gagal dan Anda tidak boleh melanjutkan dengan instalasi.

- a. Unduh dan instal gpg perintah menggunakan manajer paket Anda. Untuk informasi selengkapnya GnuPG, lihat situs web [GnuPG](#).
- b. Untuk membuat file kunci publik, buat file teks dan tempel dalam teks berikut.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
mQINBF2Cr7UBEADJZHcgus0J17ENSyUmXh85z0TRV0xJorM2B/JL0kH0yigQ1uUG
ZMLhEnAG0bYatdrKP+3H911vK050pXwn0/R7fB/FSTouki4ciIx50uLlnJZIxSzx
PqG10mkxImLnbGwoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+Lb1HAGet5G
TfnxEKJ8soPLyWmwDH6HWcnjZ/aIQRBTIQ05uVeEoYxSh6w0ai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2bnfgp3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU00UCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
94zkcgJ0z3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3Tqgvdx4bdkhf5cH+7NtW0
lrFj6UwAsGukBTA0xC01/dnSmZhJ7Z1KmEWilro/g0rjt0xqRQut1IqG22TaqoPG
fYVN+en3ZwbT97kcgZDwqbuykNt64oZwc4XKCa3mprEGC3IbJTBFqglXmZ719ywG
EEUJY01b2XrSuPwm139beWdKM8kzr10jn10m6+lpTRCBfo0wa9F8YZRhHPAkWkKX
XDe0GpWrj4oh0x0d2GWkyV5xyN14p2tQ0Cd00Dmz80yUTgRpPVQUt0EhXQARAQAB
tCFBV1MgQ0xJIFR1YW0gPGF3cy1jbG1AYW1hem9uLmNvbT6JAlQEewEiAD4CGwMF
CwkIBwIGFQoJCAcCBByCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUC
ZMKcEgUJCSEf3QAKCRCmMQrMRnJHXCiLD/4vior9J5tB+icri5WbDudS3ak/ve4q
XS6ZLm5S81+CBxy5aLQUlyFhuuaEHDC11fG780duxatzeHENASYVo3mmKNwrCBza
NJaeaWKLGT0MKwBSP5aa3dva8P/4oUP9GsQn0uWoXwNDwfrMbNI8gn+jC/3MigW
vD3fu6zC0WWLITNv2SJoQ1wILmb/uGfha68o4iTB0vcftVRua06DyqF+CrHX/0j0
k1EDQFMY9M4tsYT7X8NwfI8Vmc89nzpvL9fwd44WwpKIw1FBZP8S0sgDx2xDsxv
L8kM2Gt0iH0cHqF0+V7xtTKZyloLiDbJKhu80Kc+YC/TmozD8oeGU2rEFXfLegwS
zT9N+jB38+dqaP9pRDSi45iGqyA8yavVBabPL0IQ9jU6eIV+kmcjIjcun/Uo8SjJ
0xQAsm41rxPaKV6vJUn10wVnuhSkKk8mzN01SZwu7Hua6rdcCaGeB8uJ44AP3QzW
BNnrjtoN6A1N0D2wFmfE/YL/rHPxU1XwPntubYB/t3rXFL7ENQ00QH0KVXgRC1ey
sHMglg46c+nQLRzVTshjDjmtzvH9rcV9RKR0PetEggzCoD89veDA9jPR2Kw6RYkS
XzYm2fEv16/HRNYt7hJzneFqRIjHW5qAgSs/bcaRWpAU/QQzzJPVKCQNr4y0weyG
B8HctGjfoD0p1A==
=gdMc
-----END PGP PUBLIC KEY BLOCK-----
```

Sebagai referensi, berikut ini adalah rincian kunci publik.

```
Key ID:          A6310ACC4672475C
Type:            RSA
Size:           4096/4096
Created:        2019-09-18
Expires:        2024-07-26
User ID:        AWS CLI Team <aws-cli@amazon.com>
Key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
```

- c. Impor kunci AWS CLI publik dengan perintah berikut, ganti *public-key-file-name* dengan nama file kunci publik yang Anda buat.

```
$ gpg --import public-key-file-name
```

```
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
imported
gpg: Total number processed: 1
gpg:             imported: 1
```

- d. Unduh file AWS CLI tanda tangan untuk paket yang Anda unduh. Ini memiliki jalur dan nama yang sama dengan .zip file yang sesuai dengannya, tetapi memiliki ekstensi .sig. Dalam contoh berikut, kami menyimpannya ke direktori saat ini sebagai file bernama `awscliv2.sig`.

#### Linux x86 (64-bit)

Untuk versi terbaru AWS CLI, gunakan blok perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip.sig
```

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi `2.0.30` akan `awscli-exe-linux-x86_64-2.0.30.zip.sig` menghasilkan perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip.sig
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

#### Linux ARM

Untuk versi terbaru AWS CLI, gunakan blok perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip.sig
```

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi `2.0.30` akan `awscli-exe-linux-aarch64-2.0.30.zip.sig` menghasilkan perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip.sig
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

- e. Verifikasi tanda tangan, meneruskan nama yang diunduh `.sig` dan `.zip` file sebagai parameter ke `gpg` perintah.

```
$ gpg --verify awscliv2.sig awscliv2.zip
```

Outputnya akan terlihat serupa dengan yang berikut ini:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672
475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:                There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
```

#### Important

Peringatan dalam output diharapkan dan tidak menunjukkan masalah. Itu terjadi karena tidak ada rantai kepercayaan antara kunci PGP pribadi Anda (jika Anda memilikinya) dan kunci AWS CLI PGP. Untuk informasi selengkapnya, lihat [Web kepercayaan](#).

3. Buka zip penginstal. Jika distribusi Linux Anda tidak memiliki `unzip` perintah bawaan, gunakan yang setara dengan `unzip`. Contoh perintah berikut membuka `ritsleting` paket dan membuat direktori bernama `aws` di bawah direktori saat ini.

```
$ unzip awscliv2.zip
```

#### Note

Saat memperbarui dari versi sebelumnya, `unzip` perintah meminta untuk menimpa file yang ada. Untuk melewati petunjuk ini, seperti dengan otomatisasi skrip, gunakan bendera `-u` pembaruan untuk `unzip`. Bendera ini secara otomatis memperbarui file yang ada dan membuat yang baru sesuai kebutuhan.



```
$ unzip -u awscliv2.zip
```

4. Jalankan program instal. Perintah instalasi menggunakan file bernama `install` dalam direktori yang baru di-unzipaws. Secara default, semua file diinstal/`usr/local/aws-cli`, dan tautan simbolis dibuat di `/usr/local/bin`. Perintah termasuk `sudo` untuk memberikan izin menulis ke direktori tersebut.

```
$ sudo ./aws/install
```

Anda dapat menginstal tanpa `sudo` jika Anda menentukan direktori yang sudah Anda miliki izin menulis. Gunakan instruksi berikut untuk `install` perintah untuk menentukan lokasi instalasi:

- Pastikan jalur yang Anda berikan ke `-b` parameter `-i` dan tidak berisi nama volume atau nama direktori yang berisi karakter spasi atau karakter spasi putih lainnya. Jika ada ruang, instalasi gagal.
- `--install-dir` atau `-i` — Opsi ini menentukan direktori untuk menyalin semua file ke.  
  
Nilai default-nya adalah `/usr/local/aws-cli`.
- `--bin-dir` atau `-b` — Opsi ini menentukan bahwa aws program utama dalam direktori `install` secara simbolis terkait dengan file `aws` di jalur yang ditentukan. Anda harus memiliki izin menulis ke direktori yang ditentukan. Membuat symlink ke direktori yang sudah ada di jalur Anda menghilangkan kebutuhan untuk menambahkan direktori `install` ke `$PATH` variabel pengguna.  
  
Nilai default-nya adalah `/usr/local/bin`.

```
$ ./aws/install -i /usr/local/aws-cli -b /usr/local/bin
```

#### Note

Untuk memperbarui instalasi Anda saat ini AWS CLI, tambahkan symlink yang ada dan informasi installer untuk membuat `install` perintah dengan parameter `--update`

```
$ sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-  
cli --update
```

Untuk menemukan direktori symlink dan instalasi yang ada, gunakan langkah-langkah berikut:

1. Gunakan `which` perintah untuk menemukan symlink Anda. Ini memberi Anda jalur untuk digunakan dengan `--bin-dir` parameter.

```
$ which aws
/usr/local/bin/aws
```

2. Gunakan `ls` perintah untuk menemukan direktori yang ditunjuk symlink Anda. Ini memberi Anda jalur untuk digunakan dengan `--install-dir` parameter.

```
$ ls -l /usr/local/bin/aws
lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/
local/aws-cli/v2/current/bin/aws
```

5. Konfirmasikan instalasi dengan perintah berikut.

```
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## macOS

### Instal dan perbarui persyaratan

- Kami mendukung versi macOS 10.9 dan yang lebih baru. AWS CLI Untuk informasi selengkapnya, lihat [pembaruan kebijakan dukungan macOS untuk AWS CLI v2 di Blog](#) Alat AWS Pengembang.
- Karena AWS tidak memelihara repositori pihak ketiga, kami tidak dapat menjamin bahwa repositori tersebut berisi versi terbaru dari. AWS CLI

### Instal atau perbarui AWS CLI

Jika Anda memperbarui ke versi terbaru, gunakan metode instalasi yang sama dengan yang Anda gunakan dalam versi Anda saat ini. Anda dapat menginstal AWS CLI di macOS dengan cara berikut.

## GUI installer

Langkah-langkah berikut menunjukkan cara menginstal versi terbaru AWS CLI dengan menggunakan antarmuka pengguna macOS standar dan browser Anda.

1. Di browser Anda, unduh file macOS pkg: <https://awscli.amazonaws.com/AWSCLIV2.pkg>
2. Jalankan file yang Anda unduh dan ikuti petunjuk di layar. Anda dapat memilih untuk menginstal dengan cara berikut: AWS CLI
  - Untuk semua pengguna di komputer (memerlukan **sudo**)
    - Anda dapat menginstal ke folder apa pun, atau memilih folder default yang direkomendasikan `/usr/local/aws-cli`.
    - Pemasang secara otomatis membuat symlink di tautan `/usr/local/bin/aws` tersebut ke program utama di folder instalasi yang Anda pilih.
  - Hanya untuk pengguna saat ini (tidak memerlukan **sudo**)
    - Anda dapat menginstal ke folder mana pun yang Anda memiliki izin menulis.
    - Karena izin pengguna standar, setelah penginstal selesai, Anda harus secara manual membuat file symlink di Anda `$PATH` yang menunjuk ke `aws` dan `aws_completer` program dengan menggunakan perintah berikut pada prompt perintah. Jika Anda `$PATH` menyertakan folder yang dapat Anda tulis, Anda dapat menjalankan perintah berikut tanpa `sudo` jika Anda menentukan folder itu sebagai jalur target. Jika Anda tidak memiliki folder yang dapat ditulis di dalam Anda `$PATH`, Anda harus menggunakan `sudo` perintah untuk mendapatkan izin untuk menulis ke folder target yang ditentukan. Lokasi default untuk symlink adalah `/usr/local/bin/`.

```
$ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
$ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/
aws_completer
```

### Note

Anda dapat melihat log debug untuk instalasi dengan menekan `Cmd+L` di mana saja di installer. Ini membuka panel log yang memungkinkan Anda memfilter dan menyimpan log. File log juga disimpan secara otomatis ke `/var/log/install.log`.

3. Untuk memverifikasi bahwa shell dapat menemukan dan menjalankan aws perintah di Anda\$PATH, gunakan perintah berikut.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Darwin/23.3.0 botocore/2.4.5
```

Jika aws perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Command line installer - All users

Jika Anda memiliki sudo izin, Anda dapat menginstal AWS CLI untuk semua pengguna di komputer. Kami menyediakan langkah-langkah dalam satu grup yang mudah disalin dan ditempel. Lihat deskripsi setiap baris dalam langkah-langkah berikut.

```
$ curl "https://awscli.amazonaws.com/AWSCLIV2.pkg" -o "AWSCLIV2.pkg"
$ sudo installer -pkg AWSCLIV2.pkg -target /
```

## Petunjuk instalasi terpandu

1. Unduh file menggunakan curl perintah. -oOpsi ini menentukan nama file yang paket download ditulis. Dalam contoh ini, file ditulis ke AWSCLIV2.pkg dalam folder saat ini.

```
$ curl "https://awscli.amazonaws.com/AWSCLIV2.pkg" -o "AWSCLIV2.pkg"
```

2. Jalankan installer program macOS standar, tentukan .pkg file yang diunduh sebagai sumbernya. Gunakan -pkg parameter untuk menentukan nama paket yang akan diinstal, dan -target / parameter untuk drive mana untuk menginstal paket. File diinstal ke/usr/local/aws-cli, dan symlink secara otomatis dibuat di/usr/local/bin. Anda harus menyertakan sudo perintah untuk memberikan izin menulis ke folder tersebut.

```
$ sudo installer -pkg ./AWSCLIV2.pkg -target /
```

Setelah instalasi selesai, log debug ditulis ke/var/log/install.log.

3. Untuk memverifikasi bahwa shell dapat menemukan dan menjalankan aws perintah di Anda\$PATH, gunakan perintah berikut.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Darwin/23.3.0 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Command line - Current user

1. Untuk menentukan folder mana AWS CLI yang diinstal, Anda harus membuat file XML dengan nama file apa pun. File ini adalah file berformat XML yang terlihat seperti contoh berikut. Biarkan semua nilai seperti yang ditunjukkan, kecuali Anda harus mengganti jalur `/Users/MyUsername` di baris 9 dengan jalur ke folder yang ingin Anda instal. AWS CLI Folder harus sudah ada, atau perintah gagal. Contoh XHTML berikut, bernama `choices.xml`, menentukan installer untuk menginstal dalam folder `/Users/myusername`, AWS CLI di mana ia membuat folder bernama `aws-cli`

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
  <array>
    <dict>
      <key>choiceAttribute</key>
      <string>customLocation</string>
      <key>attributeSetting</key>
      <string>/Users/myusername</string>
      <key>choiceIdentifier</key>
      <string>default</string>
    </dict>
  </array>
</plist>
```

2. Unduh pkg penginstal menggunakan `curl` perintah. `-o` Opsi ini menentukan nama file yang paket download ditulis. Dalam contoh ini, file ditulis ke `AWSCLI2.pkg` dalam folder saat ini.

```
$ curl "https://awscli.amazonaws.com/AWSCLI2.pkg" -o "AWSCLI2.pkg"
```

3. Jalankan `installer` program macOS standar dengan opsi berikut:
  - Tentukan nama paket yang akan diinstal dengan menggunakan `-pkg` parameter.
  - Tentukan penginstalan ke pengguna saat ini hanya dengan mengatur `-target` parameter ke `CurrentUserHomeDirectory`.
  - Tentukan jalur (relatif terhadap folder saat ini) dan nama file XHTML yang Anda buat di `-applyChoiceChangesXML` parameter.

Contoh berikut menginstal AWS CLI dalam folder `/Users/myusername/aws-cli`.

```
$ installer -pkg AWSCLIV2.pkg \  
            -target CurrentUserHomeDirectory \  
            -applyChoiceChangesXML choices.xml
```

4. Karena izin pengguna standar biasanya tidak mengizinkan penulisan ke folder di folder `Anda$PATH`, penginstal dalam mode ini tidak mencoba menambahkan symlink ke dan program `aws` `aws_completer`. Agar dapat berjalan dengan benar, Anda harus membuat symlink secara manual setelah penginstal selesai. AWS CLI Jika Anda `$PATH` menyertakan folder yang dapat Anda tulis dan Anda menentukan folder sebagai jalur target, Anda dapat menjalankan perintah berikut tanpa `sudo`. Jika Anda tidak memiliki folder yang dapat ditulis di `Anda$PATH`, Anda harus menggunakan izin `sudo` untuk menulis ke folder target yang ditentukan. Lokasi default untuk symlink adalah `/usr/local/bin/`. Ganti folder `installed` dengan jalur ke AWS CLI instalasi Anda.

```
$ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws  
$ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/  
aws_completer
```

Setelah instalasi selesai, log debug ditulis ke `/var/log/install.log`.

5. Untuk memverifikasi bahwa shell dapat menemukan dan menjalankan `aws` perintah di `Anda$PATH`, gunakan perintah berikut.

```
$ which aws  
/usr/local/bin/aws  
$ aws --version  
aws-cli/2.15.30 Python/3.11.6 Darwin/23.3.0 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Windows

### Instal dan perbarui persyaratan

- Kami mendukung versi AWS CLI Windows 64-bit yang didukung Microsoft.
- Hak admin untuk menginstal perangkat lunak

### Instal atau perbarui AWS CLI

Untuk memperbarui instalasi Anda saat ini AWS CLI di Windows, unduh penginstal baru setiap kali Anda memperbarui untuk menimpa versi sebelumnya. AWS CLI diperbarui secara berkala. Untuk melihat kapan versi terbaru dirilis, lihat [Changelog AWS CLI versi 2](#) aktif. GitHub

1. Unduh dan jalankan penginstal AWS CLI MSI untuk Windows (64-bit):

<https://awscli.amazonaws.com/AWSCLIV2.msi>

Atau, Anda dapat menjalankan `msiexec` perintah untuk menjalankan installer MSI.

```
C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi
```

Untuk berbagai parameter yang dapat digunakan `msiexec`, lihat [msiexec](#) di situs web Microsoft Docs. Misalnya, Anda dapat menggunakan `/qn` bendera untuk instalasi diam.

```
C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi /qn
```

2. Untuk mengkonfirmasi instalasi, buka menu Start, cari `cmd` untuk membuka jendela command prompt, dan pada command prompt gunakan `aws --version` perintah.

```
C:\> aws --version
aws-cli/2.15.30 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
```

Jika Windows tidak dapat menemukan program, Anda mungkin perlu menutup dan membuka kembali jendela prompt perintah untuk menyegarkan jalur, atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi

Jika Anda menemukan masalah setelah menginstal atau menghapus instalasi AWS CLI, lihat [Memecahkan masalah kesalahan](#) untuk langkah-langkah pemecahan masalah. Untuk langkah pemecahan masalah yang paling relevan, lihat, [the section called “Perintah tidak ditemukan kesalahan”](#) [the section called “Perintah aws --version”](#) mengembalikan versi yang berbeda dari yang Anda instal”, dan [the section called “Perintah aws --version”](#) mengembalikan versi setelah mencopot pemasangan AWS CLI”

### Langkah selanjutnya

Setelah Anda berhasil menginstal AWS CLI, Anda dapat menghapus file installer yang diunduh dengan aman. Setelah menyelesaikan langkah-langkah [the section called “Prasyarat”](#) dan menginstal AWS CLI, Anda harus melakukan [the section called “Pengaturan”](#).

## Instal rilis sebelumnya dari AWS CLI versi 2

Topik ini menjelaskan cara menginstal rilis sebelumnya dari AWS Command Line Interface versi 2 (AWS CLI) pada sistem operasi yang didukung. Untuk informasi tentang rilis AWS CLI versi 2, lihat [Changelog AWS CLI versi 2](#) aktif. GitHub

AWS CLI instruksi instalasi versi 2:

### Linux

#### Persyaratan instalasi

- Anda tahu rilis AWS CLI versi 2 mana yang ingin Anda instal. Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di. GitHub
- Anda harus dapat mengekstrak atau “unzip” paket yang diunduh. Gunakan ekuivalen jika sistem pengoperasian Anda tidak memiliki perintah unzip.
- AWS CLI Versi 2 menggunakan `libc`, `groff`, dan `less`. Ini disertakan secara default di sebagian besar distribusi utama Linux.
- Kami mendukung AWS CLI versi 2 pada versi 64-bit dari distribusi terbaru CentOS, Fedora, Ubuntu, Amazon Linux 1, Amazon Linux 2 dan Linux ARM.
- Karena AWS tidak memelihara repositori pihak ketiga, kami tidak dapat menjamin bahwa repositori tersebut berisi versi terbaru dari. AWS CLI



## Petunjuk pemasangan

Ikuti langkah-langkah ini dari baris perintah untuk menginstal AWS CLI di Linux.

Kami menyediakan langkah-langkah dalam satu grup yang mudah disalin dan ditempel berdasarkan apakah Anda menggunakan Linux 64-bit atau Linux ARM. Lihat deskripsi setiap baris dalam langkah-langkah berikut.

### Linux x86 (64-bit)

#### Note

(Opsional) Perintah berikut memblokir unduhan dan menginstal AWS CLI tanpa terlebih dahulu memverifikasi integritas unduhan Anda. Untuk memverifikasi integritas unduhan Anda, gunakan petunjuk langkah demi langkah di bawah ini.

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

Untuk menginstal AWS CLI, jalankan perintah berikut.

Untuk menentukan versi, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-x86_64-2.0.30.zip` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip" -o
  "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Untuk memperbarui instalasi Anda saat ini AWS CLI, tambahkan symlink yang ada dan informasi installer untuk membuat `install` perintah menggunakan `--bin-dir`, `--install-dir`, dan parameter. *--update* **Blok perintah berikut menggunakan contoh symlink dari `/usr/local/bin` dan contoh lokasi installer `/usr/local/aws-cli`.**

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip" -o
  "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

## Linux ARM

### Note

(Opsional) Perintah berikut memblokir unduhan dan menginstal AWS CLI tanpa terlebih dahulu memverifikasi integritas unduhan Anda. Untuk memverifikasi integritas unduhan Anda, gunakan petunjuk langkah demi langkah di bawah ini.

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

Untuk menginstal AWS CLI, jalankan perintah berikut.

Untuk menentukan versi, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-aarch64-2.0.30.zip` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip" -o
  "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Untuk memperbarui instalasi Anda saat ini AWS CLI, tambahkan symlink yang ada dan informasi installer untuk membuat `install` perintah menggunakan `--bin-dir`, `--install-dir`, dan parameter. `--update` *Blok perintah berikut menggunakan contoh symlink dari /usr/local/bin dan contoh lokasi installer /usr/local/aws-cli.*

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip" -o
  "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

1. Unduh file instalasi dengan salah satu cara berikut:

Linux x86 (64-bit)

- Gunakan **curl** perintah - `-o` Opsi menentukan nama file yang paket download ditulis. Opsi pada perintah contoh berikut menulis file yang diunduh ke direktori saat ini dengan nama `lokalawscliv2.zip`.

Untuk menentukan versi, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-x86_64-2.0.30.zip` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip" -o  
"awscliv2.zip"
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

- Download dari URL —

Di browser Anda, unduh versi spesifik Anda AWS CLI dengan menambahkan tanda hubung dan nomor versi ke nama file.

```
https://awscli.amazonaws.com/awscli-exe-linux-x86_64-version.number.zip
```

Untuk contoh ini nama file untuk versi **2.0.30** akan menjadi `awscli-exe-linux-x86_64-2.0.30.zip` menghasilkan tautan berikut: [https://awscli.amazonaws.com/awscli-exe-linux-x86\\_64-2.0.30.zip](https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip)

## Linux ARM

- Gunakan `curl` perintah - `-o` Opsi menentukan nama file yang paket download ditulis. Opsi pada perintah contoh berikut menulis file yang diunduh ke direktori saat ini dengan nama `lokalawscliv2.zip`.

Untuk menentukan versi, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-aarch64-2.0.30.zip` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip" -o  
"awscliv2.zip"  
unzip awscliv2.zip  
sudo ./aws/install
```

- Download dari URL —

Di browser Anda, unduh versi spesifik Anda AWS CLI dengan menambahkan tanda hubung dan nomor versi ke nama file.

```
https://awscli.amazonaws.com/awscli-exe-linux-aarch64-version.number.zip
```

Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-aarch64-2.0.30.zip` menghasilkan tautan berikut: <https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip>

## 2. (Opsional) Memverifikasi integritas file zip yang Anda unduh

Jika Anda memilih untuk mengunduh paket AWS CLI penginstal secara manual pada langkah-langkah .zip di atas, Anda dapat menggunakan langkah-langkah berikut untuk memverifikasi tanda tangan dengan menggunakan alat ini. GnuPG

.zipFile paket AWS CLI installer ditandatangani secara kriptografis menggunakan tanda tangan PGP. Jika ada kerusakan atau perubahan file, verifikasi ini gagal dan Anda tidak boleh melanjutkan dengan instalasi.

- a. Unduh dan instal gpg perintah menggunakan manajer paket Anda. Untuk informasi selengkapnyaGnuPG, lihat situs web [GnuPG](#).
- b. Untuk membuat file kunci publik, buat file teks dan tempel dalam teks berikut.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
mQINBF2Cr7UBEADJZHcgus0J17ENSyumXh85z0TRV0xJorM2B/JL0kH0yigQ1uUG
ZMLhENaG0bYatdrKP+3H911vK050pXwn0/R7fB/FSouki4ciIx50uLlnJZIxSzx
PqG10mkxImLnbGwoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+Lb1HAget5G
TfNxEKJ8soPLyWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6w0ai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2nbfpg3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU00UCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
94zkcgJ0z3KbZGYMiTh277Fvj9zzvZsbMCedV1BTg3Tqgvdx4bdkhf5cH+7NtW0
lrFj6UwAsGukBTA0xC01/dnSmZhJ7Z1KmEWilro/g0rjt0xqRQut1IqG22TaqoPG
fYVN+en3ZwbT97kcgZDwqbuykNt64oZWc4XKCa3mprEGC3IbJTBFqg1XmZ719ywG
EEUJY01b2XrSuPWml39beWdKM8kzr10jnl0m6+lpTRCBfo0wa9F8YZRhHPAkWkKX
XDe0GpWRj4oh0x0d2GwkyV5xyN14p2tQ0Cd00Dmz80yUTgRpPVQUt0EhXQARAQAB
tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbG1AYW1hem9uLmNvbT6JAlQEEwEiAD4CGwMF
CwkIBwIGFQoJCAwCBByCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUC
ZMKcEgUJCSEf3QAKCRCmMQrMRnJHXCiLD/4vior9J5tB+icri5WbDudS3ak/ve4q
XS6ZLm5S81+CBxy5aLQUlyFhuuaEHDC11fG780duxatzeHENASYVo3mmKNwrCBza
NJaeaWKLGT0MKwBSP5aa3dva8P/4oUP9GsQn0uWoXwNDwfrMbNI8gn+jC/3MigW
vD3fu6zCOWWLITNv2SJoQ1wILmb/uGfha68o4iTB0vcftVRuao6DyqF+CrHX/0j0
k1EDQFM9M4tsYT7X8NwfI8Vmc89nzpVl9fwda44WwpKIw1FBZP8S0sgDx2xDsXv
L8kM2Gt0iH0cHqF0+V7xtTKZy1o1iDbJKhu80Kc+YC/TmozD8oeGU2rEFXfLegwS
```

```

zT9N+jB38+dqaP9pRDsi45iGqyA8yavVBabpL0IQ9jU6eIV+kmcjIjcun/Uo8SjJ
0xQAsm41rxPaKV6vJUn10wVNUhSkKk8mzN01SZwu7Hua6rxcCaGeB8uJ44AP3QzW
BNnrjtoN6A1N0D2wFmfE/YL/rHPxU1XwPntubYB/t3rXFL7ENQ00QH0KVXgRC1ey
sHMglg46c+nQLRzVTshjDjmtzv9rcV9RKR0PetEggzCoD89veDA9jPR2Kw6RYkS
XzYm2fEv16/HRNYt7hJzneFqRIjHW5qAgSs/bcaRWpAU/QQzzJPVKCQNr4y0weyg
B8HCtGjfod0p1A==
=gdMc
-----END PGP PUBLIC KEY BLOCK-----

```

Sebagai referensi, berikut ini adalah rincian kunci publik.

```

Key ID:          A6310ACC4672
Type:            RSA
Size:            4096/4096
Created:         2019-09-18
Expires:         2024-07-26
User ID:         AWS CLI Team <aws-cli@amazon.com>
Key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C

```

- c. Impor kunci AWS CLI publik dengan perintah berikut, ganti *public-key-file-name* dengan nama file kunci publik yang Anda buat.

```

$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
imported
gpg: Total number processed: 1
gpg:          imported: 1

```

- d. Unduh file AWS CLI tanda tangan untuk paket yang Anda unduh. Ini memiliki jalur dan nama yang sama dengan .zip file yang sesuai dengannya, tetapi memiliki ekstensi .sig. Dalam contoh berikut, kami menyimpannya ke direktori saat ini sebagai file bernama `awscliv2.sig`.

Linux x86 (64-bit)

Untuk versi terbaru AWS CLI, gunakan blok perintah berikut:

```

$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip.sig

```

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-x86_64-2.0.30.zip.sig` menghasilkan perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip.sig
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

## Linux ARM

Untuk versi terbaru AWS CLI, gunakan blok perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip.sig
```

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-aarch64-2.0.30.zip.sig` menghasilkan perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip.sig
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

- e. Verifikasi tanda tangan, meneruskan nama yang diunduh `.sig` dan `.zip` file sebagai parameter ke `gpg` perintah.

```
$ gpg --verify awscliv2.sig awscliv2.zip
```

Outputnya akan terlihat serupa dengan yang berikut ini:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672
 475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:                There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
```

**⚠ Important**

Peringatan dalam output diharapkan dan tidak menunjukkan masalah. Itu terjadi karena tidak ada rantai kepercayaan antara kunci PGP pribadi Anda (jika Anda memilikinya) dan kunci AWS CLI PGP. Untuk informasi selengkapnya, lihat [Web kepercayaan](#).

3. Buka zip penginstal. Jika distribusi Linux Anda tidak memiliki unzip perintah bawaan, gunakan yang setara dengan unzip. Contoh perintah berikut membuka ritsleting paket dan membuat direktori bernama aws di bawah direktori saat ini.

```
$ unzip awscliv2.zip
```

4. Jalankan program instal. Perintah instalasi menggunakan file bernama `install` dalam direktori yang baru di-unzipaws. Secara default, semua file diinstal/`/usr/local/aws-cli`, dan tautan simbolis dibuat di `/usr/local/bin` Perintah termasuk `sudo` untuk memberikan izin menulis ke direktori tersebut.

```
$ sudo ./aws/install
```

Anda dapat menginstal tanpa `sudo` jika Anda menentukan direktori yang sudah Anda miliki izin menulis. Gunakan instruksi berikut untuk `install` perintah untuk menentukan lokasi instalasi:

- Pastikan jalur yang Anda berikan ke `-b` parameter `-i` dan tidak berisi nama volume atau nama direktori yang berisi karakter spasi atau karakter spasi putih lainnya. Jika ada ruang, instalasi gagal.
- `--install-dir` atau `-i` — Opsi ini menentukan direktori untuk menyalin semua file ke.

Nilai default-nya adalah `/usr/local/aws-cli`.

- `--bin-dir` atau `-b` — Opsi ini menentukan bahwa aws program utama dalam direktori `install` secara simbolis terkait dengan file `aws` di jalur yang ditentukan. Anda harus memiliki izin menulis ke direktori yang ditentukan. Membuat symlink ke direktori yang sudah ada di jalur Anda menghilangkan kebutuhan untuk menambahkan direktori `install` ke `$PATH` variabel pengguna.

Nilai default-nya adalah `/usr/local/bin`.

```
$ ./aws/install -i /usr/local/aws-cli -b /usr/local/bin
```

### Note

Untuk memperbarui instalasi AWS CLI versi 2 Anda saat ini ke versi yang lebih baru, tambahkan symlink dan informasi penginstal yang ada untuk membuat `install` perintah dengan parameter. `--update`

```
$ sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --update
```

Untuk menemukan direktori symlink dan instalasi yang ada, gunakan langkah-langkah berikut:

1. Gunakan `which` perintah untuk menemukan symlink Anda. Ini memberi Anda jalur untuk digunakan dengan `--bin-dir` parameter.

```
$ which aws
/usr/local/bin/aws
```

2. Gunakan `ls` perintah untuk menemukan direktori yang ditunjuk symlink Anda. Ini memberi Anda jalur untuk digunakan dengan `--install-dir` parameter.

```
$ ls -l /usr/local/bin/aws
lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/local/aws-cli/v2/current/bin/aws
```

5. Konfirmasikan instalasi dengan perintah berikut.

```
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)



**(Opsional) Memverifikasi integritas file zip yang Anda unduh**

Jika Anda memilih untuk mengunduh paket penginstal AWS CLI versi 2 secara manual .zip pada langkah-langkah di atas, Anda dapat menggunakan langkah-langkah berikut untuk memverifikasi tanda tangan dengan menggunakan alat ini. GnuPG

.zipFile paket installer AWS CLI versi 2 ditandatangani secara kriptografis menggunakan tanda tangan PGP. Jika ada kerusakan atau perubahan file, verifikasi ini gagal dan Anda tidak boleh melanjutkan dengan instalasi.

1. Unduh dan instal gpg perintah menggunakan manajer paket Anda. Untuk informasi selengkapnyaGnuPG, lihat situs web [GnuPG](#).
2. Untuk membuat file kunci publik, buat file teks dan tempel dalam teks berikut.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
mQINBF2Cr7UBEADJZHcgus0Jl7ENSyumXh85z0TRV0xJorM2B/JL0kH0yigQ1uUG
ZMLhENaG0bYatdrKP+3H911vK050pXwn0/R7fB/FSTouki4ciIx50uLlnJZIxSzx
PqG10mkxImLNBGwoi6Lto0LYxqHN2iQtz1wTVmq9733zd3XfcXrZ3+Lb1HAgEt5G
TfNxEKJ8soPLyWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6w0ai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2nbfgp3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU00UCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
94zkcqJ0z3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3TqgvdX4bdkhf5cH+7NtW0
lrFj6UwAsGukBTA0xC01/dnSmZhJ7Z1KmEWilro/g0rjt0xqRQut1IqG22TaqoPG
fYVN+en3ZwbT97kcgZDwqbuykNt64oZwc4XKCa3mprEGC3IbJTBfqq1XmZ719ywG
EEUJY01b2XrSuPwm139beWdKM8kzr10jnl0m6+lpTRCBfo0wa9F8YZRhHPakwKkX
XDe0GpWrj4oh0x0d2GWkyV5xyN14p2tQ0Cd00Dmz80yUTgRpPVQuT0EhXQARAQAB
tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbG1AYW1hem9uLmNvbT6JAlQEEwEiAD4CGwMF
CwkIBwIGFQoJCAcCBBYCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUc
ZMKcEgUJCSEf3QAKCRCmMQrMRnJHXciLD/4vior9J5tB+icri5WbDudS3ak/ve4q
XS6ZLm5S81+CBxy5aLQUlyFhuaaEHDC11fG780duxatzeHENASYVo3mmKNwrCBza
NJaeaWKLGT0MKwBSP5aa3dva8P/4oUP9GsQn0uWoXwNDWfrMbNI8gn+jC/3MigW
vD3fu6zC0WWLITNv2SJoQ1wILmb/uGfha68o4iTB0vcftVRuao6DyqF+CrHX/0j0
k1EDQFMY9M4tsYT7X8NwfI8Vmc89nzpvL9fwda44WwpKIw1FBZP8S0sgDx2xDsxv
L8kM2Gt0iH0cHqF0+V7xtTKZy1o1iDbJKhu80Kc+YC/TmozD8oeGU2rEFXfLegwS
zT9N+jB38+dqaP9pRdsi45iGqyA8yavVBabpL0IQ9jU6eIV+kmcjIjcun/Uo8SjJ
0xQAsm41rxPaKv6vJUn10wVNuhSkKk8mzN01SZwu7Hua6rdcCaGeB8uJ44AP3QzW
BNnrjtoN6A1N0D2wFmfE/YL/rHPxU1XwPntubYB/t3rXFL7ENQ00QH0KVXgRC1ey
sHMglg46c+nQLRzVTshjDjmtzvH9rcV9RKR0PetEggzCoD89veDA9jPR2Kw6RYkS
XzYm2fEv16/HRNYt7HJzneFqRIjHW5qAgSs/bcaRwPAU/QQzzJPVKCQNr4y0weyg
B8HCtGj fod0p1A==
=gdMc
```

```
-----END PGP PUBLIC KEY BLOCK-----
```

Sebagai referensi, berikut ini adalah rincian kunci publik.

```
Key ID:           A6310ACC4672
Type:             RSA
Size:            4096/4096
Created:         2019-09-18
Expires:        2024-07-26
User ID:         AWS CLI Team <aws-cli@amazon.com>
Key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
```

3. Impor kunci AWS CLI publik dengan perintah berikut, ganti *public-key-file-name* dengan nama file kunci publik yang Anda buat.

```
$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>" imported
gpg: Total number processed: 1
gpg:             imported: 1
```

4. Unduh file AWS CLI tanda tangan untuk paket yang Anda unduh. Ini memiliki jalur dan nama yang sama dengan .zip file yang sesuai dengannya, tetapi memiliki ekstensi .sig. Dalam contoh berikut, kami menyimpannya ke direktori saat ini sebagai file bernama `awscliv2.sig`.

Linux x86 (64-bit)

Untuk versi terbaru AWS CLI, gunakan blok perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip.sig
```

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi `2.0.30` akan `awscli-exe-linux-x86_64-2.0.30.zip.sig` menghasilkan perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip.sig
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

## Linux ARM

Untuk versi terbaru AWS CLI, gunakan blok perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip.sig
```

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `awscli-exe-linux-aarch64-2.0.30.zip.sig` menghasilkan perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip.sig
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

5. Verifikasi tanda tangan, meneruskan nama yang diunduh `.sig` dan `.zip` file sebagai parameter ke `gpg` perintah.

```
$ gpg --verify awscliv2.sig awscliv2.zip
```

Outputnya akan terlihat serupa dengan yang berikut ini:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:                There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
```

### Important

Peringatan dalam output diharapkan dan tidak menunjukkan masalah. Itu terjadi karena tidak ada rantai kepercayaan antara kunci PGP pribadi Anda (jika Anda memilikinya) dan kunci AWS CLI PGP. Untuk informasi selengkapnya, lihat [Web kepercayaan](#).

## macOS

### Persyaratan instalasi

- Anda tahu rilis AWS CLI versi 2 mana yang ingin Anda instal. Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub
- Kami mendukung AWS CLI versi 2 pada versi macOS 64-bit yang didukung Apple.
- Karena AWS tidak memelihara repositori pihak ketiga, kami tidak dapat menjamin bahwa repositori tersebut berisi versi terbaru dari AWS CLI

### Petunjuk pemasangan

Anda dapat menginstal AWS CLI versi 2 di macOS dengan cara berikut.

### GUI installer

Langkah-langkah berikut menunjukkan cara menginstal atau memperbarui ke versi terbaru AWS CLI versi 2 dengan menggunakan antarmuka pengguna macOS standar dan browser Anda. Jika Anda memperbarui ke versi terbaru, gunakan metode instalasi yang sama dengan yang Anda gunakan untuk versi Anda saat ini.

1. Di browser Anda, unduh versi spesifik Anda AWS CLI dengan menambahkan tanda hubung dan nomor versi ke nama file.

```
https://awscli.amazonaws.com/AWSCLIV2-version.number.pkg
```

Untuk contoh ini, nama file untuk versi **2.0.30** akan `AWSCLIV2-2.0.30.pkg` menghasilkan tautan berikut: <https://awscli.amazonaws.com/AWSCLIV2-2.0.30.pkg>

2. Jalankan file yang Anda unduh dan ikuti petunjuk di layar. Anda dapat memilih untuk menginstal AWS CLI versi 2 dengan cara berikut:
  - Untuk semua pengguna di komputer (memerlukan **sudo**)
    - Anda dapat menginstal ke folder apa pun, atau memilih folder default yang direkomendasikan `/usr/local/aws-cli`.
    - Pemasang secara otomatis membuat symlink di tautan `/usr/local/bin/aws` tersebut ke program utama di folder instalasi yang Anda pilih.
  - Hanya untuk pengguna saat ini (tidak memerlukan **sudo**)

- Anda dapat menginstal ke folder mana pun yang Anda memiliki izin menulis.
- Karena izin pengguna standar, setelah penginstal selesai, Anda harus secara manual membuat file symlink di Anda \$PATH yang menunjuk ke `aws` dan `aws_completer` program dengan menggunakan perintah berikut pada prompt perintah. Jika Anda \$PATH menyertakan folder yang dapat Anda tulis, Anda dapat menjalankan perintah berikut tanpa `sudo` jika Anda menentukan folder itu sebagai jalur target. Jika Anda tidak memiliki folder yang dapat ditulis di dalam Anda\$PATH, Anda harus menggunakan `sudo` perintah untuk mendapatkan izin untuk menulis ke folder target yang ditentukan. Lokasi default untuk symlink adalah `/usr/local/bin/`.

```
$ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
$ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/
aws_completer
```

#### Note

Anda dapat melihat log debug untuk instalasi dengan menekan `Cmd+L` di mana saja di installer. Ini membuka panel log yang memungkinkan Anda memfilter dan menyimpan log. File log juga disimpan secara otomatis ke `/var/log/install.log`.

3. Untuk memverifikasi bahwa shell dapat menemukan dan menjalankan `aws` perintah di Anda\$PATH, gunakan perintah berikut.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Darwin/23.3.0 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Command line installer - All users

Jika Anda memiliki `sudo` izin, Anda dapat menginstal AWS CLI versi 2 untuk semua pengguna di komputer. Kami menyediakan langkah-langkah dalam satu grup yang mudah disalin dan ditempel. Lihat deskripsi setiap baris dalam langkah-langkah berikut.

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `AWSCLI2-2.0.30.pkg` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/AWSCLI2-2.0.30.pkg" -o "AWSCLI2.pkg"
$ sudo installer -pkg AWSCLI2.pkg -target /
```

1. Unduh file menggunakan `curl` perintah. `-o` Opsi ini menentukan nama file yang paket yang diunduh ditulis. Dalam contoh ini, file ditulis ke `AWSCLI2.pkg` dalam folder saat ini.

Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file. Untuk contoh ini nama file untuk versi **2.0.30** akan `AWSCLI2-2.0.30.pkg` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/AWSCLI2-2.0.30.pkg" -o "AWSCLI2.pkg"
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

2. Jalankan `installer` program macOS standar, tentukan `.pkg` file yang diunduh sebagai sumbernya. Gunakan `-pkg` parameter untuk menentukan nama paket yang akan diinstal, dan `-target /` parameter untuk drive mana untuk menginstal paket. File diinstal ke `/usr/local/aws-cli`, dan symlink secara otomatis dibuat di `/usr/local/bin`. Anda harus menyertakan `sudo` perintah untuk memberikan izin menulis ke folder tersebut.

```
$ sudo installer -pkg ./AWSCLI2.pkg -target /
```

Setelah instalasi selesai, log debug ditulis ke `/var/log/install.log`.

3. Untuk memverifikasi bahwa shell dapat menemukan dan menjalankan `aws` perintah di `Anda$PATH`, gunakan perintah berikut.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Darwin/23.3.0 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Command line - Current user

1. Untuk menentukan folder mana AWS CLI yang diinstal, Anda harus membuat file XHTML. File ini adalah file berformat XML yang terlihat seperti contoh berikut. Biarkan semua nilai seperti yang ditunjukkan, kecuali Anda harus mengganti path `/Users/MyUsername` di baris 9 dengan path ke folder yang Anda inginkan AWS CLI versi 2 diinstal. Folder harus sudah ada, atau perintah gagal. Contoh XHTML ini menentukan bahwa installer menginstal dalam folder `/Users/myusername`, AWS CLI di mana ia membuat folder bernama `aws-cli`

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
  <array>
    <dict>
      <key>choiceAttribute</key>
      <string>customLocation</string>
      <key>attributeSetting</key>
      <string>/Users/myusername</string>
      <key>choiceIdentifier</key>
      <string>default</string>
    </dict>
  </array>
</plist>
```

2. Unduh pkg penginstal menggunakan `curl` perintah. `-o` Opsi ini menentukan nama file yang paket yang diunduh ditulis. Dalam contoh ini, file ditulis ke `AWSCLIV2.pkg` dalam folder saat ini.

Untuk versi spesifik AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file.

Untuk contoh ini nama file untuk versi `2.0.30` akan `AWSCLIV2-2.0.30.pkg` menghasilkan perintah berikut:

```
$ curl "https://awscli.amazonaws.com/AWSCLIV2-2.0.30.pkg" -o "AWSCLIV2.pkg"
```

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

3. Jalankan `installer` program macOS standar dengan opsi berikut:
  - Tentukan nama paket yang akan diinstal dengan menggunakan `-pkg` parameter.

- Tentukan penginstalan ke pengguna saat ini hanya dengan mengatur `-target` parameter ke `CurrentUserHomeDirectory`.
- Tentukan jalur (relatif terhadap folder saat ini) dan nama file XHTML yang Anda buat di `-applyChoiceChangesXML` parameter.

Contoh berikut menginstal AWS CLI dalam folder `/Users/myusername/aws-cli`.

```
$ installer -pkg AWSCLIV2.pkg \  
            -target CurrentUserHomeDirectory \  
            -applyChoiceChangesXML choices.xml
```

4. Karena izin pengguna standar biasanya tidak mengizinkan penulisan ke folder di folder `Anda$PATH`, penginstal dalam mode ini tidak mencoba menambahkan symlink ke dan program. `aws` `aws_completer` Agar dapat berjalan dengan benar, Anda harus membuat symlink secara manual setelah penginstal selesai. AWS CLI Jika Anda `$PATH` menyertakan folder yang dapat Anda tulis dan Anda menentukan folder sebagai jalur target, Anda dapat menjalankan perintah berikut tanpa `sudo`. Jika Anda tidak memiliki folder yang dapat ditulis di `Anda$PATH`, Anda harus menggunakan izin `sudo` untuk menulis ke folder target yang ditentukan. Lokasi default untuk symlink adalah `/usr/local/bin/`.

```
$ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws  
$ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/  
aws_completer
```

Setelah instalasi selesai, log debug ditulis ke `/var/log/install.log`.

5. Untuk memverifikasi bahwa shell dapat menemukan dan menjalankan `aws` perintah di `Anda$PATH`, gunakan perintah berikut.

```
$ which aws  
/usr/local/bin/aws  
$ aws --version  
aws-cli/2.15.30 Python/3.11.6 Darwin/23.3.0 botocore/2.4.5
```

Jika `aws` perintah tidak dapat ditemukan, Anda mungkin perlu me-restart terminal Anda atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)



## Windows

### Persyaratan instalasi

- Anda tahu rilis AWS CLI versi 2 mana yang ingin Anda instal. Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub
- Kami mendukung versi AWS CLI Windows 64-bit yang didukung Microsoft.
- Hak admin untuk menginstal perangkat lunak

### Petunjuk pemasangan

Untuk memperbarui instalasi AWS CLI versi 2 Anda saat ini di Windows, unduh penginstal baru setiap kali Anda memperbarui untuk menimpa versi sebelumnya. AWS CLI diperbarui secara berkala. Untuk melihat kapan versi terbaru dirilis, lihat [Changelog AWS CLI versi 2](#) aktif. GitHub

1. Unduh dan jalankan penginstal AWS CLI MSI untuk Windows (64-bit) dengan salah satu cara berikut:
  - Mengunduh dan menjalankan penginstal MSI: Untuk membuat tautan unduhan Anda untuk versi tertentu AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file.

```
https://awscli.amazonaws.com/AWSCLIV2-version.number.msi
```

Untuk contoh ini nama file untuk versi **2.0.30** akan `AWSCLIV2-2.0.30.msi` menghasilkan tautan berikut: <https://awscli.amazonaws.com/AWSCLIV2-2.0.30.msi>

- Menggunakan perintah `msiexec`: Atau, Anda dapat menggunakan penginstal MSI dengan menambahkan tautan ke perintah `msiexec` Untuk versi tertentu dari AWS CLI, tambahkan tanda hubung dan nomor versi ke nama file.

```
C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2-version.number.msi
```

Untuk contoh ini nama file untuk versi **2.0.30** akan `AWSCLIV2-2.0.30.msi` menghasilkan tautan berikut: <https://awscli.amazonaws.com/AWSCLIV2-2.0.30.msi>

```
C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2-2.0.30.msi
```

Untuk berbagai parameter yang dapat digunakan `msiexec`, lihat [msiexec](#) di situs web Microsoft Docs.

Untuk daftar versi, lihat [Changelog AWS CLI versi 2](#) di GitHub

2. Untuk mengkonfirmasi instalasi, buka menu Start, cari cmd untuk membuka jendela command prompt, dan pada command prompt gunakan `aws --version` perintah.

```
C:\> aws --version
aws-cli/2.15.30 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
```

Jika Windows tidak dapat menemukan program, Anda mungkin perlu menutup dan membuka kembali jendela prompt perintah untuk menyegarkan jalur, atau mengikuti pemecahan masalah di [Memecahkan masalah kesalahan](#)

## Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi

Jika Anda menemukan masalah setelah menginstal atau menghapus instalasi AWS CLI, lihat [Memecahkan masalah kesalahan](#) untuk langkah-langkah pemecahan masalah. Untuk langkah pemecahan masalah yang paling relevan, lihat, [the section called “Perintah tidak ditemukan kesalahan”](#) [the section called “Perintah aws --version ” mengembalikan versi yang berbeda dari yang Anda instal”](#), dan [the section called “Perintah aws --version ” mengembalikan versi setelah mencopot pemasangan AWS CLI”](#)

### Langkah selanjutnya

Setelah menyelesaikan langkah-langkah [the section called “Prasyarat”](#) dan menginstal AWS CLI, Anda harus melakukan [the section called “Pengaturan”](#).

## Membangun dan menginstal AWS CLI dari sumber

Topik ini menjelaskan cara menginstal atau memperbarui dari sumber ke rilis terbaru AWS Command Line Interface (AWS CLI) pada sistem operasi yang didukung.

Untuk informasi tentang rilis terbaru AWS CLI, lihat [Changelog AWS CLI versi 2](#) di GitHub

**⚠ Important**

AWS CLI versi 1 dan 2 menggunakan nama `aws` perintah yang sama. Jika sebelumnya Anda menginstal AWS CLI versi 1, lihat [Migrasi dari AWS CLI versi 1 ke versi 2](#).

## Topik

- [Mengapa membangun dari sumber?](#)
- [Langkah cepat](#)
- [Langkah 1: Siapkan semua persyaratan](#)
- [Langkah 2: Mengkonfigurasi instalasi AWS CLI sumber](#)
- [Langkah 3: Membangun AWS CLI](#)
- [Langkah 4: Memasang AWS CLI](#)
- [Langkah 5: Memverifikasi AWS CLI instalasi](#)
- [Contoh alur kerja](#)
- [Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi](#)
- [Langkah selanjutnya](#)

## Mengapa membangun dari sumber?

AWS CLI ini [tersedia sebagai installer pra-bangun](#) untuk sebagian besar platform dan lingkungan serta gambar Docker.

Umumnya, installer ini menyediakan cakupan untuk sebagian besar kasus penggunaan. Petunjuk untuk menginstal dari sumber adalah untuk membantu kasus penggunaan yang tidak dicakup oleh penginstal kami. Beberapa kasus penggunaan ini meliputi yang berikut:

- Pemasang pra-bangun tidak mendukung lingkungan Anda. Misalnya, ARM 32-bit tidak didukung oleh installer pra-bangun.
- Pemasang pra-bangun memiliki dependensi yang tidak dimiliki lingkungan Anda. Misalnya, Alpine Linux menggunakan [musl](#), tetapi installer saat ini memerlukan `glibc` menyebabkan installer pre-built tidak segera berfungsi.
- Pemasang pra-bangun memerlukan sumber daya yang membatasi akses lingkungan Anda. Misalnya, sistem keamanan yang diperkeras mungkin tidak memberikan izin ke memori bersama. Ini diperlukan untuk `aws` installer beku.

- Pemasang pra-bangun sering kali merupakan pemblokir untuk pengelola di manajer paket, karena kontrol penuh atas proses pembangunan untuk kode dan paket lebih disukai. Membangun dari sumber memungkinkan pengelola distribusi proses yang lebih efisien untuk menjaga pembaruan. AWS CLI Mengaktifkan pengelola memberi pelanggan lebih banyak up-to-date versi AWS CLI saat menginstal dari manajer paket pihak ke-3 seperti brew,, yum dan. apt
- Pelanggan yang menambal AWS CLI fungsionalitas memerlukan membangun dan menginstal AWS CLI from source. Ini sangat penting bagi anggota komunitas yang ingin menguji perubahan yang telah mereka buat pada sumber sebelum memberikan kontribusi perubahan ke AWS CLI GitHub repositori.

## Langkah cepat

### Note

Semua contoh kode diasumsikan berjalan dari root direktori sumber.

Untuk membangun dan menginstal AWS CLI from source, ikuti langkah-langkah di bagian ini. Ini AWS CLI memanfaatkan [GNU Autotools](#) untuk menginstal dari sumber. Dalam kasus yang paling sederhana, AWS CLI dapat diinstal dari sumber dengan menjalankan perintah contoh default dari root AWS CLI GitHub repositori.

1. [Siapkan semua persyaratan untuk lingkungan Anda](#). Ini termasuk dapat menjalankan file yang dihasilkan [GNU Autotools](#) dan Python 3.8 atau yang lebih baru diinstal.
2. Di terminal Anda, arahkan ke tingkat atas folder AWS CLI sumber dan jalankan `./configure` perintah. Perintah ini memeriksa sistem untuk semua dependensi yang diperlukan dan menghasilkan `Makefile` untuk membangun dan menginstal AWS CLI berdasarkan konfigurasi yang terdeteksi dan ditentukan.

### Linux and macOS

Contoh `./configure` perintah berikut menetapkan konfigurasi build untuk AWS CLI menggunakan pengaturan default.

```
$ ./configure
```

## Windows PowerShell

Sebelum menjalankan perintah apa pun yang memanggil MSYS2, Anda harus mempertahankan direktori kerja Anda saat ini:

```
PS C:\> $env:CHERE_INVOKING = 'yes'
```

Kemudian gunakan contoh `./configure` perintah berikut untuk mengatur konfigurasi build untuk AWS CLI menggunakan jalur lokal Anda ke executable Python Anda, menginstal ke `C:\Program Files\AWSCLI`, dan men-download semua dependensi.

```
PS C:\> C:\msys64\usr\bin\bash -lc " PYTHON='C:\path\to\python.exe' ./configure --prefix='C:\Program Files\AWSCLI' --with-download-deps "
```

Untuk detail, opsi konfigurasi yang tersedia, dan informasi pengaturan default, lihat [the section called “Langkah 2: Mengkonfigurasi instalasi AWS CLI sumber”](#) bagian.

3. Jalankan perintah `make`. Perintah ini membangun AWS CLI sesuai dengan pengaturan konfigurasi Anda.

Contoh `make` perintah berikut dibangun dengan opsi default menggunakan `./configure` pengaturan yang ada.

## Linux and macOS

```
$ make
```

## Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "make"
```

Untuk detail dan opsi build yang tersedia, lihat [the section called “Langkah 3: Membangun AWS CLI”](#) bagian.

4. Jalankan perintah `make install`. Perintah ini menginstal bawaan Anda AWS CLI ke lokasi yang dikonfigurasi di sistem Anda.

Contoh `make install` perintah berikut menginstal bawaan Anda AWS CLI dan membuat symlink di lokasi yang dikonfigurasi menggunakan pengaturan perintah default.

Linux and macOS

```
$ make install
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "make install"
```

Setelah menginstal, tambahkan jalur ke AWS CLI menggunakan yang berikut ini:

```
PS C:\> $Env: PATH +=";C:\Program Files\AWSCLI\bin\"
```

Untuk detail dan opsi pemasangan yang tersedia, lihat [the section called “Langkah 4: Memasang AWS CLI”](#) bagian.

5. Konfirmasikan AWS CLI berhasil diinstal menggunakan perintah berikut:

```
$ aws --version  
aws-cli/2.15.30 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
```

Untuk langkah-langkah pemecahan masalah untuk kesalahan penginstalan lihat bagian. [the section called “Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi”](#)

## Langkah 1: Siapkan semua persyaratan

Untuk membangun AWS CLI from source, Anda perlu yang berikut ini diselesaikan sebelumnya:

### Note

Semua contoh kode diasumsikan berjalan dari root direktori sumber.

1. Unduh AWS CLI sumbernya dengan mem-forking AWS CLI GitHub repositori atau mengunduh tarball sumber. Instruksi adalah salah satu dari yang berikut:

- Garpu dan kloning [AWS CLI repositori](#) dari GitHub Untuk informasi selengkapnya, lihat [Memfork repo](#) di GitHubDocs.
- Unduh tarball sumber terbaru di <https://awscli.amazonaws.com/awscli.tar.gz> ekstrak konten menggunakan perintah berikut:

```
$ curl -o awscli.tar.gz https://awscli.amazonaws.com/awscli.tar.gz
$ tar -xzf awscli.tar.gz
```

### Note

Untuk mengunduh versi tertentu, gunakan format tautan berikut: [https://awscli.amazonaws.com/awscli - \*versionnumber\* .tar.gz](https://awscli.amazonaws.com/awscli - versionnumber .tar.gz)

*Misalnya, untuk versi 2.10.0 tautannya adalah sebagai berikut:*

<https://awscli.amazonaws.com/awscli - 2.10.0 .tar.gz>

Versi sumber tersedia dimulai dengan versi 2.10.0 dari versi AWS CLI

(Opsional) Memverifikasi integritas file zip yang Anda unduh dengan menyelesaikan langkah-langkah berikut:

1. Anda dapat menggunakan langkah-langkah berikut untuk memverifikasi tanda tangan dengan menggunakan alat ini. GnuPG

.zipFile paket AWS CLI installer ditandatangani secara kriptografis menggunakan tanda tangan PGP. Jika ada kerusakan atau perubahan file, verifikasi ini gagal dan Anda tidak boleh melanjutkan dengan instalasi.

2. Unduh dan instal gpg perintah menggunakan manajer paket Anda. Untuk informasi selengkapnyaGnuPG, lihat situs web [GnuPG](#).
3. Untuk membuat file kunci publik, buat file teks dan tempel dalam teks berikut.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
mQINBF2Cr7UBEADJZHcgus0Jl17ENSyumXh85z0TRV0xJorM2B/JL0kH0yigQ1uUG
ZMLhENaG0bYatdrKP+3H911vK050pXwn0/R7fB/FSTouki4ciIx50uLlnJZIxSzx
PqG10mkxImLNBGwoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+Lb1HAgEt5G
TfNxEKJ8soPLYWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6w0ai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2nbfgp3DSasaLZedCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU00UCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
```

```

94zkcgJ0z3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3TqgvdX4bdkhf5cH+7NtW0
1rFj6UwAsGukBTA0xC01/dnSmZhJ7Z1KmEWi1ro/g0rjt0xqRQut1IqG22TaqpG
fYVN+en3Zwbt97kcgZDwqbuykNt64oZWc4XKCa3mprEGC3IbJTBfqq1XmZ719ywG
EEUJY01b2XrSuPwM139beWdKM8kzr10jnl0m6+1pTRCBfo0wa9F8YZRhHPAkWkKX
XDe0GpWRj4oh0x0d2GWkyV5xyN14p2tQ0Cd00Dmz80yUTgRpPVQUt0EhXQARAQAB
tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbG1AYW1hem9uLmNvbT6JAlQEEwEIAD4WIQT7
Xbd/1cEYuAURraimMQrMRnJHXAUCXYKvtQIbAwUJB4TOAAULCQgHAgYVCgkICwIE
FgIDAQIEAQIXgAAKRCRCmMQrMRnJHXJIXEACHLUIkg80uPUKgjE3jejvQSA1aWuAM
zy6fdpd1RUz6M6nmsUh0ExjVIvibEJpzK5mhuSZ41b0vJ2ZUPgCv4zs2nBd7BGJ
MxKiWgBREgVtdqZ0SzyYH4PYCJSE732x/Fw9hfnh1dMTXNcrQXzw0mmFNNegG00x
au+VnpcR5Kz3smiTrIwZbRudo1ijhCYPQ7t5Cmp9kjC6b0bvy1hSIg2xNbMAN/Do
ikebAl36uA6Y/Uczjj3GxZW4ZWeFirMidKbtqvUz2y0UFszobjiBSqZZHCreC34B
hw9bFNpuWC/0SrXgohdsc6vK50pDgDv5kM2qo9tMQ/izsAwTh/d/GzZv8H41V9e0
tEis+EpR497PaxKKh9tJf0N6Q1YLRHof5xePZt0I1S3gfvSH5hXA3HJ9yIxb8T0H
QYmVr3aIUes20i6meI3fuV36VFupwfrTKaL7VXnsrK2fq5cRvyJLNzXucg0WAjPF
RrAGLzY7nP1xeg1a0aeP+pdsqjq1PJom80Cwc1+6DWbg0jsC74WoesAqgBiT0DMB
rsa11y/q+bPzpsnWjzHV8+1/EtZmSc8ZUGSJOPkfc7h0bnfk118h+1QtKTjZme4d
H17gsBJr+opwJw/Zio2LMjQB0qlm3K1A4zFTh7wBC7He6KPQea1p2XAMgtvATtNe
YLZATHZKTJyiqA==
=vY0k
-----END PGP PUBLIC KEY BLOCK-----

```

Sebagai referensi, berikut ini adalah rincian kunci publik.

```

Key ID:           A6310ACC4672
Type:             RSA
Size:             4096/4096
Created:          2019-09-18
Expires:          2023-09-17
User ID:          AWS CLI Team <aws-cli@amazon.com>
Key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C

```

4. Impor kunci AWS CLI publik dengan perintah berikut, ganti *public-key-file-name* dengan nama file kunci publik yang Anda buat.

```

$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
imported
gpg: Total number processed: 1
gpg:             imported: 1

```



5. Unduh file AWS CLI tanda tangan untuk paket yang Anda unduh di <https://awscli.amazonaws.com/awscli.tar.gz.sig>. Ini memiliki jalur dan nama yang sama dengan file tarball yang sesuai dengannya, tetapi memiliki ekstensi .sig. Simpan di jalur yang sama dengan file tarball. Atau gunakan blok perintah berikut:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli.tar.gz.sig
```

6. Verifikasi tanda tangan, meneruskan nama yang diunduh .sig dan .zip file sebagai parameter ke gpg perintah.

```
$ gpg --verify awscliv2.sig awscli.tar.gz
```

Outputnya akan terlihat serupa dengan yang berikut ini:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672
475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:                There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672 475C
```

#### Important

Peringatan dalam output diharapkan dan tidak menunjukkan masalah. Itu terjadi karena tidak ada rantai kepercayaan antara kunci PGP pribadi Anda (jika Anda memilikinya) dan kunci AWS CLI PGP. Untuk informasi selengkapnya, lihat [Web kepercayaan](#).

2. Anda memiliki lingkungan yang dapat menjalankan file [GNU Autotools](#) yang dihasilkan seperti dan. configure Makefile File-file ini secara luas portabel di seluruh platform POSIX.

#### Linux and macOS

Jika Autotools belum diinstal di lingkungan Anda atau Anda perlu memperbaruinya, ikuti petunjuk instalasi yang ditemukan di [Bagaimana cara menginstal Autotools \(sebagai pengguna\)?](#) atau [Instalasi Dasar](#) dalam dokumentasi GNU.

## Windows PowerShell

### Warning

Kami menyarankan jika Anda berada di lingkungan Windows, Anda menggunakan installer pra-dibangun. Untuk petunjuk pemasangan pada installer yang sudah dibuat sebelumnya, lihat [the section called "Instal/Perbarui"](#)

Karena Windows tidak dilengkapi dengan shell yang sesuai dengan POSIX, Anda perlu menginstal perangkat lunak tambahan untuk menginstal dari AWS CLI sumber. [MSYS2](#) menyediakan kumpulan alat dan pustaka untuk membantu membangun dan menginstal perangkat lunak Windows, terutama untuk skrip berbasis POSIX yang digunakan Autotools.

1. Instal MSYS2. Untuk informasi tentang menginstal dan menggunakan MSYS2, lihat [petunjuk pemasangan dan penggunaan di Dokumentasi MSYS2](#).
2. Buka terminal MSYS2 dan instal autotools menggunakan perintah berikut.

```
$ pacman -S autotools
```

### Note

Saat menggunakan konfigurasi, bangun, dan instal contoh kode dalam panduan ini untuk Windows, jalur instalasi MSYS2 default diasumsikan. `C:\msys64\usr\bin\bash` Saat memanggil MSYS2 di dalam PowerShell Anda akan menggunakan format berikut, dengan perintah bash dalam tanda kutip:

```
PS C:\> C:\msys64\usr\bin\bash -lc "command example"
```

Contoh perintah berikut memanggil `./configure` perintah.

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure"
```

3. Sebuah interpreter Python 3.8 atau yang lebih baru diinstal. Versi Python minimum yang diperlukan mengikuti garis waktu yang sama dengan [kebijakan dukungan Python](#) resmi untuk SDK dan Tools. AWS Penerjemah hanya didukung 6 bulan setelah end-of-support tanggalnya.

4. (Opsional) Instal semua dependensi pustaka Python build dan runtime dari file. AWS CLI `./configure` Perintah memberi tahu Anda jika Anda kehilangan dependensi dan cara menginstalnya.

Anda dapat secara otomatis menginstal dan menggunakan dependensi ini melalui konfigurasi, lihat [the section called “Mengunduh dependensi”](#) untuk informasi lebih lanjut.

## Langkah 2: Mengkonfigurasi instalasi AWS CLI sumber

Konfigurasi untuk membangun dan menginstal AWS CLI ditentukan menggunakan `configure` skrip. Untuk dokumentasi semua opsi konfigurasi, jalankan `configure` skrip dengan `--help` opsi:

Linux and macOS

```
$ ./configure --help
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure --help"
```

Opsi yang paling penting adalah sebagai berikut:

- [Instal lokasi](#)
- [Penerjemah Python](#)
- [Mengunduh dependensi](#)
- [Tipe Instal](#)

### Instal lokasi

Instalasi sumber AWS CLI menggunakan dua direktori yang dapat dikonfigurasi untuk menginstal: AWS CLI

- `libdir`- Direktori induk tempat AWS CLI akan diinstal. Jalan menuju AWS CLI instalasi adalah `<libdir-value>/aws-cli`. `libdir` Nilai default untuk Linux dan macOS adalah `/usr/local/lib` membuat direktori instalasi default `/usr/local/lib/aws-cli`
- `bindir`- Direktori tempat AWS CLI executable diinstal. Lokasi default adalah `/usr/local/bin`.

configureOpsi berikut mengontrol direktori yang digunakan:

- `--prefix`- Menetapkan awalan direktori untuk digunakan untuk instalasi. Nilai default untuk Linux dan macOS adalah `/usr/local`
- `--libdir`- Mengatur `libdir` untuk digunakan untuk menginstal AWS CLI. Nilai default-nya adalah `<prefix-value>/lib`. Jika `--prefix` keduanya `--libdir` dan tidak ditentukan, default untuk Linux dan macOS adalah `/usr/local/lib/`
- `--bindir`- Menetapkan `bindir` untuk digunakan untuk menginstal AWS CLI `aws` dan `aws_completer` executable. Nilai default-nya adalah `<prefix-value>/bin`. Jika `--prefix` keduanya `bindir` dan tidak ditentukan, default untuk Linux dan macOS adalah `/usr/local/bin/`

## Linux and macOS

Contoh perintah berikut menggunakan `--prefix` opsi untuk melakukan instalasi pengguna lokal dari AWS CLI. Perintah ini menginstal AWS CLI in `$HOME/.local/lib/aws-cli` dan executable di: `$HOME/.local/bin`

```
$ ./configure --prefix=$HOME/.local
```

Contoh perintah berikut menggunakan `--libdir` opsi untuk menginstal AWS CLI sebagai aplikasi add-on di `/opt` direktori. Perintah ini menginstal AWS CLI at `/opt/aws-cli` dan executable di lokasi default mereka. `/usr/local/bin`

```
$ ./configure --libdir=/opt
```

## Windows PowerShell

Contoh perintah berikut menggunakan `--prefix` opsi untuk melakukan instalasi pengguna lokal dari AWS CLI. Perintah ini menginstal AWS CLI in `$HOME/.local/lib/aws-cli` dan executable di: `$HOME/.local/bin`

```
$ C:\msys64\usr\bin\bash -lc "./configure --prefix='C:\Program Files\AWSCLI'"
```

Contoh perintah berikut menggunakan `--libdir` opsi untuk menginstal AWS CLI sebagai aplikasi add-on di `/opt` direktori. Perintah ini menginstal AWS CLI at `C:\Program Files\AWSCLI\opt\aws-cli`.

## Penerjemah Python

### Note

Sangat disarankan untuk menentukan penerjemah Python saat menginstal untuk Windows.

`./configure` skrip secara otomatis memilih penerjemah Python 3.8 atau yang lebih baru yang diinstal untuk digunakan dalam membangun dan menjalankan AWS CLI menggunakan makro Autoconf. [AM\\_PATH\\_PYTHON](#)

Penerjemah Python yang akan digunakan dapat diatur secara eksplisit menggunakan variabel `PYTHON` lingkungan saat menjalankan skrip: `configure`

Linux and macOS

```
$ PYTHON=/path/to/python ./configure
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "PYTHON='C:\i/path/to/python' ./configure"
```

## Mengunduh dependensi

Secara default, semua dependensi build dan runtime diperlukan AWS CLI sudah diinstal pada sistem. Ini termasuk dependensi pustaka Python. Semua dependensi diperiksa saat `configure` skrip dijalankan, dan jika sistem kehilangan dependensi Python, skrip akan hilang. `configure`

Contoh kode berikut error saat sistem Anda kehilangan dependensi:

Linux and macOS

```
$ ./configure
checking for a Python interpreter with version >= 3.8... python
checking for python... /Users/username/.envs/env3.11/bin/python
checking for python version... 3.11
checking for python platform... darwin
checking for GNU default python prefix... ${prefix}
checking for GNU default python exec_prefix... ${exec_prefix}
```

```

checking for python script directory (pythondir)... ${PYTHON_PREFIX}/lib/python3.11/
site-packages
checking for python extension module directory (pyexecdir)... ${PYTHON_EXEC_PREFIX}/
lib/python3.11/site-packages
checking for --with-install-type... system-sandbox
checking for --with-download-deps... Traceback (most recent call last):
  File "<frozen runpy>", line 198, in _run_module_as_main
  File "<frozen runpy>", line 88, in _run_code
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
125, in <module>
    main()
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
121, in main
    parsed_args.func(parsed_args)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
49, in validate
    validate_env(parsed_args.artifact)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/validate_env.py",
line 68, in validate_env
    raise UnmetDependenciesException(unmet_deps, in_venv)
validate_env.UnmetDependenciesException: Environment requires following Python
dependencies:

awsCRT (required: ('>=0.12.4', '<0.17.0')) (version installed: None)

We recommend using --with-download-deps flag to automatically create a virtualenv
and download the dependencies.

If you want to manage the dependencies yourself instead, run the following pip
command:
/Users/username/.envs/env3.11/bin/python -m pip install --prefer-binary
'awsCRT>=0.12.4,<0.17.0'

configure: error: "Python dependencies not met."

```

## Windows PowerShell

```

PS C:\> C:\msys64\usr\bin\bash -lc "./configure"
checking for a Python interpreter with version >= 3.8... python
checking for python... /Users/username/.envs/env3.11/bin/python
checking for python version... 3.11
checking for python platform... darwin
checking for GNU default python prefix... ${prefix}

```

```

checking for GNU default python exec_prefix... ${exec_prefix}
checking for python script directory (pythondir)... ${PYTHON_PREFIX}/lib/python3.11/
site-packages
checking for python extension module directory (pyexecdir)... ${PYTHON_EXEC_PREFIX}/
lib/python3.11/site-packages
checking for --with-install-type... system-sandbox
checking for --with-download-deps... Traceback (most recent call last):
  File "<frozen runpy>", line 198, in _run_module_as_main
  File "<frozen runpy>", line 88, in _run_code
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
125, in <module>
    main()
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
121, in main
    parsed_args.func(parsed_args)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
49, in validate
    validate_env(parsed_args.artifact)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/validate_env.py",
line 68, in validate_env
    raise UnmetDependenciesException(unmet_deps, in_venv)
validate_env.UnmetDependenciesException: Environment requires following Python
dependencies:

awscrt (required: ('>=0.12.4', '<0.17.0')) (version installed: None)

We recommend using --with-download-deps flag to automatically create a virtualenv
and download the dependencies.

If you want to manage the dependencies yourself instead, run the following pip
command:
/Users/username/.envs/env3.11/bin/python -m pip install --prefer-binary
'awscrt>=0.12.4,<0.17.0'

configure: error: "Python dependencies not met."

```

Untuk menginstal dependensi Python yang diperlukan secara otomatis, gunakan opsi. `--with-download-deps` Saat menggunakan flag ini, proses build melakukan hal berikut:

- Melewati pemeriksaan dependensi pustaka Python.

- Mengonfigurasi pengaturan untuk mengunduh semua dependensi Python yang diperlukan dan hanya menggunakan dependensi yang diunduh untuk membangun selama pembuatan. AWS CLI `make`

Contoh perintah konfigurasi berikut menggunakan `--with-download-deps` opsi untuk mengunduh dan menggunakan dependensi Python:

Linux and macOS

```
$ ./configure --with-download-deps
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure --with-download-deps"
```

## Tipe Instal

Proses instalasi sumber mendukung jenis instalasi berikut:

- `system-sandbox-` (Default) Menciptakan lingkungan virtual Python yang terisolasi, menginstal AWS CLI ke dalam lingkungan virtual, dan symlink ke dan dapat `aws_completer` dieksekusi di `aws` lingkungan virtual. Instalasi ini AWS CLI bergantung langsung pada penerjemah Python yang dipilih untuk runtime-nya.

Ini adalah mekanisme instalasi ringan untuk menginstal sistem dan mengikuti praktik Python terbaik dengan melakukan sandboxing instalasi di lingkungan virtual. AWS CLI Instalasi ini ditujukan untuk pelanggan yang ingin menginstal AWS CLI from source dengan cara yang paling mudah dengan instalasi yang digabungkan dengan instalasi Python Anda.

- `portable-exe-` Membekukan AWS CLI ke dalam executable mandiri yang dapat didistribusikan ke lingkungan arsitektur serupa. Ini adalah proses yang sama yang digunakan untuk menghasilkan executable pra-built resmi dari AWS CLI `portable-exe` Pembekuan dalam salinan penerjemah Python yang dipilih dalam langkah `configure` yang akan digunakan untuk runtime. AWS CLI Hal ini memungkinkan untuk dipindahkan ke mesin lain yang mungkin tidak memiliki penerjemah Python.

Jenis build ini berguna karena Anda dapat memastikan AWS CLI instalasi Anda tidak digabungkan ke versi Python yang diinstal lingkungan dan Anda dapat mendistribusikan build ke sistem lain



yang mungkin belum menginstal Python. Ini memungkinkan Anda untuk mengontrol dependensi dan keamanan pada AWS CLI executable yang Anda gunakan.

Untuk mengkonfigurasi jenis instalasi, gunakan `--with-install-type` opsi dan tentukan nilai `portable-exe` atau `system-sandbox`.

Contoh `./configure` perintah berikut menentukan nilai: `portable-exe`

Linux and macOS

```
$ ./configure --with-install-type=portable-exe
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure --with-install-type=portable-exe"
```

## Langkah 3: Membangun AWS CLI

Gunakan `make` perintah untuk membangun AWS CLI menggunakan pengaturan konfigurasi Anda:

Linux and macOS

```
$ make
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "make"
```

### Note

Saat menggunakan `make` perintah, langkah-langkah berikut diselesaikan di belakang layar:

1. Lingkungan virtual dibuat di direktori build menggunakan modul Python [venv](#). Lingkungan virtual di-bootstrap dengan [versi pip yang dijual di pustaka standar Python](#).
2. Menyalin dependensi pustaka Python. Bergantung pada apakah `--with-download-deps` bendera ditentukan dalam `configure` perintah, langkah ini melakukan salah satu hal berikut:

- Yang `--with-download-deps` ditentukan. Dependensi Python diinstal pip. Ini termasuk `wheel`, `setuptools`, dan semua dependensi AWS CLI runtime. Jika Anda sedang membangun `portable-exe`, `pyinstaller` diinstal. Persyaratan ini semua ditentukan dalam file kunci yang dihasilkan dari [pip-compile](#).
  - `--with-download-deps` Tidak ditentukan. Pustaka Python dari paket situs penerjemah Python ditambah skrip apa pun (misalnya `pyinstaller`) disalin ke lingkungan virtual yang digunakan untuk build.
3. Berjalan `pip install` langsung di AWS CLI basis kode untuk melakukan pembuatan dan pemasangan offline di pohon AWS CLI ke dalam lingkungan virtual build. [Instalasi ini menggunakan bendera `pip --no-build-isolation`, `--use-feature=`, `-in-tree-build`, dan `no-cache-dir --no-index`](#)
  4. (Opsional) Jika `--install-type` diatur ke `portable-exe` dalam konfigurasi perintah, membangun executable mandiri menggunakan [pyinstaller](#)

## Langkah 4: Memasang AWS CLI

`make install` Perintah menginstal bawaan Anda AWS CLI ke lokasi yang dikonfigurasi pada sistem.

### Linux and macOS

Contoh perintah berikut menginstal AWS CLI menggunakan konfigurasi dan pengaturan build Anda:

```
$ make install
```

### Windows PowerShell

Contoh perintah berikut menginstal AWS CLI menggunakan konfigurasi dan pengaturan build Anda, lalu menambahkan variabel lingkungan dengan jalur untuk: AWS CLI

```
PS C:\> C:\msys64\usr\bin\bash -lc " make install "  
PS C:\> $Env: PATH +=";C:\Program Files\AWSCLI\bin\"
```

make installAturan mendukung [DESTDIR](#)variabel. Ketika ditentukan, variabel ini mengawali jalur yang ditentukan ke jalur instalasi yang sudah dikonfigurasi saat menginstal. AWS CLI Secara default, tidak ada nilai yang ditetapkan untuk variabel ini.

## Linux and macOS

Contoh kode berikut menggunakan `--prefix=/usr/local` bendera untuk mengkonfigurasi lokasi penginstalan, dan kemudian mengubah tujuan yang digunakan `DESTDIR=/tmp/stage` untuk perintah. `make install` Perintah-perintah ini menghasilkan AWS CLI yang diinstal di `/tmp/stage/usr/local/lib/aws-cli` dan executable yang terletak di `/tmp/stage/usr/local/bin`

```
$ ./configure --prefix=/usr/local
$ make
$ make DESTDIR=/tmp/stage install
```

## Windows PowerShell

Contoh kode berikut menggunakan `--prefix=\awscli` bendera untuk mengkonfigurasi lokasi penginstalan, dan kemudian mengubah tujuan yang digunakan `DESTDIR=C:\Program Files` untuk perintah. `make install` Perintah-perintah ini menghasilkan AWS CLI yang diinstal di `C:\Program Files\awscli`.

```
$ ./configure --prefix=\awscli
$ make
$ make DESTDIR='C:\Program Files' install
```

### Note

Saat berlari `make install`, langkah-langkah berikut diselesaikan di belakang layar

1. Memindahkan salah satu dari berikut ini ke direktori instalasi yang dikonfigurasi:
  - Jika jenis instalasinya `system-sandbox`, pindahkan lingkungan virtual bawaan Anda.
  - Jika tipe penginstalan adalah `portable-exe`, pindahkan executable mandiri bawaan Anda.

2. Membuat symlink untuk kedua `aws` dan `aws_completer` executable di direktori `bin` dikonfigurasi Anda.

## Langkah 5: Memverifikasi AWS CLI instalasi

Konfirmasikan AWS CLI berhasil diinstal dengan menggunakan perintah berikut:

```
$ aws --version
aws-cli/2.15.30 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
```

Jika `aws` perintah tidak dikenali, Anda mungkin perlu me-restart terminal Anda agar symlink baru dapat diperbarui. Jika Anda menemukan masalah tambahan setelah menginstal atau menghapus instalasi AWS CLI, lihat [Memecahkan masalah kesalahan](#) langkah-langkah pemecahan masalah umum

## Contoh alur kerja

Bagian ini memberikan beberapa contoh alur kerja dasar untuk menginstal dari sumber.

### Instalasi dasar Linux dan macOS

Contoh berikut adalah alur kerja instalasi dasar di AWS CLI mana diinstal di lokasi default. `/usr/local/lib/aws-cli`

```
$ cd path/to/cli/respository/
$ ./configure
$ make
$ make install
```

### Instalasi Windows otomatis

#### Note

Anda harus menjalankan PowerShell sebagai Administrator untuk menggunakan alur kerja ini.

MSYS2 dapat digunakan secara otomatis dalam pengaturan CI, lihat [Menggunakan MSYS2 di CI di Dokumentasi MSYS2](#).

## Downloaded Tarball

Unduh `awscli.tar.gz` file, ekstrak, dan instal file AWS CLI. Saat menggunakan perintah berikut, ganti jalur berikut:

- `C:\msys64\usr\bin\bash` dengan lokasi jalur MSYS2 Anda.
- `.\awscli-2.x.x` dengan nama `awscli.tar.gz` folder yang diekstraksi.
- `PYTHON='C:\path\to\python.exe'` dengan jalur Python lokal Anda.

Contoh kode berikut mengotomatiskan membangun dan menginstal AWS CLI from PowerShell menggunakan MSYS2, dan menentukan instalasi lokal Python mana yang akan digunakan:

```
PS C:\> curl -o awscli.tar.gz https://awscli.amazonaws.com/awscli.tar.gz #
  Download the awscli.tar.gz file in the current working directory
PS C:\> tar -xvzf .\awscli.tar.gz # Extract awscli.tar.gz file
PS C:\> cd .\awscli-2.x.x\ # Navigate to the root of the extracted files
PS C:\> $env:CHERE_INVOKING = 'yes' # Preserve the current working directory
PS C:\> C:\msys64\usr\bin\bash -lc " PYTHON='C:\path\to\python.exe' ./configure --
prefix='C:\Program Files\AWSCLI' --with-download-deps "
PS C:\> C:\msys64\usr\bin\bash -lc "make"
PS C:\> C:\msys64\usr\bin\bash -lc "make install"
PS C:\> $Env:PATH +=";C:\Program Files\AWSCLI\bin\"
PS C:\> aws --version
aws-cli/2.15.30 Python/3.11.6 Windows/10 source-sandbox/AMD64 prompt/off
```

## GitHub Repository

Unduh `awscli.tar.gz` file, ekstrak, dan instal file AWS CLI. Saat menggunakan perintah berikut, ganti jalur berikut:

- `C:\msys64\usr\bin\bash` dengan lokasi jalur MSYS2 Anda.
- `C:\path\to\cli\repository` dengan jalur ke [AWS CLI repositori](#) kloning Anda dari GitHub. Untuk informasi selengkapnya, lihat [Mem-fork repo di Docs](#) GitHub
- `PYTHON='C:\path\to\python.exe'` dengan jalur Python lokal Anda.

Contoh kode berikut mengotomatiskan membangun dan menginstal AWS CLI from PowerShell menggunakan MSYS2, dan menentukan instalasi lokal Python mana yang akan digunakan:

```
PS C:\> cd C:path\to\cli\repository\  
PS C:\> $env:CHERE_INVOKING = 'yes' # Preserve the current working directory  
PS C:\> C:\msys64\usr\bin\bash -lc " PYTHON='C:\path\to\python.exe' ./configure --  
prefix='C:\Program Files\AWSCLI' --with-download-deps "  
PS C:\> C:\msys64\usr\bin\bash -lc "make"  
PS C:\> C:\msys64\usr\bin\bash -lc "make install"  
PS C:\> $Env:PATH +=";C:\Program Files\AWSCLI\bin\  
PS C:\> aws --version
```

## Wadah Alpine Linux

Di bawah ini adalah contoh Dockerfile yang dapat digunakan untuk mendapatkan instalasi kerja AWS CLI dalam wadah Alpine Linux sebagai [alternatif binari pra-bangun untuk](#) Alpine. Saat menggunakan contoh ini, ganti `AWSCLI_VERSION` dengan nomor AWS CLI versi yang Anda inginkan:

```
FROM python:3.8-alpine AS builder  
  
ENV AWSCLI_VERSION=2.10.1  
  
RUN apk add --no-cache \  
    curl \  
    make \  
    cmake \  
    gcc \  
    g++ \  
    libc-dev \  
    libffi-dev \  
    openssl-dev \  
    && curl https://awscli.amazonaws.com/awscli-${AWSCLI_VERSION}.tar.gz | tar -xz \  
    && cd awscli-${AWSCLI_VERSION} \  
    && ./configure --prefix=/opt/aws-cli/ --with-download-deps \  
    && make \  
    && make install  
  
FROM python:3.8-alpine  
  
RUN apk --no-cache add groff
```

```
COPY --from=builder /opt/aws-cli/ /opt/aws-cli/  
  
ENTRYPOINT ["/opt/aws-cli/bin/aws"]
```

Gambar ini dibuat dan AWS CLI dipanggil dari wadah yang mirip dengan yang dibangun di Amazon Linux 2:

```
$ docker build --tag awscli-alpine .  
$ docker run --rm -it awscli-alpine --version  
aws-cli/2.2.1 Python/3.8.11 Linux/5.10.25-linuxkit source-sandbox/x86_64.alpine.3  
prompt/off
```

Ukuran akhir gambar ini lebih kecil dari ukuran gambar AWS CLI Docker resmi. Untuk informasi tentang gambar Docker resmi, lihat [the section called “Amazon ECR Publik/Docker”](#).

## Memecahkan masalah kesalahan AWS CLI pemasangan dan hapus instalasi

Untuk langkah pemecahan masalah kesalahan penginstalan, lihat [Memecahkan masalah kesalahan langkah-langkah pemecahan masalah umum](#). Untuk langkah pemecahan masalah yang paling relevan, lihat, [the section called “Perintah tidak ditemukan kesalahan”](#) [the section called “Perintah aws --version”](#) [mengembalikan versi yang berbeda dari yang Anda instal](#), dan [the section called “Perintah aws --version”](#) [mengembalikan versi setelah mencopot pemasangan AWS CLI](#)

Untuk masalah apa pun yang tidak tercakup dalam panduan pemecahan masalah, cari masalah dengan source-distribution label di [AWS CLI Repositori](#) aktif. GitHub Jika tidak ada masalah yang menutupi kesalahan Anda, [buat masalah baru](#) untuk menerima bantuan dari AWS CLI pengelola.

## Langkah selanjutnya

Setelah menginstal AWS CLI, Anda harus melakukan [the section called “Pengaturan”](#).

## Jalankan AWS CLI dari gambar Amazon ECR Public atau Docker resmi

Topik ini menjelaskan cara menjalankan, mengontrol versi, dan mengonfigurasi AWS CLI versi 2 di Docker menggunakan image resmi Amazon Elastic Container Registry Public (Amazon ECR

Public) atau Docker Hub. Untuk informasi selengkapnya tentang cara menggunakan Docker, lihat [dokumentasi Docker](#).

Gambar resmi menyediakan isolasi, portabilitas, dan keamanan yang AWS secara langsung mendukung dan memelihara. Ini memungkinkan Anda untuk menggunakan AWS CLI versi 2 di lingkungan berbasis wadah tanpa harus mengelola instalasi sendiri.

## Topik

- [Prasyarat](#)
- [Memutuskan antara Amazon ECR Public dan Docker Hub](#)
- [Jalankan gambar AWS CLI versi resmi 2](#)
- [Catatan tentang antarmuka dan kompatibilitas mundur dari gambar resmi](#)
- [Gunakan versi dan tag tertentu](#)
- [Perbarui ke gambar resmi terbaru](#)
- [Bagikan file host, kredensial, variabel lingkungan, dan konfigurasi](#)
- [Mempersingkat perintah docker run](#)

## Prasyarat

Anda harus menginstal Docker. Untuk petunjuk penginstalan, lihat [situs web Docker](#).

Untuk memverifikasi instalasi Docker Anda, jalankan perintah berikut dan konfirmasikan ada output.

```
$ docker --version
Docker version 19.03.1
```

## Memutuskan antara Amazon ECR Public dan Docker Hub

Sebaiknya gunakan Amazon ECR Public melalui Docker Hub untuk AWS CLI gambar. Docker Hub memiliki pembatasan tarif yang lebih ketat untuk konsumen publik yang dapat menyebabkan masalah pelambatan. Selain itu, Amazon ECR Public mereplikasi gambar di lebih dari satu wilayah untuk memberikan ketersediaan yang kuat dan menangani masalah pemadaman wilayah.

Untuk informasi selengkapnya tentang pembatasan laju Docker Hub, lihat [Memahami Pembatasan Tingkat Hub Docker di situs web Docker](#).



## Jalankan gambar AWS CLI versi resmi 2

Pertama kali Anda menggunakan `docker run` perintah, gambar terbaru diunduh ke komputer Anda. Setiap penggunaan `docker run` perintah selanjutnya berjalan dari salinan lokal Anda.

Untuk menjalankan gambar Docker AWS CLI versi 2, gunakan `docker run` perintah.

### Amazon ECR Public

AWS CLI [Versi resmi 2 Amazon ECR Public image di-host di Amazon ECR Public di repositori. `aws-cli/aws-cli`](#)

```
$ docker run --rm -it public.ecr.aws/aws-cli/aws-cli command
```

### Docker Hub

Gambar Docker AWS CLI versi 2 resmi di-host di Docker Hub di repositori. `amazon/aws-cli`

```
$ docker run --rm -it amazon/aws-cli command
```

Beginilah fungsi perintah:

- `docker run --rm -it repository/name`— Setara dengan yang dapat `aws` dieksekusi. Setiap kali Anda menjalankan perintah ini, Docker memutar wadah gambar yang Anda unduh, dan menjalankan perintah `aws`. Secara default, gambar menggunakan versi terbaru dari AWS CLI versi 2.

Misalnya, untuk memanggil `aws --version` perintah di Docker, Anda menjalankan yang berikut ini.

### Amazon ECR Public

```
$ docker run --rm -it public.ecr.aws/aws-cli/aws-cli --version  
aws-cli/2.15.30 Python/3.7.3 Linux/4.9.184-linuxkit botocore/2.4.5dev10
```

### Docker Hub

```
$ docker run --rm -it amazon/aws-cli --version  
aws-cli/2.15.30 Python/3.7.3 Linux/4.9.184-linuxkit botocore/2.4.5dev10
```

- `--rm`- Menentukan untuk membersihkan wadah setelah perintah keluar.
- `-it`- Menentukan untuk membuka pseudo-TTY dengan `stdin`. Ini memungkinkan Anda untuk memberikan masukan ke AWS CLI versi 2 saat berjalan dalam wadah, misalnya, dengan menggunakan `aws help` perintah `aws configure and`. Saat memilih apakah akan dihilangkan `-it`, pertimbangkan hal berikut:
  - Jika Anda menjalankan skrip, tidak `-it` diperlukan.
  - Jika Anda mengalami kesalahan dengan skrip Anda, menghilangkan `-it` panggilan Docker Anda mungkin memperbaiki masalah.
  - Jika Anda mencoba menyalurkan output, `-it` dapat menyebabkan kesalahan dan menghilangkan `-it` panggilan Docker Anda dapat menyelesaikan masalah ini. Jika Anda ingin menyimpan `-it` flag, tetapi masih ingin menyalurkan output, menonaktifkan [pager sisi klien](#) yang AWS CLI digunakan secara default akan menyelesaikan masalah.

Untuk informasi selengkapnya tentang `docker run` perintah, lihat [panduan referensi Docker](#).

## Catatan tentang antarmuka dan kompatibilitas mundur dari gambar resmi

- Satu-satunya alat yang didukung pada gambar adalah AWS CLI. Hanya `aws executable` yang harus langsung dijalankan. Misalnya, meskipun `less` dan `groff` secara eksplisit diinstal pada gambar, mereka tidak boleh dieksekusi langsung di luar AWS CLI perintah.
- Direktori `/aws` kerja dikendalikan pengguna. Gambar tidak akan menulis ke direktori ini, kecuali diinstruksikan oleh pengguna dalam menjalankan AWS CLI perintah.
- Tidak ada jaminan kompatibilitas mundur dalam mengandalkan tag terbaru. Untuk menjamin kompatibilitas mundur, Anda harus menyematkan ke `<major.minor.patch>` tag tertentu karena tag tersebut tidak dapat diubah; mereka hanya akan didorong satu kali.

## Gunakan versi dan tag tertentu

Gambar resmi AWS CLI versi 2 memiliki beberapa versi yang dapat Anda gunakan, dimulai dengan `versi2.0.6`. Untuk menjalankan versi tertentu dari AWS CLI versi 2, tambahkan tag yang sesuai ke `docker run` perintah Anda. Pertama kali Anda menggunakan `docker run` perintah dengan tag, gambar terbaru untuk tag itu diunduh ke komputer Anda. Setiap penggunaan `docker run` perintah berikutnya dengan tag tersebut berjalan dari salinan lokal Anda.

Anda dapat menggunakan dua jenis tag:

- `latest`— Mendefinisikan versi terbaru dari AWS CLI versi 2 untuk gambar. Kami menyarankan Anda menggunakan `latest` tag ketika Anda menginginkan versi terbaru dari AWS CLI versi 2. Namun, tidak ada jaminan kompatibilitas mundur saat mengandalkan tag ini. `latestTag` digunakan secara default dalam `docker run` perintah. Untuk menggunakan tag secara eksplisit, tambahkan `latest` tag ke nama gambar kontainer.

Amazon ECR Public

```
$ docker run --rm -it public.ecr.aws/aws-cli/aws-cli:latest command
```

Docker Hub

```
$ docker run --rm -it amazon/aws-cli:latest command
```

- `<major.minor.patch>`— Mendefinisikan versi tertentu dari AWS CLI versi 2 untuk gambar. Jika Anda berencana untuk menggunakan gambar resmi dalam produksi, kami sarankan Anda menggunakan versi tertentu dari AWS CLI versi 2 untuk memastikan kompatibilitas mundur. Misalnya, untuk menjalankan versi `2.0.6`, tambahkan versi ke nama gambar kontainer.

Amazon ECR Public

```
$ docker run --rm -it public.ecr.aws/aws-cli/aws-cli:2.0.6 command
```

Docker Hub

```
$ docker run --rm -it amazon/aws-cli:2.0.6 command
```

## Perbarui ke gambar resmi terbaru

Karena gambar terbaru diunduh ke komputer Anda hanya saat pertama kali Anda menggunakan `docker run` perintah, Anda perlu menarik gambar yang diperbarui secara manual. Untuk memperbarui secara manual ke versi terbaru, kami sarankan Anda menarik gambar yang `latest` ditandai. Menarik gambar mengunduh versi terbaru ke komputer Anda.

Amazon ECR Public

```
$ docker pull public.ecr.aws/aws-cli/aws-cli:latest
```

## Docker Hub

```
$ docker pull amazon/aws-cli:latest
```

## Bagikan file host, kredensyal, variabel lingkungan, dan konfigurasi

Karena AWS CLI versi 2 dijalankan dalam wadah, secara default CLI tidak dapat mengakses sistem file host, yang mencakup konfigurasi dan kredensyal. Untuk berbagi sistem file host, kredensyal, dan konfigurasi ke wadah, pasang ~/.aws direktori sistem host ke wadah di /root/.aws dengan -v tanda ke perintah. `docker run` ini memungkinkan AWS CLI versi 2 yang berjalan di wadah untuk menemukan informasi file host.

### Amazon ECR Public

#### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws public.ecr.aws/aws-cli/aws-cli command
```

#### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws public.ecr.aws/aws-cli/aws-cli command
```

#### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws public.ecr.aws/aws-cli/aws-cli command
```

## Docker Hub

#### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws amazon/aws-cli command
```

#### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws amazon/aws-cli command
```

#### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws amazon/aws-cli command
```

Untuk informasi selengkapnya tentang `-v` flag dan pemasangan, lihat [panduan referensi Docker](#).

#### Note

Untuk informasi tentang config dan credentials file, lihat [the section called “Konfigurasi dan pengaturan file kredensi”](#).

## Contoh 1: Memberikan kredensyal dan konfigurasi

Dalam contoh ini, kami menyediakan kredensyal dan konfigurasi host saat menjalankan `s3 ls` perintah untuk mencantumkan bucket Anda di Amazon Simple Storage Service (Amazon S3). Contoh di bawah ini menggunakan lokasi default untuk AWS CLI kredensyal dan file konfigurasi, untuk menggunakan lokasi yang berbeda, ubah jalur file.

### Amazon ECR Public

#### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws public.ecr.aws/aws-cli/aws-cli s3 ls  
2020-03-25 00:30:48 aws-cli-docker-demo
```

#### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%.aws:/root/.aws public.ecr.aws/aws-cli/aws-  
cli s3 ls  
2020-03-25 00:30:48 aws-cli-docker-demo
```

#### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws public.ecr.aws/aws-cli/  
aws-cli s3 ls
```

### Docker Hub

#### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

## Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%\aws:/root/.aws amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

## Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\aws:/root/.aws amazon/aws-cli s3 ls
```

Anda dapat memanggil variabel lingkungan sistem tertentu menggunakan `-e` bendera. Untuk menggunakan variabel lingkungan, panggil dengan nama.

## Amazon ECR Public

### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e ENVVAR_NAME public.ecr.aws/aws-cli/
aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

## Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%\aws:/root/.aws -e ENVVAR_NAME
public.ecr.aws/aws-cli/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

## Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\aws:/root/.aws -e ENVVAR_NAME
public.ecr.aws/aws-cli/aws-cli s3 ls
```

## Docker Hub

### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e ENVVAR_NAME amazon/aws-cli s3 ls
```

```
2020-03-25 00:30:48 aws-cli-docker-demo
```

## Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%\aws:/root/.aws -e ENVVAR_NAME amazon/aws-cli  
s3 ls  
2020-03-25 00:30:48 aws-cli-docker-demo
```

## Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\aws:/root/.aws -e ENVVAR_NAME amazon/  
aws-cli s3 ls
```

## Contoh 2: Mengunduh file Amazon S3 ke sistem host Anda

Untuk beberapa perintah AWS CLI versi 2, Anda dapat membaca file dari sistem host di wadah atau menulis file dari wadah ke sistem host.

Dalam contoh ini, kita men-download S3 objek `s3://aws-cli-docker-demo/hello` ke sistem file lokal Anda dengan memasang direktori kerja saat ini ke `/aws` direktori container. Dengan mengunduh `hello` objek ke `/aws` direktori kontainer, file disimpan ke direktori kerja sistem host saat ini juga.

## Amazon ECR Public

### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws public.ecr.aws/aws-cli/  
aws-cli s3 cp s3://aws-cli-docker-demo/hello .  
download: s3://aws-cli-docker-demo/hello to ./hello
```

### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%\aws:/root/.aws -v %cd%:/aws public.ecr.aws/  
aws-cli/aws-cli s3 cp s3://aws-cli-docker-demo/hello .  
download: s3://aws-cli-docker-demo/hello to ./hello
```

### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v $pwd\aws:/aws
public.ecr.aws/aws-cli/aws-cli s3 cp s3://aws-cli-docker-demo/hello .
```

## Docker Hub

### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws amazon/aws-cli s3 cp s3://
aws-cli-docker-demo/hello .
download: s3://aws-cli-docker-demo/hello to ./hello
```

### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%.aws:/root/.aws -v %cd%:/aws amazon/aws-cli
s3 cp s3://aws-cli-docker-demo/hello .
download: s3://aws-cli-docker-demo/hello to ./hello
```

### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v $pwd\aws:/aws
amazon/aws-cli s3 cp s3://aws-cli-docker-demo/hello .
```

Untuk mengonfirmasi file yang diunduh ada di sistem file lokal, jalankan yang berikut ini.

### Linux dan macOS

```
$ cat hello
Hello from Docker!
```

### Jendela PowerShell

```
$ type hello
Hello from Docker!
```

## Contoh 3: Menggunakan variabel lingkungan AWS\_PROFILE Anda

Anda dapat memanggil variabel lingkungan sistem tertentu menggunakan `-e` bendera. Panggil setiap variabel lingkungan yang ingin Anda gunakan. Dalam contoh ini, kami menyediakan kredensial



host, konfigurasi, dan variabel lingkungan *AWS\_PROFILE* saat menjalankan `s3 ls` perintah untuk mencantumkan bucket Anda di Amazon Simple Storage Service (Amazon S3).

## Amazon ECR Public

### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e AWS_PROFILE public.ecr.aws/aws-cli/
aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%.aws:/root/.aws -e AWS_PROFILE
public.ecr.aws/aws-cli/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -e AWS_PROFILE
public.ecr.aws/aws-cli/aws-cli s3 ls
```

## Docker Hub

### Linux dan macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e AWS_PROFILE amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

### Prompt Perintah Windows

```
$ docker run --rm -it -v %userprofile%.aws:/root/.aws -e AWS_PROFILE amazon/aws-cli
s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

### Jendela PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -e AWS_PROFILE amazon/
aws-cli s3 ls
```

## Mempersingkat perintah docker run

Untuk mempersingkat `docker run` perintah, kami sarankan Anda menggunakan kemampuan sistem operasi Anda untuk membuat [symbolic link](#) (symlink) atau [alias](#) di Linux dan macOS, atau di Windows. [doskey](#) Untuk mengatur aws alias, Anda dapat menjalankan salah satu perintah berikut.

- Untuk akses dasar ke aws perintah, jalankan yang berikut ini.

### Amazon ECR Public

#### Linux dan macOS

```
$ alias aws='docker run --rm -it public.ecr.aws/aws-cli/aws-cli'
```

#### Prompt Perintah Windows

```
C:\> doskey aws=docker run --rm -it public.ecr.aws/aws-cli/aws-cli $*
```

#### Jendela PowerShell

```
C:\> Function AWSCLI {docker run --rm -it public.ecr.aws/aws-cli/aws-cli $args}  
Set-Alias -Name aws -Value AWSCLI
```

### Docker Hub

#### Linux dan macOS

```
$ alias aws='docker run --rm -it amazon/aws-cli'
```

#### Prompt Perintah Windows

```
C:\> doskey aws=docker run --rm -it amazon/aws-cli $*
```

#### Jendela PowerShell

```
C:\> Function AWSCLI {docker run --rm -it amazon/aws-cli $args}  
Set-Alias -Name aws -Value AWSCLI
```

- Untuk akses ke sistem file host dan pengaturan konfigurasi saat menggunakan aws perintah, jalankan yang berikut ini.

## Amazon ECR Public

### Linux dan macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws  
public.ecr.aws/aws-cli/aws-cli'
```

### Prompt Perintah Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%.aws:/root/.aws -v %cd%:/aws  
public.ecr.aws/aws-cli/aws-cli $*
```

### Jendela PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v  
$pwd\aws:/aws public.ecr.aws/aws-cli/aws-cli $args}  
Set-Alias -Name aws -Value AWSCLI
```

## Docker Hub

### Linux dan macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws amazon/aws-  
cli'
```

### Prompt Perintah Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%.aws:/root/.aws -v %cd%:/aws  
amazon/aws-cli $*
```

### Jendela PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v  
$pwd\aws:/aws amazon/aws-cli $args}  
Set-Alias -Name aws -Value AWSCLI
```

- Untuk menetapkan versi tertentu untuk digunakan dalam aws alias Anda, tambahkan tag versi Anda.

## Amazon ECR Public

### Linux dan macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws  
public.ecr.aws/aws-cli/aws-cli:2.0.6'
```

### Prompt Perintah Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%.aws:/root/.aws -v %cd%:/aws  
public.ecr.aws/aws-cli/aws-cli:2.0.6 $*
```

### Jendela PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v  
$pwd\aws:/aws public.ecr.aws/aws-cli/aws-cli:2.0.6 $args}  
Set-Alias -Name aws -Value AWSCLI
```

## Docker Hub

### Linux dan macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws amazon/aws-  
cli:2.0.6'
```

### Prompt Perintah Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%.aws:/root/.aws -v %cd%:/aws  
amazon/aws-cli:2.0.6 $*
```

### Jendela PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v  
$pwd\aws:/aws amazon/aws-cli:2.0.6 $args}  
Set-Alias -Name aws -Value AWSCLI
```

Setelah mengatur alias Anda, Anda dapat menjalankan AWS CLI versi 2 dari dalam wadah seolah-olah itu diinstal pada sistem host Anda.

```
$ aws --version
aws-cli/2.15.30 Python/3.7.3 Linux/4.9.184-linuxkit botocore/2.4.5dev10
```

## Mengatur AWS CLI

Topik ini menjelaskan cara mengonfigurasi setelan dasar yang digunakan AWS Command Line Interface (AWS CLI) untuk berinteraksi dengan cepat AWS. Ini termasuk kredensial keamanan Anda, format output default, dan Wilayah default AWS .

Topik

- [Kumpulkan informasi kredensi Anda untuk akses terprogram](#)
- [Menyiapkan konfigurasi dan kredensial baru](#)
- [Menggunakan file konfigurasi dan kredensial yang ada](#)

## Kumpulkan informasi kredensi Anda untuk akses terprogram

Anda akan memerlukan akses terprogram jika Anda ingin berinteraksi dengan AWS luar. AWS Management Console Untuk instruksi otentikasi dan kredensi, pilih salah satu opsi berikut:

Jenis otentikasi	Tujuan	Petunjuk
Kredensi jangka pendek pengguna IAM Identity Center tenaga kerja	(Disarankan) Gunakan kredensi jangka pendek untuk pengguna tenaga kerja IAM Identity Center.  Praktik keamanan terbaik adalah AWS Organizations dengan menggunakan IAM Identity Center. Ini menggabungkan kredensi jangka pendek dengan direktori pengguna, seperti direktori Pusat	<a href="#">the section called “Autentikasi Pusat Identitas IAM”</a>

Jenis otentikasi	Tujuan	Petunjuk
	Identitas IAM bawaan atau Direktori Aktif.	
Kredensi jangka pendek pengguna IAM	Gunakan kredensi jangka pendek pengguna IAM, yang lebih aman daripada kredensi jangka panjang. Jika kredensi Anda dikompromikan, ada waktu terbatas mereka dapat digunakan sebelum kedaluwarsa.	<a href="#">the section called “Kredensi jangka pendek”</a>
Pengguna IAM atau IAM Identity Center pada instans Amazon EC2.	Gunakan metadata instans Amazon EC2 untuk menanyakan kredensial sementara menggunakan peran yang ditetapkan ke instans Amazon EC2.	<a href="#">the section called “Menggunakan kredensial untuk metadata instans Amazon EC2”</a>
Asumsikan peran untuk izin	Pasangkan metode kredensi lain dan ambil peran untuk akses sementara ke pengguna Layanan AWS Anda mungkin tidak dapat mengaksesnya.	<a href="#">the section called “Peran IAM”</a>
Kredensi jangka panjang pengguna IAM	(Tidak disarankan) Gunakan kredensial jangka panjang, yang tidak memiliki kedaluwarsa.	<a href="#">the section called “Pengguna IAM”</a>
Penyimpanan eksternal pengguna tenaga kerja IAM atau IAM Identity Center	(Tidak disarankan) Pasangkan metode kredensi lain tetapi simpan nilai kredensi di lokasi di luar. AWS CLI Metode ini hanya seaman lokasi eksternal kredensialnya disimpan.	<a href="#">the section called “Kredensial eksternal”</a>

## Menyiapkan konfigurasi dan kredensial baru

AWS CLI Menyimpan konfigurasi dan informasi kredensi Anda dalam profil (kumpulan pengaturan) di `credentials` dan `config` file.

Terutama ada dua metode untuk mendapatkan pengaturan dengan cepat:

- [Mengkonfigurasi menggunakan perintah AWS CLI](#)
- [Mengedit kredensi dan file konfigurasi secara manual](#)

Contoh berikut menggunakan nilai sampel untuk masing-masing metode otentikasi. Ganti nilai sampel dengan nilai Anda sendiri.

### Mengkonfigurasi menggunakan perintah AWS CLI

Untuk penggunaan umum, `aws configure sso` perintah `aws configure` atau di terminal pilihan Anda adalah cara tercepat untuk mengatur AWS CLI instalasi Anda. Berdasarkan metode kredensi yang Anda inginkan, AWS CLI meminta Anda untuk informasi yang relevan. Secara default, informasi dalam profil ini digunakan saat Anda menjalankan AWS CLI perintah yang tidak secara eksplisit menentukan profil yang akan digunakan.

Untuk informasi lebih lanjut tentang `config` file `credentials` dan, lihat [Konfigurasi dan pengaturan file kredensi](#).

### IAM Identity Center (SSO)

Contoh ini untuk AWS IAM Identity Center menggunakan `aws configure sso wizard`. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi penyegaran token otomatis”](#).

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

Attempting to automatically open the SSO authorization page in your default browser.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
   ProductionAccount, production-account-admin@example.com (444455556666)

Using the account ID 111122223333
```

```
There are 2 roles available to you.
> ReadOnly
  FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```

## IAM Identity Center (Legacy SSO)

Contoh ini untuk metode lama AWS IAM Identity Center menggunakan `aws configure sso wizard`. Untuk menggunakan SSO lama, biarkan nama sesi kosong. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi lawas yang tidak dapat disegarkan”](#).

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
  ProductionAccount, production-account-admin@example.com (444455556666)

Using the account ID 111122223333

There are 2 roles available to you.
> ReadOnly
  FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```





```
$ aws configure set output json
```

## Long-term credentials

### Warning

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti [AWS IAM Identity Center](#).

Contoh ini untuk kredensi jangka panjang dari AWS Identity and Access Management Untuk informasi selengkapnya, lihat [the section called “Pengguna IAM”](#).

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

Untuk informasi lebih rinci tentang otentikasi dan metode kredensi lihat. [Otentikasi dan akses kredensial](#)

## Mengedit kredensi dan file konfigurasi secara manual

Saat menyalin dan menempelkan informasi, kami sarankan untuk mengedit `credentials` file `config` dan secara manual. Berdasarkan metode kredensi yang Anda inginkan, file diatur dengan cara yang berbeda.

File disimpan di direktori home Anda di bawah `.aws` folder. Di mana Anda menemukan lokasi direktori home Anda bervariasi berdasarkan sistem operasi, tetapi disebut menggunakan variabel lingkungan `%UserProfile%` di Windows dan `$HOME` atau `~` (tilde) dalam sistem berbasis Unix. Untuk informasi selengkapnya tentang tempat pengaturan ini disimpan, lihat [the section called “Di mana pengaturan konfigurasi disimpan?”](#).

Contoh berikut menunjukkan default profil dan profil bernama `user1` dan menggunakan nilai sampel. Ganti nilai sampel dengan nilai Anda sendiri. Untuk informasi lebih lanjut tentang `config` file `credentials` dan, lihat [Konfigurasi dan pengaturan file kredensi](#).

## IAM Identity Center (SSO)

Contoh ini untuk AWS IAM Identity Center. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi penyegaran token otomatis”](#).

### File kredensial

`credentialsFile` tidak digunakan untuk metode otentikasi ini.

### Berkas Config

```
[default]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text

[profile user1]
sso_session = my-sso
sso_account_id = 444455556666
sso_role_name = readOnly
region = us-east-1
output = json

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

## IAM Identity Center (Legacy SSO)

Contoh ini untuk metode warisan. AWS IAM Identity Center Untuk informasi selengkapnya, lihat [the section called “Konfigurasi lawas yang tidak dapat disegarkan”](#).

### File kredensial

`credentialsFile` tidak digunakan untuk metode otentikasi ini.

### Berkas Config

```
[default]
sso_start_url = https://my-sso-portal.awsapps.com/start
```





```
[default]
role_arn=arn:aws:iam::123456789012:role/defaultrole
credential_source=Ec2InstanceMetadata
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
credential_source=Ec2InstanceMetadata
region=us-east-1
output=text
```

## Long-term credentials

### Warning

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti [AWS IAM Identity Center](#).

Contoh ini untuk kredensi jangka panjang dari AWS Identity and Access Management Untuk informasi selengkapnya, lihat [the section called “Pengguna IAM”](#).

## File kredensial

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY

[user1]
aws_access_key_id=AKIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

## Berkas Config

```
[default]
region=us-west-2
output=json

[profile user1]
```

```
region=us-east-1  
output=text
```

Untuk informasi lebih rinci tentang otentikasi dan metode kredensi lihat. [Otentikasi dan akses kredensial](#)

## Menggunakan file konfigurasi dan kredensial yang ada

Jika Anda memiliki file konfigurasi dan kredensial yang ada, ini dapat digunakan untuk file. AWS CLI

Untuk menggunakan `credentials` file config dan, pindahkan ke folder bernama `.aws` di direktori home Anda. Di mana Anda menemukan lokasi direktori home Anda bervariasi berdasarkan sistem operasi, tetapi disebut menggunakan variabel lingkungan `%UserProfile%` di Windows dan `$HOME` atau `~` (tilde) dalam sistem berbasis Unix.

Anda dapat menentukan lokasi non-default untuk `credentials` file config dan dengan mengatur variabel `AWS_CONFIG_FILE` dan `AWS_SHARED_CREDENTIALS_FILE` lingkungan ke jalur lokal lain. Lihat [Variabel lingkungan untuk mengkonfigurasi AWS CLI](#) untuk detail.

Untuk informasi lebih rinci tentang konfigurasi dan file kredensial, lihat. [the section called "Konfigurasi dan pengaturan file kredensi"](#)

# Konfigurasi AWS CLI

Bagian ini menjelaskan cara mengonfigurasi pengaturan yang digunakan AWS Command Line Interface (AWS CLI) untuk berinteraksi AWS. Sumber daya yang dimaksud meliputi:

- Kredensial mengidentifikasi siapa yang memanggil API. Kredensial akses digunakan untuk mengenkripsi permintaan ke AWS server untuk mengonfirmasi identitas Anda dan mengambil kebijakan izin terkait. Izin ini menentukan tindakan yang dapat Anda lakukan. Untuk informasi tentang pengaturan kredensial Anda, lihat. [Otentikasi dan akses kredensial](#)
- Detail konfigurasi lainnya untuk memberi tahu AWS CLI cara memproses permintaan, seperti format output default dan AWS Wilayah default.

## Note

AWS mengharuskan semua permintaan yang masuk ditandatangani secara kriptografis. Yang AWS CLI melakukan ini untuk Anda. “Tanda tangan” termasuk stempel tanggal/waktu. Oleh karena itu, Anda harus memastikan bahwa tanggal dan waktu komputer Anda diatur dengan benar. Jika tidak, dan tanggal/waktu dalam tanda tangan terlalu jauh dari tanggal/waktu yang diakui oleh AWS layanan, AWS tolak permintaan tersebut.

## Konfigurasi dan kredensialnya diutamakan

Kredensial dan pengaturan konfigurasi terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dinyatakan pada baris perintah sebagai parameter. Lokasi tertentu lebih diutamakan daripada yang lain. Pengaturan AWS CLI kredensial dan konfigurasi diutamakan dalam urutan berikut:

1. [Opsis baris perintah](#) — Mengganti pengaturan di lokasi lain, seperti, `--region--output`, dan `--profile` parameter.
2. [Variabel lingkungan](#) — Anda dapat menyimpan nilai dalam variabel lingkungan sistem Anda.
3. [Asumsikan peran](#) - Asumsikan izin peran IAM melalui konfigurasi atau perintah. [aws sts assume-role](#)
4. [Asumsikan peran dengan identitas web](#) - Asumsikan izin peran IAM menggunakan identitas web melalui konfigurasi atau perintah. [aws sts assume-role](#)



5. [AWS IAM Identity Center](#)— Pengaturan konfigurasi Pusat Identitas IAM yang disimpan dalam config file diperbarui saat Anda menjalankan `aws configure sso` perintah. Kredensial kemudian diautentikasi ketika Anda menjalankan perintah. `aws sso login` configFile ini terletak `~/.aws/config` di Linux atau macOS, atau `C:\Users\USERNAME\.aws\config` di Windows.
6. [File kredensial](#) — config File credentials dan diperbarui saat Anda menjalankan perintah. `aws configure credentials`File ini terletak `~/.aws/credentials` di Linux atau macOS, atau `C:\Users\USERNAME\.aws\credentials` di Windows.
7. [Proses kustom](#) - Dapatkan kredensial Anda dari sumber eksternal.
8. [File konfigurasi](#) — config File credentials dan diperbarui saat Anda menjalankan perintah `aws configure`. configFile ini terletak `~/.aws/config` di Linux atau macOS, atau `C:\Users\USERNAME\.aws\config` di Windows.
9. [Kredensi kontainer](#) - Anda dapat mengaitkan peran IAM dengan masing-masing definisi tugas Amazon Elastic Container Service (Amazon ECS). Kredensi sementara untuk peran itu kemudian tersedia untuk wadah tugas itu. Untuk informasi lebih lanjut, lihat [Peran IAM untuk Tugas](#) dalam Panduan Developer Amazon Elastic Container Service.
10. [Kredensial profil instans Amazon EC2](#) — Anda dapat mengaitkan peran IAM dengan setiap instans Amazon Elastic Compute Cloud (Amazon EC2). Kredensi sementara untuk peran itu kemudian tersedia untuk kode yang berjalan dalam instance. Kredensialnya dikirimkan melalui layanan metadata Amazon EC2. Untuk informasi selengkapnya, lihat [Peran IAM untuk Amazon EC2](#) di Panduan Pengguna Amazon EC2 [dan Menggunakan Profil Instance](#) di Panduan Pengguna IAM.

## Topik tambahan di bagian ini

- [the section called “Konfigurasi dan pengaturan file kredensi”](#)
- [the section called “Variabel lingkungan”](#)
- [the section called “Opsis baris perintah”](#)
- [the section called “Penyelesaian perintah”](#)
- [the section called “Percobaan ulang”](#)
- [the section called “Gunakan proxy HTTP”](#)

## Konfigurasi dan pengaturan file kredensi

Anda dapat menyimpan pengaturan konfigurasi dan kredensial yang sering digunakan dalam file yang dikelola oleh file. AWS CLI

File-file tersebut dibagi menjadi `profiles`. Secara default, AWS CLI menggunakan pengaturan yang ditemukan di profil bernama `default`. Untuk menggunakan pengaturan alternatif, Anda dapat membuat dan mereferensikan profil tambahan.

Anda dapat mengganti pengaturan individual dengan menyetel salah satu variabel lingkungan yang didukung, atau dengan menggunakan parameter baris perintah. Untuk informasi selengkapnya tentang prioritas pengaturan konfigurasi, lihat. [Konfigurasi AWS CLI](#)

### Note

Untuk informasi tentang pengaturan kredensial Anda, lihat. [Otentikasi dan akses kredensial](#)

### Topik

- [Format konfigurasi dan file kredensial](#)
- [Di mana pengaturan konfigurasi disimpan?](#)
- [Menggunakan profil bernama](#)
- [Mengatur dan melihat pengaturan konfigurasi menggunakan perintah](#)
- [Menyetel konfigurasi baru dan contoh perintah kredensial](#)
- [Pengaturan config file yang didukung](#)

## Format konfigurasi dan file kredensial

`credentialsFile config` dan disusun menjadi beberapa bagian. Bagian termasuk profil, sesi sso, dan layanan. Bagian adalah kumpulan pengaturan bernama, dan berlanjut sampai garis definisi bagian lain ditemukan. Beberapa profil dan bagian dapat disimpan dalam `credentials file config` dan.

File-file ini adalah file teks biasa yang menggunakan format berikut:

- Nama bagian terlampir dalam tanda kurung [] seperti [default],[profile *user1*], dan. [sso-session]
- Semua entri dalam suatu bagian mengambil bentuk umum dari. setting\_name=value
- Baris dapat dikomentari dengan memulai baris dengan karakter hash (#).

credentialsFile config dan berisi jenis bagian berikut:

- [Jenis bagian: profile](#)
- [Jenis bagian: sso-session](#)
- [Jenis bagian: services](#)

## Jenis bagian: **profile**

AWS CLI Toko-toko

Tergantung pada file, nama bagian profil menggunakan format berikut:

- Berkas Config: [default] [profile *user1*]
- File kredensial: [default] [*user1*]

Jangan gunakan kata profile saat membuat entri dalam credentials file.

Setiap profil dapat menentukan kredensial yang berbeda dan juga dapat menentukan AWS Wilayah dan format output yang berbeda. Saat menamai profil dalam config file, sertakan kata awalan "profile", tetapi jangan sertakan dalam credentials file.

Contoh berikut menunjukkan config file credentials dan dengan dua profil, wilayah, dan output yang ditentukan. [default] pertama digunakan saat Anda menjalankan AWS CLI perintah tanpa profil yang ditentukan. Yang kedua digunakan ketika Anda menjalankan AWS CLI perintah dengan --profile user1 parameter.

## IAM Identity Center (SSO)

Contoh ini untuk AWS IAM Identity Center. Untuk informasi selengkapnya, lihat [the section called "Konfigurasi penyegaran token otomatis"](#).

### File kredensial

`credentialsFile` tidak digunakan untuk metode otentikasi ini.

## Berkas Config

```
[default]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text

[profile user1]
sso_session = my-sso
sso_account_id = 444455556666
sso_role_name = readOnly
region = us-east-1
output = json

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

## IAM Identity Center (Legacy SSO)

Contoh ini untuk metode warisan. AWS IAM Identity Center Untuk informasi selengkapnya, lihat [the section called “Konfigurasi lawas yang tidak dapat disegarkan”](#).

## File kredensial

`credentialsFile` tidak digunakan untuk metode otentikasi ini.

## Berkas Config

```
[default]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text
```





```
credential_source=Ec2InstanceMetadata
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
credential_source=Ec2InstanceMetadata
region=us-east-1
output=text
```

## Long-term credentials

### Warning

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti [AWS IAM Identity Center](#).

Contoh ini untuk kredensi jangka panjang dari AWS Identity and Access Management Untuk informasi selengkapnya, lihat [the section called “Pengguna IAM”](#).

## File kredensial

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY

[user1]
aws_access_key_id=AKIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

## Berkas Config

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

Untuk informasi selengkapnya dan otorisasi tambahan dan metode kredensi lihat, lihat. [the section called “Pengguna IAM”](#)

## Jenis bagian: **sso-session**

sso-session Bagian config file digunakan untuk mengelompokkan variabel konfigurasi untuk memperoleh token akses SSO, yang kemudian dapat digunakan untuk memperoleh AWS kredensial. Pengaturan berikut digunakan:

- (Diperlukan) [sso\\_start\\_url](#)
- (Diperlukan) [sso\\_region](#)
- [sso\\_account\\_id](#)
- [sso\\_role\\_name](#)
- [sso\\_registration\\_scopes](#)

Anda menentukan sso-session bagian dan mengaitkannya ke profil. sso\_region dan sso\_start\_url harus diatur dalam sso-session bagian. Biasanya, sso\_account\_id dan sso\_role\_name harus diatur di profile bagian sehingga SDK dapat meminta kredensial SSO.

Contoh berikut mengonfigurasi SDK untuk meminta kredensial SSO dan mendukung penyegaran token otomatis:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Ini juga memungkinkan sso-session konfigurasi untuk digunakan kembali di beberapa profil:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole
```



```
[profile prod]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole2

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Namun, `sso_account_id` dan `sso_role_name` tidak diperlukan untuk semua skenario konfigurasi token SSO. Jika aplikasi Anda hanya menggunakan AWS layanan yang mendukung otentikasi pembawa, maka AWS kredensi tradisional tidak diperlukan. Otentikasi pembawa adalah skema otentikasi HTTP yang menggunakan token keamanan yang disebut token pembawa. Dalam skenario ini, `sso_account_id` dan `sso_role_name` tidak diperlukan. Lihat panduan individual untuk AWS layanan Anda untuk menentukan apakah itu mendukung otorisasi token pembawa.

Selain itu, cakupan pendaftaran dapat dikonfigurasi sebagai bagian dari file. `sso-session` Cakupan adalah mekanisme di OAuth 2.0 untuk membatasi akses aplikasi ke akun pengguna. Aplikasi dapat meminta satu atau lebih cakupan, dan token akses yang dikeluarkan untuk aplikasi akan terbatas pada cakupan yang diberikan. Cakupan ini menentukan izin yang diminta untuk diotorisasi untuk klien OIDC terdaftar dan token akses yang diambil oleh klien. Contoh berikut menetapkan `sso_registration_scopes` untuk menyediakan akses untuk daftar akun/peran:

```
[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

Token otentikasi di-cache ke disk di bawah `~/.aws/sso/cache` direktori dengan nama file berdasarkan nama sesi.

Untuk informasi selengkapnya tentang jenis konfigurasi ini, lihat [the section called “Konfigurasi penyegaran token otomatis”](#).

## Jenis bagian: **services**

`services` Bagian ini adalah sekelompok pengaturan yang mengonfigurasi titik akhir kustom untuk Layanan AWS permintaan. Profil kemudian ditautkan ke `services` bagian.

```
[profile dev]
```

```
services = my-services
```

servicesBagian ini dipisahkan menjadi subbagian dengan <SERVICE> = baris, di mana <SERVICE> adalah kunci Layanan AWS pengenalan. Layanan AWS Pengenalan didasarkan pada model API serviceId dengan mengganti semua spasi dengan garis bawah dan huruf kecil semua huruf. Untuk daftar semua kunci pengenalan layanan yang akan digunakan di services bagian ini, lihat [Gunakan titik akhir di AWS CLI](#). Kunci pengenalan layanan diikuti oleh pengaturan bersarang dengan masing-masing pada barisnya sendiri dan menjorok oleh dua spasi.

*Contoh berikut mengonfigurasi titik akhir yang akan digunakan untuk permintaan yang dibuat ke Amazon DynamoDB layanan di bagian layanan saya yang digunakan dalam profil dev.* Baris yang segera mengikuti yang menjorok termasuk dalam ayat itu dan berlaku untuk layanan itu.

```
[profile dev]  
services = my-services  
  
[services my-services]  
dynamodb =  
  endpoint_url = http://localhost:8000
```

Untuk informasi selengkapnya tentang titik akhir khusus layanan, lihat [Gunakan titik akhir di AWS CLI](#)

Jika profil Anda memiliki kredensial berbasis peran yang dikonfigurasi melalui `source_profile` parameter untuk fungsionalitas peran IAM, SDK hanya menggunakan konfigurasi layanan untuk profil yang ditentukan. Itu tidak menggunakan profil yang dirantai peran untuk itu. Misalnya, menggunakan config file bersama berikut:

```
[profile A]  
credential_source = Ec2InstanceMetadata  
endpoint_url = https://profile-a-endpoint.aws/  
  
[profile B]  
source_profile = A  
role_arn = arn:aws:iam::123456789012:role/roleB  
services = profileB  
  
[services profileB]  
ec2 =
```

```
endpoint_url = https://profile-b-ec2-endpoint.aws
```

Jika Anda menggunakan profil B dan membuat panggilan dalam kode Anda ke Amazon EC2, titik akhir akan diselesaikan sebagai `https://profile-b-ec2-endpoint.aws`. Jika kode Anda membuat permintaan ke layanan lain, resolusi titik akhir tidak akan mengikuti logika kustom apa pun. Titik akhir tidak menyelesaikan titik akhir global yang ditentukan dalam profil. Agar titik akhir global berlaku untuk profil B, Anda perlu mengatur `endpoint_url` langsung di dalam profil B.

## Di mana pengaturan konfigurasi disimpan?

AWS CLI Menyimpan informasi kredensi sensitif yang Anda tentukan `aws configure` dalam file lokal bernama `credentials`, dalam folder bernama `.aws` di direktori home Anda. Opsi konfigurasi yang kurang sensitif yang Anda tentukan `aws configure` disimpan dalam file lokal bernama `config`, juga disimpan di `.aws` folder di direktori home Anda.

### Menyimpan kredensi dalam file konfigurasi

Anda dapat menyimpan semua pengaturan profil Anda dalam satu file karena AWS CLI dapat membaca kredensial dari file `config`. Jika ada kredensial di kedua file untuk profil yang berbagi nama yang sama, kunci dalam file kredensial diutamakan. Kami menyarankan untuk menyimpan kredensial dalam file `credentials`. File-file ini juga digunakan oleh berbagai kit pengembangan perangkat lunak bahasa (SDK). Jika Anda menggunakan salah satu SDK selain AWS CLI, konfirmasi apakah kredensialnya harus disimpan dalam file mereka sendiri.

Di mana Anda menemukan lokasi direktori home Anda bervariasi berdasarkan sistem operasi, tetapi disebut menggunakan variabel lingkungan `%UserProfile%` di Windows dan `$HOME` atau `~` (tilde) dalam sistem berbasis Unix. Anda dapat menentukan lokasi non-default untuk file dengan mengatur variabel `AWS_CONFIG_FILE` dan `AWS_SHARED_CREDENTIALS_FILE` lingkungan ke jalur lokal lain. Lihat [Variabel lingkungan untuk mengkonfigurasi AWS CLI](#) untuk detail.

Bila Anda menggunakan profil bersama yang menentukan peran AWS Identity and Access Management (IAM), operasi akan AWS CLI memanggil AWS STS `AssumeRole` operasi untuk mengambil kredensi sementara. Kredensial ini kemudian disimpan (in `~/ .aws/cli/cache`). AWS CLI Perintah selanjutnya menggunakan kredensial sementara yang di-cache sampai kedaluwarsa, dan pada saat itu AWS CLI secara otomatis menyegarkan kredensialnya.

## Menggunakan profil bernama

Jika tidak ada profil yang didefinisikan secara eksplisit, default profil digunakan.

Untuk menggunakan profil bernama, tambahkan `--profile profile-name` opsi ke perintah Anda. Contoh berikut mencantumkan semua instans Amazon EC2 Anda menggunakan kredensi dan setelan yang ditentukan dalam profil. `user1`

```
$ aws ec2 describe-instances --profile user1
```

Untuk menggunakan profil bernama untuk beberapa perintah, Anda dapat menghindari menentukan profil di setiap perintah dengan mengatur variabel `AWS_PROFILE` lingkungan sebagai profil default. Anda dapat mengganti pengaturan ini dengan menggunakan `--profile` parameter.

Linux or macOS

```
$ export AWS_PROFILE=user1
```

Windows

```
C:\> setx AWS_PROFILE user1
```

Menggunakan [set](#) untuk mengatur variabel lingkungan mengubah nilai yang digunakan sampai akhir sesi prompt perintah saat ini, atau sampai Anda mengatur variabel ke nilai yang berbeda.

Menggunakan [setx](#) untuk mengatur variabel lingkungan mengubah nilai di semua shell perintah yang Anda buat setelah menjalankan perintah. Itu tidak mempengaruhi shell perintah apa pun yang sudah berjalan pada saat Anda menjalankan perintah. Tutup dan mulai ulang shell perintah untuk melihat efek perubahan.

Menyetel variabel lingkungan akan mengubah profil default hingga akhir sesi shell, atau hingga Anda menyetel variabel ke nilai yang berbeda. Anda dapat membuat variabel lingkungan persisten di seluruh sesi masa depan dengan memasukkannya ke dalam skrip startup shell Anda. Untuk informasi selengkapnya, lihat [Variabel lingkungan untuk mengkonfigurasi AWS CLI](#).

## Mengatur dan melihat pengaturan konfigurasi menggunakan perintah

Ada beberapa cara untuk melihat dan mengatur pengaturan konfigurasi Anda menggunakan perintah.

## aws configure

Jalankan perintah ini untuk mengatur dan melihat kredensial, Wilayah, dan format output dengan cepat. Contoh berikut menunjukkan nilai sampel.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

## aws configure set

Anda dapat mengatur kredensi atau pengaturan konfigurasi apa pun menggunakan `aws configure set`. Tentukan profil yang ingin Anda lihat atau modifikasi dengan `--profile` pengaturan.

Misalnya, perintah berikut menetapkan `region` dalam profil bernama `integ`.

```
$ aws configure set region us-west-2 --profile integ
```

Untuk menghapus pengaturan, gunakan string kosong sebagai nilai, atau hapus pengaturan secara manual di `credentials` file Anda `config` dan dalam editor teks.

```
$ aws configure set cli_pager "" --profile integ
```

## aws configure get

Anda dapat mengambil semua kredensi atau pengaturan konfigurasi yang telah Anda tetapkan. `aws configure get`. Tentukan profil yang ingin Anda lihat atau modifikasi dengan `--profile` pengaturan.

Misalnya, perintah berikut mengambil `region` pengaturan di profil bernama `integ`.

```
$ aws configure get region --profile integ
us-west-2
```

Jika output kosong, pengaturan tidak secara eksplisit diatur dan menggunakan nilai default.

## aws configure import

Impor CSV kredensi yang dihasilkan dari konsol web IAM. Ini bukan untuk kredensial yang dihasilkan dari IAM Identity Center; pelanggan yang menggunakan IAM Identity Center harus menggunakan `aws configure sso`. File CSV diimpor dengan nama profil yang cocok dengan nama pengguna. File CSV harus berisi header berikut.

- Nama Pengguna
- ID kunci akses
- Kunci akses rahasia

### Note

Selama pembuatan key pair awal, setelah Anda menutup kotak dialog Download file.csv, Anda tidak dapat mengakses kunci akses rahasia Anda setelah Anda menutup kotak dialog. Jika Anda memerlukan .csv file, Anda harus membuatnya sendiri dengan header yang diperlukan dan informasi key pair yang disimpan. Jika Anda tidak memiliki akses ke informasi key pair Anda, Anda perlu membuat key pair baru.

```
$ aws configure import --csv file://credentials.csv
```

## aws configure list

Untuk membuat daftar data konfigurasi, gunakan `aws configure list` perintah. Perintah ini mencantumkan profil, kunci akses, kunci rahasia, dan informasi konfigurasi wilayah yang digunakan untuk profil yang ditentukan. Untuk setiap item konfigurasi, ini menunjukkan nilai, di mana nilai konfigurasi diambil, dan nama variabel konfigurasi.

Misalnya, jika Anda memberikan variabel Wilayah AWS dalam lingkungan, perintah ini menunjukkan nama wilayah yang telah Anda konfigurasi, bahwa nilai ini berasal dari variabel lingkungan, dan nama variabel lingkungan.

Untuk metode kredensi sementara seperti peran dan Pusat Identitas IAM, perintah ini menampilkan kunci akses cache sementara dan kunci akses rahasia ditampilkan.

```
$ aws configure list
```

Name	Value	Type	Location
----	-----	----	-----
profile	<not set>	None	None

```
access_key      *****ABCD  shared-credentials-file
secret_key      *****ABCD  shared-credentials-file
region          us-west-2     env    AWS_DEFAULT_REGION
```

## aws configure list-profiles

Untuk mencantumkan semua nama profil Anda, gunakan `aws configure list-profiles` perintah.

```
$ aws configure list-profiles
default
test
```

## aws configure sso

Jalankan perintah ini untuk mengatur dan melihat AWS IAM Identity Center kredensial, Wilayah, dan format output dengan cepat. Contoh berikut menunjukkan nilai sampel.

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

## aws configure sso-session

Jalankan perintah ini untuk dengan cepat mengatur dan melihat AWS IAM Identity Center kredensial Anda, Wilayah, dan format output di bagian `sso-session` dan `file.credentials config` Contoh berikut menunjukkan nilai sampel.

```
$ aws configure sso-session
SSO session name: my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

## Menyetel konfigurasi baru dan contoh perintah kredensial

Contoh berikut menunjukkan konfigurasi profil default dengan kredensial, wilayah, dan output yang ditentukan untuk metode otentikasi yang berbeda.

## IAM Identity Center (SSO)

Contoh ini untuk AWS IAM Identity Center menggunakan `aws configure sso wizard`. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi penyegaran token otomatis”](#).

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

Attempting to automatically open the SSO authorization page in your default browser.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
   ProductionAccount, production-account-admin@example.com (444455556666)

Using the account ID 111122223333

There are 2 roles available to you.
> ReadOnly
   FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```

## IAM Identity Center (Legacy SSO)

Contoh ini untuk metode lama AWS IAM Identity Center menggunakan `aws configure sso wizard`. Untuk menggunakan SSO lama, biarkan nama sesi kosong. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi lawas yang tidak dapat disegarkan”](#).

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.

There are 2 AWS accounts available to you.
```





```
$ aws configure set role_session_name session_user1
$ aws configure set region us-west-2
$ aws configure set output json
```

## Amazon EC2 instance metadata credentials

Contoh ini untuk kredensial yang diperoleh dari metadata instans Amazon EC2 hosting. Tidak ada wizard untuk proses ini, oleh karena itu setiap nilai diatur menggunakan `aws configure set` perintah. Untuk informasi selengkapnya, lihat [the section called “Menggunakan kredensyal untuk metadata instans Amazon EC2”](#).

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set credential_source Ec2InstanceMetadata
$ aws configure set region us-west-2
$ aws configure set output json
```

## Long-term credentials

### Warning

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti [AWS IAM Identity Center](#).

Contoh ini untuk kredensi jangka panjang dari AWS Identity and Access Management Untuk informasi selengkapnya, lihat [the section called “Pengguna IAM”](#).

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

## Pengaturan **config** file yang didukung

### Topik

- [Pengaturan global](#)

- [Pengaturan perintah kustom S3](#)

Pengaturan berikut didukung dalam `config` file. Nilai yang tercantum dalam profil yang ditentukan (atau default) digunakan kecuali jika diganti dengan adanya variabel lingkungan dengan nama yang sama, atau opsi baris perintah dengan nama yang sama. Untuk informasi selengkapnya tentang pengaturan pesanan apa yang didahulukan, lihat [Konfigurasi AWS CLI](#)

## Pengaturan global

### **aws\_access\_key\_id**

Menentukan kunci AWS akses yang digunakan sebagai bagian dari kredensial untuk mengautentikasi permintaan perintah. Meskipun ini dapat disimpan dalam `config` file, kami sarankan Anda menyimpannya di `credentials` file.

Dapat diganti oleh variabel lingkungan. `AWS_ACCESS_KEY_ID` Anda tidak dapat menentukan ID kunci akses sebagai opsi baris perintah.

```
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
```

### **aws\_secret\_access\_key**

Menentukan kunci AWS rahasia yang digunakan sebagai bagian dari kredensial untuk mengautentikasi permintaan perintah. Meskipun ini dapat disimpan dalam `config` file, kami sarankan Anda menyimpannya di `credentials` file.

Dapat diganti oleh variabel lingkungan. `AWS_SECRET_ACCESS_KEY` Anda tidak dapat menentukan kunci akses rahasia sebagai opsi baris perintah.

```
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

### **aws\_session\_token**

Menentukan token AWS sesi. Token sesi hanya diperlukan jika Anda secara manual menentukan kredensial keamanan sementara. Meskipun ini dapat disimpan dalam `config` file, kami sarankan Anda menyimpannya di `credentials` file.

Dapat diganti oleh variabel lingkungan. `AWS_SESSION_TOKEN` Anda tidak dapat menentukan token sesi sebagai opsi baris perintah.

```
aws_session_token = AQoEXAMPLEH4aoAH0gNCAPyJxz4BlCFFxWNE1OPTgk5TthT
+FvwnKwRcOIfrRh3c/LTo6UDdyJw00vEVPvLXCrrrUtdnniCEXAMPLE/
IvU1dYUg2RVAJBanLiHb4IgRmpRV3zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
```

## ca\_bundle

Menentukan bundel sertifikat CA (file dengan .pem ekstensi) yang digunakan untuk memverifikasi sertifikat SSL.

Dapat diganti oleh variabel [AWS\\_CA\\_BUNDLE](#) lingkungan atau opsi [--ca-bundle](#) baris perintah.

```
ca_bundle = dev/apps/ca-certs/cabundle-2019mar05.pem
```

## cli\_auto\_prompt

Mengaktifkan prompt otomatis untuk AWS CLI versi 2. Ada dua pengaturan yang dapat digunakan:

- **on** menggunakan mode auto-prompt penuh setiap kali Anda mencoba menjalankan aws perintah. Ini termasuk menekan ENTER setelah perintah lengkap atau perintah tidak lengkap.

```
cli_auto_prompt = on
```

- **on-partial** menggunakan mode auto-prompt sebagian. Jika perintah tidak lengkap atau tidak dapat dijalankan karena kesalahan validasi sisi klien, prompt otomatis digunakan. Mode ini sangat berguna jika Anda memiliki skrip, runbook, atau Anda hanya ingin diminta secara otomatis untuk perintah yang tidak Anda kenal daripada diminta pada setiap perintah.

```
cli_auto_prompt = on-partial
```

Anda dapat mengganti pengaturan ini dengan menggunakan variabel [aws\\_cli\\_auto\\_prompt](#) lingkungan atau parameter baris [--no-cli-auto-prompt](#) perintah [--cli-auto-prompt](#) dan.

Untuk informasi tentang fitur prompt otomatis AWS CLI versi 2, lihat [AWS CLIMinta Anda meminta perintah](#).

## cli\_binary\_format

Menentukan bagaimana AWS CLI versi 2 menafsirkan parameter input biner. Ini bisa menjadi salah satu nilai berikut:

- `base64` - Ini adalah nilai default. Parameter input yang diketik sebagai objek besar biner (BLOB) menerima string yang dikodekan base64. Untuk meneruskan konten biner yang benar, letakkan konten dalam file dan berikan path dan nama file dengan `fileb://` awalan sebagai nilai parameter. Untuk meneruskan teks berenkode base64 yang terkandung dalam file, berikan path dan nama file dengan `file://` awalan sebagai nilai parameter.
- `raw-in-base64-out` - Default untuk AWS CLI versi 1. Jika nilai pengaturan adalah `raw-in-base64-out`, file yang direferensikan menggunakan `file://` awalan dibaca sebagai teks dan kemudian AWS CLI upaya untuk menyandikannya ke biner.

Entri ini tidak memiliki variabel lingkungan yang setara. Anda dapat menentukan nilai pada satu perintah dengan menggunakan `--cli-binary-format raw-in-base64-out` parameter.

```
cli_binary_format = raw-in-base64-out
```

Jika Anda mereferensikan nilai biner dalam file menggunakan notasi `fileb://` awalan, AWS CLI selalu mengharapkan file tersebut berisi konten biner mentah dan tidak mencoba mengonversi nilainya.

Jika Anda mereferensikan nilai biner dalam file menggunakan notasi `file://` awalan, AWS CLI menangani file sesuai dengan pengaturan saat ini `cli_binary_format`. Jika nilai pengaturan itu `base64` (default saat tidak disetel secara eksplisit), file tersebut AWS CLI mengharapkan file berisi teks yang dikodekan base64. Jika nilai pengaturan itu `raw-in-base64-out`, file AWS CLI mengharapkan berisi konten biner mentah.

## **cli\_history**

Dinonaktifkan secara default. Pengaturan ini memungkinkan riwayat perintah untuk file AWS CLI. Setelah mengaktifkan pengaturan ini, AWS CLI catatan sejarah aws perintah.

```
cli_history = enabled
```

Anda dapat membuat daftar riwayat Anda menggunakan `aws history list` perintah, dan menggunakan `aws history show` perintah yang dihasilkan `command_ids` untuk detailnya. Untuk informasi lebih lanjut lihat di panduan AWS CLI referensi.

## **cli\_pager**

Menentukan program pager yang digunakan untuk output. Secara default, AWS CLI versi 2 mengembalikan semua output melalui program pager default sistem operasi Anda.

Dapat diganti oleh variabel lingkungan `AWS_PAGER`.

```
cli_pager=less
```

Untuk menonaktifkan semua penggunaan program paging eksternal, atur variabel ke string kosong seperti yang ditunjukkan pada contoh berikut.

```
cli_pager=
```

## **cli\_timestamp\_format**

Menentukan format nilai timestamp termasuk dalam output. Anda dapat menentukan salah satu dari nilai berikut:

- iso8601 — Nilai default untuk versi 2. AWS CLI [Jika ditentukan, AWS CLI format ulang semua stempel waktu sesuai dengan ISO 8601.](#)

Stempel waktu berformat ISO 8601 terlihat seperti contoh berikut. Contoh pertama menunjukkan waktu dalam [Coordinated Universal Time \(UTC\)](#) dengan memasukkan Z setelah waktu. Tanggal dan waktu dipisahkan oleh aT.

```
2019-10-31T22:21:41Z
```

Untuk menentukan zona waktu yang berbeda, alih-alihZ, tentukan + atau - dan jumlah jam zona waktu yang diinginkan berada di depan atau di belakang UTC, sebagai nilai dua digit. Contoh berikut menunjukkan waktu yang sama dengan contoh sebelumnya tetapi disesuaikan dengan waktu Standar Pasifik, yaitu delapan jam di belakang UTC.

```
2019-10-31T14:21:41-08
```

- wire - Nilai default untuk AWS CLI versi 1. Jika ditentukan, AWS CLI menampilkan semua nilai stempel waktu persis seperti yang diterima dalam respons kueri HTTP.

Entri ini tidak memiliki variabel lingkungan yang setara atau opsi baris perintah.

```
cli_timestamp_format = iso8601
```

## **credential\_process**

Menentukan perintah eksternal yang AWS CLI berjalan untuk menghasilkan atau mengambil kredensi otentikasi untuk digunakan untuk perintah ini. Perintah harus mengembalikan kredensi

dalam format tertentu. Untuk informasi selengkapnya tentang cara menggunakan setelan ini, lihat [Kredensi sumber dengan proses eksternal](#).

Entri ini tidak memiliki variabel lingkungan yang setara atau opsi baris perintah.

```
credential_process = /opt/bin/awscreds-retriever --username susan
```

## **credential\_source**

Digunakan dalam instans atau container Amazon EC2 untuk menentukan di mana kredensial AWS CLI dapat ditemukan yang akan digunakan untuk mengambil peran yang Anda tentukan dengan parameter `role_arn`. Anda tidak dapat menentukan keduanya `source_profile` dan `credential_source` di profil yang sama.

Parameter ini dapat memiliki salah satu dari tiga nilai:

- `Lingkungan` - Menentukan bahwa AWS CLI adalah untuk mengambil kredensial sumber dari variabel lingkungan.
- `Ec2 InstanceMetadata` - Menentukan bahwa AWS CLI adalah untuk menggunakan peran IAM yang dilampirkan ke [profil instans EC2](#) untuk mendapatkan kredensi sumber.
- `EcsContainer`— Menentukan bahwa AWS CLI adalah untuk menggunakan peran IAM yang dilampirkan ke wadah ECS sebagai kredensi sumber.

```
credential_source = Ec2InstanceMetadata
```

## **duration\_seconds**

Menentukan durasi maksimum sesi peran, dalam hitungan detik. Nilai dapat berkisar dari 900 detik (15 menit) hingga pengaturan durasi sesi maksimum untuk peran (yang bisa maksimal 43200). Ini adalah parameter opsional dan secara default, nilainya diatur ke 3600 detik.

## **endpoint\_url**

Menentukan endpoint yang digunakan untuk semua permintaan layanan. Jika pengaturan ini digunakan di [services](#) bagian config file, maka titik akhir hanya digunakan untuk layanan yang ditentukan.

Contoh berikut menggunakan endpoint global `http://localhost:1234` dan endpoint khusus layanan untuk Amazon `http://localhost:4567 S3`.

```
[profile dev]
endpoint_url = http://localhost:1234
```

```
services = s3-specific

[services s3-specific]
s3 =
  endpoint_url = http://localhost:4567
```

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau dinyatakan secara eksplisit pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris `--endpoint-url` perintah.
2. Jika diaktifkan, variabel lingkungan endpoint `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global atau pengaturan profil `ignore_configure_endpoint_urls` untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan `AWS_ENDPOINT_URL_<SERVICE>`, seperti `AWS_ENDPOINT_URL_DYNAMODB`.
4. Nilai yang disediakan oleh `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT`, dan variabel `AWS_ENDPOINT_URL` lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh `endpoint_url` pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh `endpoint_url` pengaturan profile dalam `config` file bersama.
7. `use_dualstack_endpoint`, `use_fips_endpoint`, dan `endpoint_url` pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## `ignore_configure_endpoint_urls`

Jika diaktifkan, AWS CLI mengabaikan semua konfigurasi titik akhir kustom yang ditentukan dalam file. `config` Nilai yang valid adalah **true** dan **false**.

```
ignore_configure_endpoint_urls = true
```

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau dinyatakan secara eksplisit pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:



1. Opsi baris `--endpoint-url` perintah.
2. Jika diaktifkan, variabel lingkungan endpoint `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global atau pengaturan profil `ignore_configure_endpoint_urls` untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan `AWS_ENDPOINT_URL_<SERVICE>`, seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT`, dan variabel `AWS_ENDPOINT_URL` lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh `endpoint_url` pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh `endpoint_url` pengaturan `profile` dalam `config` file bersama.
7. `use_dualstack_endpoint`, `use_fips_endpoint`, dan `endpoint_url` pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## external\_id

Menentukan pengenal unik yang digunakan oleh pihak ketiga untuk mengambil peran dalam akun pelanggan mereka. Ini memetakan ke `ExternalId` parameter dalam `AssumeRole` operasi. Parameter ini diperlukan hanya jika kebijakan kepercayaan untuk peran menentukan nilai untuk `ExternalId`. Untuk informasi selengkapnya, lihat [Cara menggunakan ID eksternal saat memberikan akses ke AWS sumber daya Anda kepada pihak ketiga](#) dalam Panduan Pengguna IAM.

## max\_attempts

Menentukan nilai percobaan ulang maksimum yang digunakan oleh penanganan AWS CLI coba lagi, di mana panggilan awal diperhitungkan terhadap nilai yang Anda berikan. `max_attempts`

Anda dapat mengganti nilai ini dengan menggunakan variabel `AWS_MAX_ATTEMPTS` lingkungan.

```
max_attempts = 3
```

## mfa\_serial

Nomor identifikasi perangkat MFA yang akan digunakan saat mengambil peran. Ini wajib hanya jika kebijakan kepercayaan dari peran yang diasumsikan mencakup kondisi yang memerlukan otentikasi MFA. Nilai dapat berupa nomor seri untuk perangkat keras

(sepertiGAHT12345678) atau Amazon Resource Name (ARN) untuk perangkat MFA virtual (seperti). `arn:aws:iam::123456789012:mfa/user`

## output

Menentukan format output default untuk perintah yang diminta menggunakan profil ini. Anda dapat menentukan salah satu nilai berikut:

- **json**— Output diformat sebagai string [JSON](#).
- **yaml**— Output diformat sebagai string [YAMB](#).
- **yaml-stream**— Output dialirkan dan diformat sebagai string [YAMM](#). Streaming memungkinkan penanganan tipe data besar yang lebih cepat.
- **text**- Output diformat sebagai beberapa baris nilai string yang dipisahkan tab. Ini dapat berguna untuk meneruskan output ke prosesor teks, seperti `grep`, `sed`, atau `awk`.
- **table**— Output diformat sebagai tabel menggunakan karakter `+|-` untuk membentuk batas sel. Ini biasanya menyajikan informasi dalam format “ramah manusia” yang jauh lebih mudah dibaca daripada yang lain, tetapi tidak berguna secara terprogram.

Dapat diganti oleh variabel `AWS_DEFAULT_OUTPUT` lingkungan atau opsi `--output` baris perintah.

```
output = table
```

## parameter\_validation

Menentukan apakah AWS CLI klien mencoba untuk memvalidasi parameter sebelum mengirim mereka ke endpoint AWS layanan.

- `true` - Ini adalah nilai default. Jika ditentukan, AWS CLI melakukan validasi lokal parameter baris perintah.
- `false` — Jika ditentukan, AWS CLI tidak memvalidasi parameter baris perintah sebelum mengirimnya ke titik akhir AWS layanan.

Entri ini tidak memiliki variabel lingkungan yang setara atau opsi baris perintah.

```
parameter_validation = false
```

## region

Menentukan Wilayah AWS untuk mengirim permintaan untuk perintah yang diminta menggunakan profil ini.

- Anda dapat menentukan salah satu kode Wilayah yang tersedia untuk layanan yang dipilih seperti yang tercantum di [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services.
- `aws_global` memungkinkan Anda menentukan titik akhir global untuk layanan yang mendukung titik akhir global selain titik akhir Regional, seperti AWS Security Token Service (AWS STS) dan Amazon Simple Storage Service (Amazon S3).

Anda dapat mengganti nilai ini dengan menggunakan variabel `AWS_REGION` lingkungan, variabel `AWS_DEFAULT_REGION` lingkungan, atau opsi baris `--region` perintah.

```
region = us-west-2
```

## [retry\\_mode](#)

Menentukan modus AWS CLI coba lagi yang menggunakan. Ada tiga mode coba lagi yang tersedia: legacy (default), standar, dan adaptif. Untuk informasi lebih lanjut tentang percobaan ulang, lihat [AWS CLI mencoba lagi](#).

Anda dapat mengganti nilai ini dengan menggunakan variabel `AWS_RETRY_MODE` lingkungan.

```
retry_mode = standard
```

## [role\\_arn](#)

Menentukan Amazon Resource Name (ARN) dari peran IAM yang ingin Anda gunakan untuk menjalankan perintah. AWS CLI Anda juga harus menentukan salah satu parameter berikut untuk mengidentifikasi kredensial yang memiliki izin untuk mengambil peran ini:

- `source_profile`
- `credential_source`

```
role_arn = arn:aws:iam::123456789012:role/role-name
```

Variabel lingkungan `AWS_ROLE_ARN` mengesampingkan pengaturan ini.

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called "Asumsikan peran dengan identitas web"](#).

## role\_session\_name

Menentukan nama untuk melampirkan ke sesi peran. Nilai ini disediakan untuk RoleSessionName parameter ketika AWS CLI panggilan AssumeRole operasi, dan menjadi bagian dari peran yang diasumsikan pengguna ARN:.

`arn:aws:sts::123456789012:assumed-role/role_name/role_session_name` Ini adalah parameter opsional. Jika Anda tidak memberikan nilai ini, nama sesi dibuat secara otomatis. Nama ini muncul di AWS CloudTrail log untuk entri yang terkait dengan sesi ini.

```
role_session_name = maria_garcia_role
```

Variabel lingkungan [AWS\\_ROLE\\_SESSION\\_NAME](#) mengesampingkan pengaturan ini.

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called “Asumsikan peran dengan identitas web”](#).

## services

Menentukan konfigurasi layanan yang akan digunakan untuk profil Anda.

```
[profile dev-s3-specific-and-global]  
endpoint_url = http://localhost:1234  
services = s3-specific  
  
[services s3-specific]  
s3 =  
  endpoint_url = http://localhost:4567
```

Untuk informasi lebih lanjut tentang services bagian ini, lihat [the section called “services”](#).

Variabel lingkungan [AWS\\_ROLE\\_SESSION\\_NAME](#) mengesampingkan pengaturan ini.

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called “Asumsikan peran dengan identitas web”](#).

## source\_profile

Menentukan profil bernama dengan kredensi jangka panjang yang AWS CLI dapat digunakan untuk mengambil peran yang Anda tentukan dengan parameter. `role_arn` Anda tidak dapat menentukan keduanya `source_profile` dan `credential_source` di profil yang sama.

```
source_profile = production-profile
```

### sso\_account\_id

Menentukan ID AWS akun yang berisi peran IAM dengan izin yang ingin Anda berikan kepada pengguna Pusat Identitas IAM terkait.

Pengaturan ini tidak memiliki variabel lingkungan atau opsi baris perintah.

```
sso_account_id = 123456789012
```

### sso\_region

Menentukan AWS Wilayah yang berisi host portal AWS akses. Ini terpisah dari, dan dapat menjadi Wilayah yang berbeda dari parameter CLI `region` default.

Pengaturan ini tidak memiliki variabel lingkungan atau opsi baris perintah.

```
sso_region = us-west-2
```

### sso\_registration\_scopes

Daftar cakupan yang dibatasi koma yang akan diotorisasi untuk `sso-session` Cakupan mengotorisasi akses ke titik akhir resmi token pembawa IAM Identity Center. Lingkup yang valid adalah string, seperti `sso:account:access`. Pengaturan ini tidak berlaku untuk konfigurasi lama yang tidak dapat disegarkan.

```
sso_registration_scopes = sso:account:access
```

### sso\_role\_name

Menentukan nama ramah peran IAM yang mendefinisikan izin pengguna saat menggunakan profil ini.

Pengaturan ini tidak memiliki variabel lingkungan atau opsi baris perintah.

```
sso_role_name = ReadAccess
```

## [sso\\_start\\_url](#)

Menentukan URL yang menunjuk ke portal AWS akses organisasi. AWS CLI Menggunakan URL ini untuk membuat sesi dengan layanan IAM Identity Center untuk mengautentikasi penggunanya. Untuk menemukan URL portal AWS akses Anda, gunakan salah satu dari berikut ini:

- Buka email undangan Anda, URL portal AWS akses terdaftar.
- Buka AWS IAM Identity Center konsol di <https://console.aws.amazon.com/singlesignon/>. URL portal AWS akses tercantum dalam pengaturan Anda.

Pengaturan ini tidak memiliki variabel lingkungan atau opsi baris perintah.

```
sso_start_url = https://my-sso-portal.awsapps.com/start
```

## [use\\_dualstack\\_endpoint](#)

Mengaktifkan penggunaan titik akhir dual-stack untuk mengirim permintaan. AWS Untuk mempelajari lebih lanjut tentang titik akhir dual-stack, yang mendukung lalu lintas IPv4 dan IPv6, lihat Menggunakan titik akhir [tumpukan ganda Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Titik akhir dual-stack tersedia untuk beberapa layanan di beberapa wilayah. Jika titik akhir tumpukan ganda tidak ada untuk layanan atau Wilayah AWS, permintaan gagal. Ini dinonaktifkan secara default.

Ini saling eksklusif dengan `use_accelerate_endpoint` pengaturan.

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau dinyatakan secara eksplisit pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris `--endpoint-url` perintah.
2. Jika diaktifkan, variabel lingkungan endpoint `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global atau pengaturan profil `ignore_configure_endpoint_urls` untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan `AWS_ENDPOINT_URL_<SERVICE>`, seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT`, dan variabel `AWS_ENDPOINT_URL` lingkungan.

5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan `profile` dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

### *use\_fips\_endpoint*

Beberapa AWS layanan menawarkan titik akhir yang mendukung [Federal Information Processing Standard \(FIPS\) 140-2](#) di beberapa Wilayah AWS. Ketika AWS layanan mendukung FIPS, pengaturan ini menentukan titik akhir FIPS apa yang harus digunakan. AWS CLI Tidak seperti AWS endpoint standar, endpoint FIPS menggunakan pustaka perangkat lunak TLS yang sesuai dengan FIPS 140-2. Titik akhir ini mungkin diperlukan oleh korporasi yang berinteraksi dengan pemerintah Amerika Serikat.

Jika pengaturan ini diaktifkan, tetapi titik akhir FIPS tidak ada untuk layanan di Anda Wilayah AWS, AWS perintah mungkin gagal. Dalam hal ini, tentukan secara manual titik akhir yang akan digunakan dalam perintah menggunakan [--endpoint-url](#) opsi atau gunakan titik akhir [khusus layanan](#).

Untuk informasi selengkapnya tentang menentukan titik akhir FIPS menurut Wilayah AWS, lihat Titik Akhir [FIPS](#) menurut Layanan.

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau dinyatakan secara eksplisit pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris [--endpoint-url](#) perintah.
2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.

5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan `profile` dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

### [web\\_identity\\_token\\_file](#)

Menentukan path ke file yang berisi token akses OAuth 2.0 atau token OpenID Connect ID yang disediakan oleh penyedia identitas. AWS CLI Memuat isi file ini dan meneruskannya sebagai `WebIdentityToken` argumen untuk `AssumeRoleWithWebIdentity` operasi.

Variabel lingkungan [AWS\\_WEB\\_IDENTITY\\_TOKEN\\_FILE](#) mengesampingkan pengaturan ini.

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called "Asumsikan peran dengan identitas web"](#).

### *tcp\_keepalive*

Menentukan apakah AWS CLI klien menggunakan paket TCP keep-alive.

Entri ini tidak memiliki variabel lingkungan yang setara atau opsi baris perintah.

```
tcp_keepalive = false
```

## Pengaturan perintah kustom S3

Amazon S3 mendukung beberapa pengaturan yang mengonfigurasi cara AWS CLI menjalankan operasi Amazon S3. Beberapa berlaku untuk semua perintah S3 di kedua `s3api` dan ruang `s3` nama. Lainnya khusus untuk perintah "kustom" S3 yang mengabstraksikan operasi umum dan melakukan lebih dari sekadar one-to-one pemetaan ke operasi API. Perintah `aws s3 transfercp`, `syncmv`, dan `rm` memiliki pengaturan tambahan yang dapat Anda gunakan untuk mengontrol transfer S3.

Semua opsi ini dapat dikonfigurasi dengan menentukan pengaturan `s3` bersarang di file `Andaconfig`. Setiap pengaturan kemudian menjorok pada garisnya sendiri.



**Note**

Pengaturan ini sepenuhnya opsional. Anda harus dapat berhasil menggunakan perintah `aws s3 transfer` tanpa mengonfigurasi pengaturan ini. Pengaturan ini disediakan untuk memungkinkan Anda menyetel kinerja atau memperhitungkan lingkungan tertentu tempat Anda menjalankan `aws s3` perintah ini.

Pengaturan ini semua diatur di bawah `s3` kunci tingkat atas dalam config file, seperti yang ditunjukkan pada contoh berikut untuk development profil.

```
[profile development]
s3 =
  max_concurrent_requests = 20
  max_queue_size = 10000
  multipart_threshold = 64MB
  multipart_chunksize = 16MB
  max_bandwidth = 50MB/s
  use_accelerate_endpoint = true
  addressing_style = path
```

Pengaturan berikut berlaku untuk perintah S3 apa pun di ruang `s3api` nama `s3` atau.

**addressing\_style**

Menentukan gaya pengalamatan yang akan digunakan. Ini mengontrol apakah nama bucket ada di nama host atau merupakan bagian dari URL. Nilai yang valid adalah: `path`, `virtual`, dan `auto`. Nilai default-nya adalah `auto`.

Ada dua gaya membangun endpoint Amazon S3. Yang pertama dipanggil `virtual` dan menyertakan nama bucket sebagai bagian dari nama host. Misalnya: `https://bucketname.s3.amazonaws.com`. Atau, dengan `path` gaya, Anda memperlakukan nama bucket seolah-olah itu adalah jalur di URI; misalnya, `https://s3.amazonaws.com/bucketname`. Nilai default dalam CLI adalah untuk menggunakan `auto`, yang mencoba menggunakan `virtual` gaya di mana ia bisa, tetapi akan kembali ke `path` gaya bila diperlukan. Misalnya, jika nama bucket Anda tidak kompatibel dengan DNS, nama bucket tidak dapat menjadi bagian dari nama host dan harus berada di jalur. Dengan `auto`, CLI akan mendeteksi kondisi ini dan secara otomatis beralih ke `path` gaya untuk Anda. Jika Anda

menyetel gaya pengalamatan `path`, Anda harus memastikan bahwa AWS Wilayah yang Anda konfigurasi AWS CLI cocok dengan Wilayah bucket Anda.

### **`payload_signing_enabled`**

Menentukan apakah untuk SHA256 menandatangani sigv4 payloads. Secara default, ini dinonaktifkan untuk streaming upload (`UploadPart` dan `PutObject`) saat menggunakan HTTPS. Secara default, ini diatur ke `false` untuk streaming upload (`UploadPart` dan `PutObject`), tetapi hanya jika `ContentMD5` ada (dihasilkan secara default) dan titik akhir menggunakan HTTPS.

Jika disetel ke `true`, permintaan S3 menerima validasi konten tambahan dalam bentuk checksum SHA256 yang dihitung untuk Anda dan disertakan dalam tanda tangan permintaan. Jika disetel ke `false`, checksum tidak dihitung. Menonaktifkan ini dapat berguna untuk mengurangi overhead kinerja yang dibuat oleh perhitungan checksum.

### **`use_accelerate_endpoint`**

Gunakan titik akhir Amazon S3 Accelerate untuk semua `s3` dan perintah. `s3api` Nilai default adalah `false`. Ini saling eksklusif dengan `use_dualstack_endpoint` pengaturan.

Jika disetel ke `true`, akan AWS CLI mengarahkan semua permintaan Amazon S3 ke titik akhir S3 Accelerate di `s3-accelerate.amazonaws.com`. Untuk menggunakan endpoint ini, Anda harus mengaktifkan bucket untuk digunakan S3 Accelerate. Semua permintaan dikirim menggunakan gaya virtual pengalamatan bucket: `my-bucket.s3-accelerate.amazonaws.com`. Setiap `ListBuckets`, `CreateBucket`, dan `DeleteBucket` permintaan tidak dikirim ke titik akhir S3 Accelerate karena titik akhir tersebut tidak mendukung operasi tersebut. Perilaku ini juga dapat diatur jika `--endpoint-url` parameter diatur ke `https://s3-accelerate.amazonaws.com` atau `http://s3-accelerate.amazonaws.com` untuk setiap `s3` atau `s3api` perintah.

Pengaturan berikut hanya berlaku untuk perintah di set perintah `s3` namespace.

### **`max_bandwidth`**

Menentukan bandwidth maksimum yang dapat dikonsumsi untuk mengunggah dan mengunduh data ke dan dari Amazon S3. Defaultnya tidak ada batas.

Ini membatasi bandwidth maksimum yang dapat digunakan perintah S3 untuk mentransfer data ke dan dari Amazon S3. Nilai ini hanya berlaku untuk unggahan dan unduhan; itu tidak berlaku untuk salinan atau penghapusan. Nilai dinyatakan sebagai byte per detik. Nilai dapat ditentukan sebagai:

- Sebuah bilangan bulat. Misalnya, 1048576 menetapkan penggunaan bandwidth maksimum menjadi 1 megabyte per detik.
- Sebuah integer diikuti oleh sufiks tingkat. Anda dapat menentukan sufiks tingkat menggunakan:KB/s,MB/s, atau. GB/s Misalnya,300KB/s,10MB/s.

Secara umum, kami menyarankan Anda terlebih dahulu mencoba menurunkan konsumsi bandwidth dengan menurunkan `max_concurrent_requests`. Jika itu tidak cukup membatasi konsumsi bandwidth ke tingkat yang diinginkan, Anda dapat menggunakan `max_bandwidth` pengaturan untuk membatasi konsumsi bandwidth lebih lanjut. Ini karena `max_concurrent_requests` mengontrol berapa banyak utas yang sedang berjalan. Jika Anda lebih dulu menurunkan `max_bandwidth` tetapi meninggalkan `max_concurrent_requests` pengaturan tinggi, itu dapat mengakibatkan utas harus menunggu yang tidak perlu. Hal ini dapat menyebabkan konsumsi sumber daya berlebih dan batas waktu koneksi.

### **max\_concurrent\_requests**

Menentukan jumlah maksimum permintaan bersamaan. Nilai default adalah 10.

Perintah `aws s3 transfer multithreaded`. Pada waktu tertentu, beberapa permintaan Amazon S3 dapat berjalan. Misalnya, ketika Anda menggunakan perintah `aws s3 cp localdir s3://bucket/ --recursive` untuk mengunggah file ke bucket S3, AWS CLI dapat mengunggah `filelocaldir/file1`, `localdir/file2`, dan `localdir/file3` secara paralel. Pengaturan `max_concurrent_requests` menentukan jumlah maksimum operasi transfer yang dapat berjalan pada saat yang sama.

Anda mungkin perlu mengubah nilai ini karena beberapa alasan:

- Mengurangi nilai ini — Pada beberapa lingkungan, default dari 10 permintaan bersamaan dapat membanjiri sistem. Ini dapat menyebabkan batas waktu koneksi atau memperlambat respons sistem. Menurunkan nilai ini membuat perintah transfer S3 kurang intensif sumber daya. Tradeoff adalah bahwa transfer S3 bisa memakan waktu lebih lama untuk diselesaikan. Menurunkan nilai ini mungkin diperlukan jika Anda menggunakan alat untuk membatasi bandwidth.
- Meningkatkan nilai ini — Dalam beberapa skenario, Anda mungkin ingin transfer Amazon S3 selesai secepat mungkin, menggunakan bandwidth jaringan sebanyak yang diperlukan. Dalam skenario ini, jumlah default permintaan bersamaan mungkin tidak cukup untuk menggunakan semua bandwidth jaringan yang tersedia. Meningkatkan nilai ini dapat meningkatkan waktu yang diperlukan untuk menyelesaikan transfer Amazon S3.

## **max\_queue\_size**

Menentukan jumlah maksimum tugas dalam antrian tugas. Nilai default adalah 1000.

AWS CLI Secara internal menggunakan model di mana ia mengantri tugas Amazon S3 yang kemudian dijalankan oleh konsumen yang jumlahnya dibatasi oleh.

`max_concurrent_requests` Sebuah tugas umumnya memetakan ke satu operasi Amazon S3. Misalnya, tugas bisa berupa `PutObjectTask`, atau `GetObjectTask`, atau `UploadPartTask`. Tingkat di mana tugas ditambahkan ke antrian bisa jauh lebih cepat daripada tingkat di mana konsumen menyelesaikan tugas. Untuk menghindari pertumbuhan tak terbatas, ukuran antrian tugas dibatasi ke ukuran tertentu. Pengaturan ini mengubah nilai angka maksimum itu.

Anda biasanya tidak perlu mengubah pengaturan ini. Pengaturan ini juga sesuai dengan jumlah tugas yang AWS CLI disadari perlu dijalankan. Ini berarti bahwa secara default hanya AWS CLI dapat melihat 1000 tugas di depan. Meningkatkan nilai ini berarti bahwa AWS CLI dapat lebih cepat mengetahui jumlah total tugas yang dibutuhkan, dengan asumsi bahwa tingkat antrian lebih cepat daripada tingkat penyelesaian tugas. Penggantiannya adalah `max_queue_size` yang lebih besar membutuhkan lebih banyak memori.

## **multipart\_chunksize**

Menentukan ukuran potongan yang AWS CLI digunakan untuk transfer multipart file individual. Nilai default adalah 8 MB, dengan minimal 5 MB.

Ketika transfer file melebihi `multipart_threshold`, AWS CLI membagi file menjadi potongan-potongan ukuran ini. Nilai ini dapat ditentukan menggunakan sintaks yang sama seperti `multipart_threshold`, baik sebagai jumlah byte sebagai integer, atau dengan menggunakan ukuran dan akhiran.

## **multipart\_threshold**

Menentukan ambang batas ukuran yang AWS CLI digunakan untuk transfer multipart file individual. Nilai default adalah 8 MB.

Saat mengunggah, mengunduh, atau menyalin file, perintah Amazon S3 beralih ke operasi multibagian jika file melebihi ukuran ini. Anda dapat menentukan nilai ini dengan salah satu dari dua cara:

- Ukuran file dalam byte. Misalnya, `1048576`.
- Ukuran file dengan akhiran ukuran. Anda dapat menggunakan `KB`, `MB`, `GB`, atau `TB`.  
Misalnya: `10MB`, `1GB`.

**Note**

S3 dapat memaksakan kendala pada nilai valid yang dapat digunakan untuk operasi multipart. Untuk informasi selengkapnya, lihat [dokumentasi Unggahan Multipart S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

## Variabel lingkungan untuk mengkonfigurasi AWS CLI

Variabel lingkungan menyediakan cara lain untuk menentukan opsi konfigurasi dan kredensial, dan dapat berguna untuk skrip atau pengaturan sementara profil bernama sebagai default.

### Prioritas opsi

- Jika Anda menentukan opsi dengan menggunakan salah satu variabel lingkungan yang dijelaskan dalam topik ini, itu akan mengganti nilai apa pun yang dimuat dari profil dalam file konfigurasi.
- Jika Anda menentukan opsi dengan menggunakan parameter pada baris AWS CLI perintah, itu akan mengganti nilai apa pun dari variabel lingkungan yang sesuai atau profil dalam file konfigurasi.

Untuk informasi selengkapnya tentang prioritas dan cara AWS CLI menentukan kredensial mana yang akan digunakan, lihat. [Konfigurasi AWS CLI](#)

### Topik

- [Cara mengatur variabel lingkungan](#)
- [AWS CLI variabel lingkungan yang didukung](#)

## Cara mengatur variabel lingkungan

Contoh berikut menunjukkan bagaimana Anda dapat mengkonfigurasi variabel lingkungan untuk pengguna default.

### Linux or macOS

```
$ export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
$ export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

```
$ export AWS_DEFAULT_REGION=us-west-2
```

Menyetel variabel lingkungan mengubah nilai yang digunakan hingga akhir sesi shell Anda, atau sampai Anda menyetel variabel ke nilai yang berbeda. Anda dapat membuat variabel persisten di seluruh sesi masa depan dengan menyetelnya di skrip startup shell Anda.

## Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE  
C:\> setx AWS_SECRET_ACCESS_KEY wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY  
C:\> setx AWS_DEFAULT_REGION us-west-2
```

Menggunakan [setx](#) untuk mengatur variabel lingkungan mengubah nilai yang digunakan dalam sesi prompt perintah saat ini dan semua sesi prompt perintah yang Anda buat setelah menjalankan perintah. Itu tidak mempengaruhi shell perintah lain yang sudah berjalan pada saat Anda menjalankan perintah. Anda mungkin perlu me-restart terminal Anda agar pengaturan dimuat.

Untuk mengatur untuk sesi saat ini saja

Menggunakan [set](#) untuk mengatur variabel lingkungan mengubah nilai yang digunakan sampai akhir sesi prompt perintah saat ini, atau sampai Anda mengatur variabel ke nilai yang berbeda.

```
C:\> set AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE  
C:\> set AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY  
C:\> set AWS_DEFAULT_REGION=us-west-2
```

## PowerShell

```
PS C:\> $Env:AWS_ACCESS_KEY_ID="AKIAIOSFODNN7EXAMPLE"  
PS C:\> $Env:AWS_SECRET_ACCESS_KEY="wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"  
PS C:\> $Env:AWS_DEFAULT_REGION="us-west-2"
```

Jika Anda menetapkan variabel lingkungan pada PowerShell prompt seperti yang ditunjukkan pada contoh sebelumnya, itu menyimpan nilai hanya untuk durasi sesi saat ini. Untuk membuat pengaturan variabel lingkungan persisten di semua sesi PowerShell Command Prompt, simpan dengan menggunakan aplikasi Sistem di Control Panel. Atau, Anda dapat mengatur variabel untuk semua PowerShell sesi future dengan menambahkannya ke PowerShell profil Anda. Lihat

[PowerShell dokumentasi](#) untuk informasi selengkapnya tentang menyimpan variabel lingkungan atau mempertahankannya di seluruh sesi.

## AWS CLI variabel lingkungan yang didukung

AWS CLI Mendukung variabel lingkungan berikut.

### **AWS\_ACCESS\_KEY\_ID**

Menentukan kunci AWS akses yang terkait dengan akun IAM.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil. `aws_access_key_id` Anda tidak dapat menentukan ID kunci akses dengan menggunakan opsi baris perintah.

### **AWS\_CA\_BUNDLE**

Menentukan jalur ke bundel sertifikat yang akan digunakan untuk validasi sertifikat HTTPS.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil. [ca\\_bundle](#) Anda dapat mengganti variabel lingkungan ini dengan menggunakan parameter baris [--ca-bundle](#) perintah.

### **AWS\_CLI\_AUTO\_PROMPT**

Mengaktifkan prompt otomatis untuk AWS CLI versi 2. Ada dua pengaturan yang dapat digunakan:

- **on** menggunakan mode auto-prompt penuh setiap kali Anda mencoba menjalankan `aws` perintah. Ini termasuk menekan ENTER setelah perintah lengkap atau perintah tidak lengkap.
- **on-partial** menggunakan mode auto-prompt sebagian. Jika perintah tidak lengkap atau tidak dapat dijalankan karena kesalahan validasi sisi klien, prompt otomatis digunakan. Mode ini berguna jika Anda memiliki skrip, runbook, atau Anda hanya ingin diminta secara otomatis untuk perintah yang tidak Anda kenal daripada diminta pada setiap perintah.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan [cli\\_auto\\_prompt](#) profil. Anda dapat mengganti variabel lingkungan ini dengan menggunakan parameter baris [--no-cli-auto-prompt](#) perintah [--cli-auto-prompt](#) dan.

Untuk informasi tentang fitur prompt otomatis AWS CLI versi 2, lihat [AWS CLIMinta Anda meminta perintah](#).

## AWS\_CLI\_FILE\_ENCODING

Menentukan pengkodean yang digunakan untuk file teks. Secara default encoding cocok dengan lokal Anda. Untuk mengatur pengkodean berbeda dari lokal, gunakan variabel `aws_cli_file_encoding` lingkungan. Misalnya, jika Anda menggunakan Windows dengan pengkodean default CP1252, pengaturan `aws_cli_file_encoding=UTF-8` menetapkan CLI untuk membuka file teks menggunakan UTF-8.

## AWS\_CLI\_S3\_MV\_VALIDATE\_SAME\_S3\_PATHS

Jika bucket sumber dan tujuan sama saat menggunakan kustom `s3 mv` perintah, file sumber atau objek dapat dipindahkan ke dirinya sendiri, yang dapat mengakibatkan penghapusan file sumber atau objek Anda secara tidak sengaja. Variabel dan `--validate-same-s3-paths` opsi `AWS_CLI_S3_MV_VALIDATE_SAME_S3_PATHS` lingkungan menentukan apakah akan memvalidasi ARN titik akses atau alias titik akses Anda di URI sumber atau tujuan Amazon S3 Anda.

### Note

Validasi jalur untuk `s3 mv` membutuhkan panggilan API tambahan.

## AWS\_CONFIG\_FILE

Menentukan lokasi file yang AWS CLI digunakan untuk menyimpan profil konfigurasi. Jalur defaultnya adalah `~/.aws/config`.

Anda tidak dapat menentukan nilai ini dalam pengaturan profil bernama atau dengan menggunakan parameter baris perintah.

## AWS\_DATA\_PATH

Daftar direktori tambahan untuk memeriksa di luar jalur pencarian bawaan `~/.aws/models` saat memuat AWS CLI data. Menyetel variabel lingkungan ini menunjukkan direktori tambahan untuk diperiksa terlebih dahulu sebelum kembali ke jalur pencarian bawaan. Beberapa entri harus dipisahkan dengan `os.pathsep` karakter, yang ada `:` di Linux atau macOS ; dan di Windows.

## AWS\_DEFAULT\_OUTPUT

Menentukan [format output](#) untuk digunakan.



Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil. `output` Anda dapat mengganti variabel lingkungan ini dengan menggunakan parameter baris `--output` perintah.

## **AWS\_DEFAULT\_REGION**

`Default region` name ini mengidentifikasi AWS Wilayah yang servernya ingin Anda kirim permintaan Anda secara default. Ini biasanya Wilayah yang paling dekat dengan Anda, tetapi bisa berupa Wilayah mana pun. Misalnya, Anda dapat mengetik `us-west-2` untuk menggunakan US West (Oregon). Ini adalah Wilayah tempat semua permintaan selanjutnya dikirim, kecuali jika Anda menentukan sebaliknya dalam perintah individual.

### Note

Anda harus menentukan AWS Region saat menggunakan AWS CLI, baik secara eksplisit atau dengan menyetel Region default. Untuk daftar Wilayah yang tersedia, lihat [Wilayah dan Titik Akhir](#). Penunjuk Wilayah yang digunakan oleh AWS CLI adalah nama yang sama dengan yang Anda lihat di AWS Management Console URL dan titik akhir layanan.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil. `region` Anda dapat mengganti variabel lingkungan ini dengan menggunakan parameter baris `--region` perintah dan variabel `AWS_REGION` lingkungan yang kompatibel dengan AWS SDK.

## **AWS\_EC2\_METADATA\_DISABLED**

Menonaktifkan penggunaan layanan metadata instans Amazon EC2 (IMDS).

Jika disetel ke `true`, kredensial atau konfigurasi pengguna (seperti Wilayah) tidak diminta dari IMDS.

## **AWS\_ENDPOINT\_URL**

Menentukan endpoint yang digunakan untuk semua permintaan layanan.

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dideklarasikan pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris `--endpoint-url` perintah.

2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan profile dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

### **AWS\_ENDPOINT\_URL\_<SERVICE>**

Menentukan endpoint kustom yang digunakan untuk layanan tertentu, di mana `<SERVICE>` diganti dengan identifier. Layanan AWS Misalnya, Amazon DynamoDB memiliki a `serviceId` dari [DynamoDB](#). Untuk layanan ini, variabel lingkungan URL endpoint adalah `AWS_ENDPOINT_URL_DYNAMODB`.

Untuk daftar semua variabel lingkungan khusus layanan, lihat. [Daftar pengidentifikasi khusus layanan](#)

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dideklarasikan pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris [--endpoint-url](#) perintah.
2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.

5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan `profile` dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## AWS\_IGNORE\_CONFIGURED\_ENDPOINT\_URLS

Jika diaktifkan, AWS CLI mengabaikan semua konfigurasi titik akhir kustom. Nilai yang valid adalah **true** dan **false**.

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dideklarasikan pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris [--endpoint-url](#) perintah.
2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan `profile` dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## **AWS\_MAX\_ATTEMPTS**

Menentukan nilai percobaan ulang maksimum yang digunakan oleh penangan AWS CLI coba lagi, di mana panggilan awal diperhitungkan terhadap nilai yang Anda berikan. Untuk informasi lebih lanjut tentang percobaan ulang, lihat [AWS CLImencoba lagi](#).

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.

`max_attempts`

## **AWS\_METADATA\_SERVICE\_NUM\_ATTEMPTS**

Saat mencoba mengambil kredensial pada instans Amazon EC2 yang telah dikonfigurasi dengan peran IAM, AWS CLI upaya untuk mengambil kredensial sekali dari layanan metadata instans sebelum berhenti. Jika Anda tahu perintah Anda akan berjalan pada instans Amazon EC2, Anda dapat meningkatkan nilai ini untuk AWS CLI mencoba lagi beberapa kali sebelum menyerah.

## **AWS\_METADATA\_SERVICE\_TIMEOUT**

Jumlah detik sebelum koneksi ke layanan metadata instance harus habis. Saat mencoba mengambil kredensial pada instans Amazon EC2 yang dikonfigurasi dengan peran IAM, koneksi ke layanan metadata instans akan habis setelah 1 detik secara default. Jika Anda tahu bahwa Anda menjalankan instans Amazon EC2 dengan peran IAM yang dikonfigurasi, Anda dapat meningkatkan nilai ini jika diperlukan.

## **AWS\_PAGER**

Menentukan program pager yang digunakan untuk output. Secara default, AWS CLI versi 2 mengembalikan semua output melalui program pager default sistem operasi Anda.

Untuk menonaktifkan semua penggunaan program paging eksternal, atur variabel ke string kosong.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.

`cli_pager`

## **AWS\_PROFILE**

Menentukan nama AWS CLI profil dengan kredensial dan pilihan untuk digunakan. Ini bisa berupa nama profil yang disimpan dalam config file `credentials` atau, atau nilai default untuk menggunakan profil default.

Jika didefinisikan, variabel lingkungan ini mengesampingkan perilaku menggunakan profil bernama `[default]` dalam file konfigurasi. Anda dapat mengganti variabel lingkungan ini dengan menggunakan parameter baris `--profile` perintah.

## AWS\_REGION

Variabel lingkungan yang kompatibel dengan AWS SDK yang menentukan AWS Region untuk mengirim permintaan.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai dalam variabel lingkungan `AWS_DEFAULT_REGION` dan pengaturan profil. `region` Anda dapat mengganti variabel lingkungan ini dengan menggunakan parameter baris `--region` perintah.

## AWS\_RETRY\_MODE

Menentukan modus AWS CLI coba lagi yang menggunakan. Ada tiga mode coba lagi yang tersedia: legacy (default), standar, dan adaptif. Untuk informasi lebih lanjut tentang percobaan ulang, lihat [AWS CLI mencoba lagi](#).

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.  
`retry_mode`

## AWS\_ROLE\_ARN

Menentukan Nama Sumber Daya Amazon (ARN) peran IAM dengan penyedia identitas web yang ingin Anda gunakan untuk menjalankan perintah. AWS CLI

Digunakan dengan variabel `AWS_WEB_IDENTITY_TOKEN_FILE` dan `AWS_ROLE_SESSION_NAME` lingkungan.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil. [role\\_arn](#) Anda tidak dapat menentukan nama sesi peran sebagai parameter baris perintah.

### Note

Variabel lingkungan ini hanya berlaku untuk peran yang diasumsikan dengan penyedia identitas web yang tidak berlaku untuk konfigurasi penyedia peran asumsi umum.

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called "Asumsikan peran dengan identitas web"](#).

## AWS\_ROLE\_SESSION\_NAME

Menentukan nama untuk melampirkan ke sesi peran. Nilai ini disediakan untuk `RoleSessionName` parameter ketika AWS CLI panggilan `AssumeRole`

operasi, dan menjadi bagian dari peran yang diasumsikan pengguna ARN:


`arn:aws:sts::123456789012:assumed-role/role_name/role_session_name` Ini adalah parameter opsional. Jika Anda tidak memberikan nilai ini, nama sesi dihasilkan secara otomatis. Nama ini muncul di AWS CloudTrail log untuk entri yang terkait dengan sesi ini.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.

[role\\_session\\_name](#)

Digunakan dengan variabel `AWS_ROLE_ARN` dan `AWS_WEB_IDENTITY_TOKEN_FILE` lingkungan.

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called “Asumsikan peran dengan identitas web”](#).

 Note

Variabel lingkungan ini hanya berlaku untuk peran yang diasumsikan dengan penyedia identitas web yang tidak berlaku untuk konfigurasi penyedia peran asumsi umum.

## **AWS\_SECRET\_ACCESS\_KEY**

Menentukan kunci rahasia yang terkait dengan kunci akses. Ini pada dasarnya adalah “kata sandi” untuk kunci akses.

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.

`aws_secret_access_key` Anda tidak dapat menentukan ID kunci akses rahasia sebagai opsi baris perintah.

## **AWS\_SESSION\_TOKEN**

Menentukan nilai token sesi yang diperlukan jika Anda menggunakan kredensial keamanan sementara yang Anda ambil langsung dari operasi. AWS STS Untuk informasi selengkapnya, lihat [bagian Output dari perintah assume-role di Command Reference](#).AWS CLI

Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.

`aws_session_token`

## **AWS\_SHARED\_CREDENTIALS\_FILE**

Menentukan lokasi file yang AWS CLI digunakan untuk menyimpan kunci akses. Jalur defaultnya adalah `~/.aws/credentials`.

Anda tidak dapat menentukan nilai ini dalam pengaturan profil bernama atau dengan menggunakan parameter baris perintah.

## **AWS\_USE\_DUALSTACK\_ENDPOINT**

Mengaktifkan penggunaan titik akhir dual-stack untuk mengirim permintaan. AWS Untuk mempelajari lebih lanjut tentang titik akhir dual-stack, yang mendukung lalu lintas IPv4 dan IPv6, lihat Menggunakan titik akhir [tumpukan ganda Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Titik akhir dual-stack tersedia untuk beberapa layanan di beberapa wilayah. Jika titik akhir tumpukan ganda tidak ada untuk layanan atau Wilayah AWS, permintaan gagal. Ini dinonaktifkan secara default.

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dideklarasikan pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris [--endpoint-url](#) perintah.
2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti. [AWS\\_ENDPOINT\\_URL\\_DYNAMODB](#)
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint-url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint-url](#) pengaturan profile dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint-url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## **AWS\_USE\_FIPS\_ENDPOINT**

Beberapa AWS layanan menawarkan titik akhir yang mendukung [Federal Information Processing Standard \(FIPS\) 140-2](#) di beberapa Wilayah AWS. Ketika AWS layanan mendukung FIPS, pengaturan ini menentukan titik akhir FIPS apa yang harus digunakan. AWS CLI Tidak seperti AWS endpoint standar, endpoint FIPS menggunakan pustaka perangkat lunak TLS yang sesuai

dengan FIPS 140-2. Titik akhir ini mungkin diperlukan oleh korporasi yang berinteraksi dengan pemerintah Amerika Serikat.

Jika pengaturan ini diaktifkan, tetapi titik akhir FIPS tidak ada untuk layanan di Anda Wilayah AWS, AWS perintah mungkin gagal. Dalam hal ini, tentukan secara manual titik akhir yang akan digunakan dalam perintah menggunakan [--endpoint-url](#) opsi atau gunakan titik akhir [khusus layanan](#).

Untuk informasi selengkapnya tentang menentukan titik akhir FIPS menurut Wilayah AWS, lihat Titik Akhir [FIPS](#) menurut Layanan.

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dideklarasikan pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris [--endpoint-url](#) perintah.
2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti. [AWS\\_ENDPOINT\\_URL\\_DYNAMODB](#)
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan profile dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## [AWS\\_WEB\\_IDENTITY\\_TOKEN\\_FILE](#)


Menentukan path ke file yang berisi token akses OAuth 2.0 atau token OpenID Connect ID yang disediakan oleh penyedia identitas. AWS CLI Memuat isi file ini dan meneruskannya sebagai `WebIdentityToken` argumen untuk `AssumeRoleWithWebIdentity` operasi.

Digunakan dengan variabel `AWS_ROLE_ARN` dan `AWS_ROLE_SESSION_NAME` lingkungan.



Jika didefinisikan, variabel lingkungan ini mengesampingkan nilai untuk pengaturan profil.  
`web_identity_token_file`

Untuk informasi selengkapnya tentang penggunaan identitas web, lihat [the section called “Asumsikan peran dengan identitas web”](#).

 Note

Variabel lingkungan ini hanya berlaku untuk peran yang diasumsikan dengan penyedia identitas web yang tidak berlaku untuk konfigurasi penyedia peran asumsi umum.

## Opsi baris perintah

DiAWS CLI, opsi baris perintah adalah parameter global yang dapat Anda gunakan untuk mengganti pengaturan konfigurasi default, pengaturan profil yang sesuai, atau pengaturan variabel lingkungan untuk perintah tunggal itu. Anda tidak dapat menggunakan opsi baris perintah untuk menentukan kredensial secara langsung, meskipun Anda dapat menentukan profil mana yang akan digunakan.

### Topik

- [Cara menggunakan opsi baris perintah](#)
- [AWS CLImendukung opsi baris perintah global](#)
- [Penggunaan umum opsi baris perintah](#)

## Cara menggunakan opsi baris perintah

Sebagian besar opsi baris perintah adalah string sederhana, seperti nama profil `profile1` dalam contoh berikut:

```
$ aws s3 ls --profile profile1
example-bucket-1
example-bucket-2
...
```

Setiap opsi yang mengambil argumen memerlukan spasi atau sama dengan tanda (=) yang memisahkan argumen dari nama opsi. Jika nilai argumen adalah string yang berisi spasi, Anda harus menggunakan tanda kutip di sekitar argumen. Untuk detail tentang tipe argumen dan pemformatan parameter, lihat [Tentukan nilai parameter untuk AWS CLI](#).

## AWS CLImendukung opsi baris perintah global

Di dalam AWS CLI Anda dapat menggunakan opsi baris perintah berikut untuk mengganti pengaturan konfigurasi default, pengaturan profil yang sesuai, atau pengaturan variabel lingkungan untuk perintah tunggal itu.

`--ca-bundel <string>`

Menentukan bundel sertifikat otoritas sertifikat (CA) yang akan digunakan saat memverifikasi sertifikat SSL.

Jika ditentukan, opsi ini mengesampingkan nilai untuk pengaturan profil [ca\\_bundle](#) dan variabel [AWS\\_CA\\_BUNDLE](#) lingkungan.

`--cli-auto-prompt`

Mengaktifkan mode auto-prompt untuk satu perintah. Seperti yang ditunjukkan contoh berikut, Anda dapat menentukannya kapan saja.

```
$ aws --cli-auto-prompt
$ aws dynamodb --cli-auto-prompt
$ aws dynamodb describe-table --cli-auto-prompt
```

Opsi ini mengesampingkan variabel [aws\\_cli\\_auto\\_prompt](#) lingkungan dan pengaturan [cli\\_auto\\_prompt](#) profil.

Untuk informasi tentang fitur prompt otomatis AWS CLI versi 2, lihat [AWS CLIMinta Anda meminta perintah](#).

`--cli-binary-format`

Menentukan bagaimana AWS CLI versi 2 menafsirkan parameter input biner. Ini bisa menjadi salah satu nilai berikut:

- `base64` - Ini adalah nilai default. Parameter input yang diketik sebagai objek besar biner (BLOB) menerima string yang dikodekan base64. Untuk meneruskan konten biner yang benar, letakkan konten dalam file dan berikan path dan nama file dengan `fileb://` awalan sebagai nilai parameter. Untuk meneruskan teks berencode base64 yang terkandung dalam file, berikan path dan nama file dengan `file://` awalan sebagai nilai parameter.
- `raw-in-base64-out` - Default untuk AWS CLI versi 1. Jika nilai pengaturan adalah `raw-in-base64-out`, file yang direferensikan menggunakan `file://` awalan dibaca sebagai teks dan kemudian AWS CLI upaya untuk menyandikannya ke biner.

Ini mengesampingkan pengaturan konfigurasi [cli\\_binary\\_format](#) file.

```
$ aws lambda invoke \  
  --cli-binary-format raw-in-base64-out \  
  --function-name my-function \  
  --invocation-type Event \  
  --payload '{ "name": "Bob" }' \  
  response.json
```

Jika Anda mereferensikan nilai biner dalam file menggunakan notasi `fileb://` awalan, AWS CLI selalu mengharapkan file tersebut berisi konten biner mentah dan tidak mencoba mengonversi nilainya.

Jika Anda mereferensikan nilai biner dalam file menggunakan notasi `file://` awalan, AWS CLI menangani file sesuai dengan pengaturan saat ini `cli_binary_format`. Jika nilai pengaturan itu `base64` (default saat tidak disetel secara eksplisit), file tersebut AWS CLI mengharapkan file berisi teks yang dikodekan `base64`. Jika nilai pengaturan itu `raw-in-base64-out`, file AWS CLI mengharapkan berisi konten biner mentah.

`--cli-connect-timeout <integer>`

Menentukan waktu koneksi soket maksimum dalam hitungan detik. Jika nilainya disetel ke nol (0), sambungan soket menunggu tanpa batas waktu (memblokir) dan tidak batas waktu.

`--cli-read-timeout <integer>`

Menentukan waktu baca soket maksimum dalam hitungan detik. Jika nilai disetel ke nol (0) pembacaan soket menunggu tanpa batas waktu (memblokir) dan tidak batas waktu.

`--warna <string>`

Menentukan dukungan untuk output warna. Nilai yang valid adalah `on`, `off`, dan `auto`. Nilai default-nya adalah `auto`.

`--debug`

Sakelar Boolean yang memungkinkan logging debug. Secara AWS CLI default menyediakan informasi yang dibersihkan mengenai keberhasilan atau kegagalan terkait hasil perintah dalam output perintah. `--debug` Opsi ini menyediakan log Python lengkap. Ini termasuk informasi `stderr` diagnostik tambahan tentang pengoperasian perintah yang dapat berguna saat memecahkan masalah mengapa perintah memberikan hasil yang tidak terduga. Untuk melihat log debug dengan mudah, kami sarankan mengirim log ke file agar lebih mudah mencari informasi. Anda dapat melakukan ini dengan menggunakan salah satu dari berikut ini.

Untuk mengirim hanya informasi `stderr` diagnostik, tambahkan `2> debug.txt` di mana nama `debug.txt` yang ingin Anda gunakan untuk file debug Anda:

```
$ aws servicename commandname options --debug 2> debug.txt
```

Untuk mengirim informasi output dan `stderr` diagnostik, tambahkan `&> debug.txt` di mana nama `debug.txt` yang ingin Anda gunakan untuk file debug Anda:

```
$ aws servicename commandname options --debug &> debug.txt
```

--titik akhir-url <string>

Menentukan URL untuk mengirim permintaan ke. Untuk sebagian besar perintah, AWS CLI secara otomatis menentukan URL berdasarkan layanan yang dipilih dan AWS Wilayah yang ditentukan. Namun, beberapa perintah mengharuskan Anda menentukan URL khusus akun. Anda juga dapat mengonfigurasi beberapa AWS layanan untuk [meng-host titik akhir langsung di dalam VPC pribadi Anda](#), yang mungkin perlu ditentukan.

Contoh perintah berikut menggunakan URL endpoint Amazon S3 kustom.

```
$ aws s3 ls --endpoint-url http://localhost:4567
```

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau dinyatakan secara eksplisit pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris `--endpoint-url` perintah.
2. Jika diaktifkan, variabel lingkungan endpoint `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global atau pengaturan profil `ignore_configure_endpoint_urls` untuk mengabaikan endpoint kustom.
3. Nilai yang diberikan oleh variabel lingkungan khusus layanan `AWS_ENDPOINT_URL_<SERVICE>`, seperti. `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT`, dan variabel `AWS_ENDPOINT_URL` lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh `endpoint_url` pengaturan dalam `services` bagian file bersama. `config`

6. Nilai yang diberikan oleh `endpoint_url` pengaturan profile dalam config file bersama.
7. `use_dualstack_endpoint`, `use_fips_endpoint`, dan `endpoint_url` pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

`--no-cli-auto-prompt`

Menonaktifkan mode prompt otomatis untuk satu perintah.

```
$ aws dynamodb describe-table --table-name Table1 --no-cli-auto-prompt
```

Opsi ini mengesampingkan variabel `aws_cli_auto_prompt` lingkungan dan pengaturan `cli_auto_prompt` profil.

Untuk informasi tentang fitur prompt otomatis AWS CLI versi 2, lihat [AWS CLI Minta Anda meminta perintah](#).

`--no-cli-pager`

Sakelar Boolean yang menonaktifkan menggunakan pager untuk output perintah.

`--no-paginate`


Sakelar Boolean yang menonaktifkan beberapa panggilan yang dilakukan secara otomatis AWS CLI untuk menerima semua hasil perintah yang membuat pagination output. Ini berarti hanya halaman pertama output Anda yang ditampilkan.

`--no-sign-request`

Sakelar Boolean yang menonaktifkan penandatanganan permintaan HTTP ke titik akhir AWS layanan. Ini mencegah kredensial dimuat.

`--no-verify-ssl`

Secara default, AWS CLI menggunakan SSL saat berkomunikasi dengan AWS layanan. Untuk setiap koneksi SSL dan panggilan, AWS CLI memverifikasi sertifikat SSL. Menggunakan opsi ini mengesampingkan perilaku default untuk memverifikasi sertifikat SSL.

 Warning

Opsi ini bukan praktik terbaik. Jika Anda menggunakannya `--no-verify-ssl`, lalu lintas antara klien dan AWS layanan Anda tidak lagi aman. Ini berarti lalu lintas Anda berisiko

keamanan dan rentan terhadap man-in-the-middle eksploitasi. Jika Anda mengalami masalah dengan sertifikat, sebaiknya selesaikan masalah tersebut. Untuk langkah pemecahan masalah sertifikat, lihat [the section called “Kesalahan sertifikat SSL”](#)

`--keluaran <string>`

Menentukan format output yang akan digunakan untuk perintah ini. Anda dapat menentukan salah satu nilai berikut:

- **`json`**— Output diformat sebagai string [JSON](#).
- **`yaml`**— Output diformat sebagai string [YAMG](#).
- **`yaml-stream`**— Output dialirkan dan diformat sebagai string [YAMB](#). Streaming memungkinkan penanganan tipe data besar yang lebih cepat.
- **`text`**- Output diformat sebagai beberapa baris nilai string yang dipisahkan tab. Ini dapat berguna untuk meneruskan output ke prosesor teks, seperti `grep`, `sed`, atau `awk`.
- **`table`**— Output diformat sebagai tabel menggunakan karakter `+|-` untuk membentuk batas sel. Ini biasanya menyajikan informasi dalam format “ramah manusia” yang jauh lebih mudah dibaca daripada yang lain, tetapi tidak berguna secara terprogram.

`--profil <string>`

Menentukan [profil bernama](#) untuk digunakan untuk perintah ini. Untuk mengatur profil bernama tambahan, Anda dapat menggunakan `aws configure` perintah dengan `--profile` opsi.

```
$ aws configure --profile <profilename>
```

`--kueri <string>`

Menentukan [query JMESPath](#) untuk digunakan dalam penyaringan data respon. Untuk informasi selengkapnya, lihat [Filter AWS CLI keluaran](#).

`--wilayah <string>`

Menentukan AWS Region mana untuk mengirim AWS permintaan perintah ini ke. Untuk daftar semua Wilayah yang dapat Anda tentukan, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services.

`--versi`

Switch Boolean yang menampilkan versi saat ini dari AWS CLI program yang sedang berjalan.

## Penggunaan umum opsi baris perintah

Penggunaan umum untuk opsi baris perintah termasuk memeriksa sumber daya Anda di beberapa AWS Wilayah, dan mengubah format output untuk keterbacaan atau kemudahan penggunaan saat membuat skrip. Dalam contoh berikut, kita menjalankan perintah `describe-instance` terhadap setiap Region sampai kita menemukan Region mana instance kita berada.

```
$ aws ec2 describe-instances --output table --region us-west-1
-----
|DescribeInstances|
+-----+
$ aws ec2 describe-instances --output table --region us-west-2
-----
|
|           DescribeInstances           |
+-----+
||
||           Reservations               ||
|+-----+|
||  OwnerId          |  012345678901          ||
||  ReservationId    |  r-abcdefgh          ||
|+-----+|
||| | | | |
|||           Instances                 |||
||+-----+||
|||  AmiLaunchIndex   |  0                          |||
|||  Architecture     |  x86_64                          |||
...

```

## Penyelesaian perintah

The AWS Command Line Interface (AWS CLI) menyertakan fitur penyelesaian perintah yang kompatibel dengan bash yang memungkinkan Anda menggunakan tombol Tab untuk menyelesaikan perintah yang dimasukkan sebagian. Pada sebagian besar sistem Anda perlu mengkonfigurasi ini secara manual.

Untuk informasi tentang fitur auto-prompt AWS CLI versi 2 sebagai gantinya, lihat [AWS CLIMinta Anda meminta perintah](#).

Topik

- [Cara kerjanya](#)
- [Mengkonfigurasi penyelesaian perintah di Linux atau macOS](#)

- [Mengkonfigurasi penyelesaian perintah pada Windows](#)

## Cara kerjanya

Saat Anda memasukkan sebagian perintah, parameter, atau opsi, fitur penyelesaian perintah secara otomatis melengkapi perintah Anda atau menampilkan daftar perintah yang disarankan. Untuk meminta penyelesaian perintah, Anda memasukkan sebagian perintah dan menekan tombol penyelesaian, yang biasanya *Tab* di sebagian besar shell.

Contoh berikut menunjukkan berbagai cara yang dapat Anda gunakan penyelesaian perintah:

- Masukkan sebagian perintah dan tekan *Tab* untuk menampilkan daftar perintah yang disarankan.

```
$ aws dynamodb dTAB
delete-backup                describe-global-table
delete-item                  describe-global-table-settings
delete-table                 describe-limits
describe-backup              describe-table
describe-continuous-backups describe-table-replica-auto-scaling
describe-contributor-insights describe-time-to-live
describe-endpoints
```

- Masukkan sebagian parameter dan tekan *Tab* untuk menampilkan daftar parameter yang disarankan.

```
$ aws dynamodb delete-table --TAB
--ca-bundle                --endpoint-url          --profile
--cli-connect-timeout      --generate-cli-skeleton --query
--cli-input-json           --no-paginate           --region
--cli-read-timeout         --no-sign-request      --table-name
--color                    --no-verify-ssl        --version
--debug                    --output
```

- Masukkan parameter dan tekan *Tab* untuk menampilkan daftar nilai sumber daya yang disarankan. Fitur ini hanya tersedia di AWS CLI versi 2.

```
$ aws dynamodb db delete-table --table-name TAB
Table 1                    Table 2                    Table 3
```



## Mengkonfigurasi penyelesaian perintah di Linux atau macOS

Untuk mengonfigurasi penyelesaian perintah di Linux atau macOS, Anda harus mengetahui nama shell yang Anda gunakan dan lokasi skrip. `aws_completer`

### Note

Penyelesaian perintah secara otomatis dikonfigurasi dan diaktifkan secara default pada instans Amazon EC2 yang menjalankan Amazon Linux.

### Topik

- [Konfirmasikan folder pelengkap ada di jalur Anda](#)
- [Aktifkan penyelesaian perintah](#)
- [Verifikasi penyelesaian perintah](#)

### Konfirmasikan folder pelengkap ada di jalur Anda

Agar AWS pelengkap berhasil bekerja, `aws_completer` kebutuhan harus berada di jalur shell Anda. `which` Perintah dapat memeriksa apakah pelengkap ada di jalur Anda.

```
$ which aws_completer
/usr/local/bin/aws_completer
```

Jika perintah yang tidak dapat menemukan pelengkap, maka gunakan langkah-langkah berikut untuk menambahkan folder pelengkap ke jalur Anda.

### Langkah 1: Temukan AWS pelengkap

Lokasi AWS pelengkap dapat bervariasi tergantung pada metode instalasi yang digunakan.

- Package Manager - Program seperti `pipyum`, `brew`, dan `apt-get` biasanya menginstal AWS completer (atau symlink ke sana) ke lokasi jalur standar.
  - Jika Anda menggunakan `pip` tanpa `--user` parameter, jalur defaultnya adalah `/usr/local/bin/aws_completer`.
  - Jika Anda menggunakan `pip --user` parameter, jalur defaultnya adalah `home/username/.local/bin/aws_completer`.

- Penginstal yang Dibundel - Jika Anda menggunakan penginstal yang dibundel, jalur defaultnya adalah `/usr/local/bin/aws_completer`

Jika semuanya gagal, Anda dapat menggunakan `find` perintah untuk mencari sistem file Anda untuk AWS pelengkap.

```
$ find / -name aws_completer
/usr/local/bin/aws_completer
```

## Langkah 2: Identifikasi shell Anda

Untuk mengidentifikasi shell yang Anda gunakan, Anda dapat menggunakan salah satu perintah berikut.

- `echo $SHELL` - Menampilkan nama file program shell. Ini biasanya cocok dengan nama shell yang sedang digunakan, kecuali jika Anda meluncurkan shell yang berbeda setelah masuk.

```
$ echo $SHELL
/bin/bash
```

- `ps` — Menampilkan proses yang berjalan untuk pengguna saat ini. Salah satunya adalah cangkangnya.

```
$ ps
  PID TTY          TIME CMD
 2148 pts/1    00:00:00 bash
 8756 pts/1    00:00:00 ps
```

## Langkah 3: Tambahkan pelengkap ke jalur Anda

1. Temukan penulisan profil shell Anda di folder pengguna Anda.

```
$ ls -a ~/
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash — `.bash_profile`, `.profile`, atau `.bash_login`
- Zsh — `.zshrc`
- Tcsh — `.tcshrc`, `.cshrc`, atau `.login`

2. Tambahkan perintah ekspor di akhir skrip profil Anda yang mirip dengan contoh berikut. Ganti `/usr/local/bin/` dengan folder yang Anda temukan di bagian sebelumnya.

```
export PATH=/usr/local/bin/:$PATH
```

3. Muat ulang profil ke sesi saat ini untuk menerapkan perubahan tersebut. Ganti `.bash_profile` dengan nama skrip shell yang Anda temukan di bagian pertama.

```
$ source ~/.bash_profile
```

## Aktifkan penyelesaian perintah

Setelah mengonfirmasi bahwa pelengkap ada di jalur Anda, aktifkan penyelesaian perintah dengan menjalankan perintah yang sesuai untuk shell yang Anda gunakan. Anda dapat menambahkan perintah ke profil shell Anda untuk menjalankannya setiap kali Anda membuka shell baru. Di setiap perintah, ganti path `/usr/local/bin/` dengan yang ditemukan di sistem Anda di [Konfirmasikan folder pelengkap ada di jalur Anda](#)

- **bash**— Gunakan perintah `bawaancomplete`.

```
$ complete -C '/usr/local/bin/aws_completer' aws
```

Tambahkan perintah sebelumnya `~/.bashrc` untuk menjalankannya setiap kali Anda membuka shell baru. Anda `~/.bash_profile` harus mencari sumber `~/.bashrc` untuk memastikan bahwa perintah juga dijalankan di shell login.

- **zsh**— Untuk menjalankan penyelesaian perintah, Anda perlu menjalankan `bashcompinit` dengan menambahkan baris autoload berikut di akhir skrip `~/.zshrc` profil Anda.

```
$ autoload bashcompinit && bashcompinit  
$ autoload -Uz compinit && compinit
```

Untuk mengaktifkan penyelesaian perintah, gunakan perintah `bawaancomplete`.

```
$ complete -C '/usr/local/bin/aws_completer' aws
```

Tambahkan perintah sebelumnya `~/.zshrc` untuk menjalankannya setiap kali Anda membuka shell baru.

- **tcsh**— Lengkap untuk tcsh mengambil jenis kata dan pola untuk menentukan perilaku penyelesaian.

```
> complete aws 'p/*/'`aws_completer`/'
```

Tambahkan perintah sebelumnya ~/.tshrc untuk menjalankannya setiap kali Anda membuka shell baru.

Setelah Anda mengaktifkan penyelesaian perintah, [Verifikasi penyelesaian perintah](#) berfungsi.

## Verifikasi penyelesaian perintah

Setelah mengaktifkan penyelesaian perintah, muat ulang shell Anda, masukkan perintah sebagian, dan tekan Tab untuk melihat perintah yang tersedia.

```
$ aws sTAB
s3          ses          sqs          sts          swf
s3api       sns          storagegateway support
```

## Mengkonfigurasi penyelesaian perintah pada Windows

### Note

Untuk informasi tentang cara PowerShell menangani penyelesaiannya, termasuk berbagai kunci penyelesaiannya, lihat [About\\_TAB\\_EXPANSION](#) di Microsoft Docs. PowerShell

Untuk mengaktifkan penyelesaian perintah PowerShell di Windows, selesaikan langkah-langkah berikut di PowerShell.

1. Buka Anda \$PROFILE dengan perintah berikut.

```
PS C:\> Notepad $PROFILE
```

Jika Anda tidak memiliki \$PROFILE, buat profil pengguna menggunakan perintah berikut.

```
PS C:\> if (!(Test-Path -Path $PROFILE ))
{ New-Item -Type File -Path $PROFILE -Force }
```

Untuk informasi selengkapnya tentang PowerShell profil, lihat [Cara Menggunakan Profil di Windows PowerShell ISE](#) di situs web Microsoft Docs.

2. Untuk mengaktifkan penyelesaian perintah, tambahkan blok kode berikut ke profil Anda, simpan, lalu tutup file.

```
Register-ArgumentCompleter -Native -CommandName aws -ScriptBlock {
    param($commandName, $wordToComplete, $cursorPosition)
    $env:COMP_LINE=$wordToComplete
    if ($env:COMP_LINE.Length -lt $cursorPosition){
        $env:COMP_LINE=$env:COMP_LINE + " "
    }
    $env:COMP_POINT=$cursorPosition
    aws_completer.exe | ForEach-Object {
        [System.Management.Automation.CompletionResult]::new($_, $_,
        'ParameterValue', $_)
    }
    Remove-Item Env:\COMP_LINE
    Remove-Item Env:\COMP_POINT
}
```

3. Setelah mengaktifkan penyelesaian perintah, muat ulang shell Anda, masukkan perintah sebagian, dan tekan Tab untuk menelusuri perintah yang tersedia.

```
$ aws sTab
```

```
$ aws s3
```

Untuk melihat semua perintah yang tersedia yang tersedia untuk Anda selesaikan, masukkan sebagian perintah dan tekan Ctrl+Space.

```
$ aws sCtrl + Space
s3          ses          sqs          sts          swf
s3api       sns          storagegateway support
```

## AWS CLImencoba lagi

Topik ini menjelaskan bagaimana AWS CLI mungkin melihat panggilan ke AWS layanan gagal karena masalah yang tidak terduga. Masalah ini dapat terjadi di sisi server atau mungkin gagal

karena pembatasan tarif dari AWS layanan yang Anda coba panggil. Kegagalan semacam ini biasanya tidak memerlukan penanganan khusus dan panggilan dilakukan secara otomatis lagi, seringkali setelah masa tunggu yang singkat. AWS CLI ini menyediakan banyak fitur untuk membantu mencoba kembali panggilan klien ke AWS layanan ketika jenis kesalahan atau pengecualian ini dialami.

Topik

- [Mode coba lagi yang tersedia](#)
- [Mengkonfigurasi mode coba lagi](#)
- [Melihat log upaya coba lagi](#)

## Mode coba lagi yang tersedia

Ini AWS CLI memiliki beberapa mode untuk dipilih tergantung pada versi Anda:

- [Mode coba lagi lama](#)
- [Mode coba lagi standar](#)
- [Mode coba lagi adaptif](#)

### Mode coba lagi lama

Mode lama menggunakan penanganan coba lagi yang lebih lama yang memiliki fungsionalitas terbatas yang meliputi:

- Nilai default 4 untuk percobaan ulang maksimum, membuat total 5 upaya panggilan. Nilai ini dapat ditimpa melalui parameter `max_attempts` konfigurasi.
- DynamoDB memiliki nilai default 9 untuk percobaan ulang maksimum, membuat total 10 upaya panggilan. Nilai ini dapat ditimpa melalui parameter `max_attempts` konfigurasi.
- Coba lagi upaya untuk sejumlah kesalahan/pengecualian terbatas berikut:
  - Kesalahan soket/koneksi umum:
    - `ConnectionError`
    - `ConnectionClosedError`
    - `ReadTimeoutError`
    - `EndpointConnectionError`
  - Kesalahan dan pengecualian pelambatan sisi layanan/batas:

- Throttling
  - ThrottlingException
  - ThrottledException
  - RequestThrottledException
  - ProvisionedThroughputExceededException
- Coba lagi pada beberapa kode status HTTP, termasuk 429, 500, 502, 503, 504, dan 509.
  - Setiap upaya coba lagi akan mencakup backoff eksponensial dengan faktor dasar 2.

## Mode coba lagi standar

Mode standar adalah seperangkat aturan coba ulang standar di seluruh AWS SDK dengan fungsionalitas lebih dari lawas. Mode ini adalah default untuk AWS CLI versi 2. Mode standar dibuat untuk AWS CLI versi 2 dan di-backport ke AWS CLI versi 1. Fungsionalitas mode standar meliputi:

- Nilai default 2 untuk percobaan ulang maksimum, membuat total 3 upaya panggilan. Nilai ini dapat ditimpa melalui parameter `max_attempts` konfigurasi.
- Coba lagi upaya untuk daftar kesalahan/pengecualian yang diperluas berikut:
  - Kesalahan/pengecualian sementara
    - RequestTimeout
    - RequestTimeoutException
    - PriorRequestNotComplete
    - ConnectionError
    - HTTPClientError
  - Kesalahan dan pengecualian pelambatan sisi layanan/batas:
    - Throttling
    - ThrottlingException
    - ThrottledException
    - RequestThrottledException
    - TooManyRequestsException
    - ProvisionedThroughputExceededException
    - TransactionInProgressException
    - RequestLimitExceeded

- `BandwidthLimitExceeded`
  - `LimitExceededException`
  - `RequestThrottled`
  - `SlowDown`
  - `EC2ThrottledException`
- Coba lagi upaya pada kode kesalahan sementara yang tidak deskriptif. Secara khusus, kode status HTTP ini: 500, 502, 503, 504.
  - Setiap upaya coba lagi akan mencakup backoff eksponensial dengan faktor dasar 2 untuk waktu backoff maksimum 20 detik.

## Mode coba lagi adaptif

### Warning

Mode adaptif adalah mode eksperimental dan dapat berubah, baik dalam fitur maupun perilaku.

Mode coba lagi adaptif adalah mode coba ulang eksperimental yang mencakup semua fitur mode standar. Selain fitur mode standar, mode adaptif juga memperkenalkan pembatasan laju sisi klien melalui penggunaan bucket token dan variabel rate-limit yang diperbarui secara dinamis dengan setiap upaya coba lagi. Mode ini menawarkan fleksibilitas dalam percobaan ulang sisi klien yang menyesuaikan dengan respons status kesalahan/pengecualian dari suatu layanan. AWS

Dengan setiap upaya coba lagi baru, mode adaptif memodifikasi variabel batas laju berdasarkan kesalahan, pengecualian, atau kode status HTTP yang disajikan dalam respons dari layanan. AWS Variabel rate-limit ini kemudian digunakan untuk menghitung tingkat panggilan baru untuk klien. Setiap respons HTTP pengecualian/kesalahan atau tidak berhasil (disediakan dalam daftar di atas) dari AWS layanan memperbarui variabel batas kecepatan saat percobaan ulang terjadi hingga keberhasilan tercapai, keranjang token habis, atau nilai upaya maksimum yang dikonfigurasi tercapai.

## Mengkonfigurasi mode coba lagi

AWS CLITermasuk berbagai konfigurasi coba lagi serta metode konfigurasi untuk dipertimbangkan saat membuat objek klien Anda.



## Metode konfigurasi yang tersedia

DiAWS CLI, pengguna dapat mengonfigurasi percobaan ulang dengan cara berikut:

- Variabel-variabel lingkungan
- AWS CLIberkas konfigurasi

Pengguna dapat menyesuaikan opsi coba lagi berikut:

- Mode coba lagi - Menentukan modus coba lagi yang digunakan. AWS CLI Seperti dijelaskan sebelumnya, ada tiga mode coba lagi yang tersedia: warisan, standar, dan adaptif. Nilai default untuk 2 adalah standar.
- Upaya maksimal - Menentukan nilai percobaan ulang maksimum yang digunakan oleh penanganan AWS CLI coba lagi, di mana panggilan awal diperhitungkan terhadap nilai yang Anda berikan. Nilai bawaannya adalah 5.

## Mendefinisikan konfigurasi coba lagi dalam variabel lingkungan Anda

Untuk menentukan konfigurasi coba ulang AndaAWS CLI, perbarui variabel lingkungan sistem operasi Anda.

Variabel lingkungan coba lagi adalah:

- AWS\_RETRY\_MODE
- AWS\_MAX\_ATTEMPTS

Untuk informasi lebih lanjut tentang variabel lingkungan, lihat[Variabel lingkungan untuk mengkonfigurasi AWS CLI](#).

## Mendefinisikan konfigurasi coba lagi dalam file konfigurasi Anda AWS

Untuk mengubah konfigurasi coba lagi, perbarui file AWS konfigurasi global Anda. Lokasi default untuk file AWS konfigurasi Anda adalah `~/.aws/config`.

Berikut ini adalah contoh file AWS konfigurasi:

```
[default]
retry_mode = standard
```

```
max_attempts = 6
```

Untuk informasi selengkapnya tentang file konfigurasi, lihat [Konfigurasi dan pengaturan file kredensi](#).

## Melihat log upaya coba lagi

AWS CLI menggunakan metodologi coba ulang Boto3 dan logging. Anda dapat menggunakan `--debug` opsi pada perintah apa pun untuk menerima log debug. Untuk informasi selengkapnya tentang cara menggunakan `--debug` opsi, lihat [Opsi baris perintah](#).

Jika Anda mencari “coba lagi” di log debug Anda, Anda akan menemukan informasi coba lagi yang Anda butuhkan. Entri log klien untuk percobaan ulang bergantung pada mode coba lagi yang telah Anda aktifkan.

Mode warisan:

Pesan coba lagi dihasilkan oleh `botocore.retryhandler`. Anda akan melihat salah satu dari tiga pesan:

- No retry needed
- Retry needed, action of: *<action\_name>*
- Reached the maximum number of retry attempts: *<attempt\_number>*

Mode standar atau adaptif:

Pesan coba lagi dihasilkan oleh `botocore.retries.standard`. Anda akan melihat salah satu dari tiga pesan:

- No retrying request
- Retry needed, retrying request after delay of: *<delay\_value>*
- Retry needed but retry quota reached, not retrying request

Untuk file definisi lengkap dari percobaan ulang botocore, lihat [\\_retry.json di repositori](#) botocore. GitHub

# Gunakan proxy HTTP

Untuk mengakses AWS melalui server proxy, Anda dapat mengonfigurasi variabel `HTTP_PROXY` dan `HTTPS_PROXY` lingkungan dengan nama domain DNS atau alamat IP dan nomor port yang digunakan server proxy Anda.

## Topik

- [Menggunakan contoh](#)
- [Mengautentikasi ke proxy](#)
- [Menggunakan proxy di instans Amazon EC2](#)
- [Pemecahan Masalah](#)

## Menggunakan contoh

### Note

Contoh berikut menunjukkan nama variabel lingkungan dalam semua huruf besar. Namun, jika Anda menentukan variabel dua kali menggunakan kasus yang berbeda, huruf kecil diutamakan. Kami menyarankan Anda mendefinisikan setiap variabel hanya sekali untuk menghindari kebingungan sistem dan perilaku yang tidak terduga.

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan alamat IP eksplisit proxy Anda atau nama DNS yang menyelesaikan ke alamat IP proxy Anda. Entah dapat diikuti oleh titik dua dan nomor port yang kueri harus dikirim.

## Linux or macOS

```
$ export HTTP_PROXY=http://10.15.20.25:1234
$ export HTTP_PROXY=http://proxy.example.com:1234
$ export HTTPS_PROXY=http://10.15.20.25:5678
$ export HTTPS_PROXY=http://proxy.example.com:5678
```

## Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx HTTP_PROXY http://10.15.20.25:1234
```

```
C:\> setx HTTP_PROXY http://proxy.example.com:1234
C:\> setx HTTPS_PROXY http://10.15.20.25:5678
C:\> setx HTTPS_PROXY http://proxy.example.com:5678
```

Menggunakan [setx](#) untuk mengatur variabel lingkungan mengubah nilai yang digunakan dalam sesi prompt perintah saat ini dan semua sesi prompt perintah yang Anda buat setelah menjalankan perintah. Itu tidak mempengaruhi shell perintah lain yang sudah berjalan pada saat Anda menjalankan perintah.

Untuk mengatur untuk sesi saat ini saja

Menggunakan [set](#) untuk mengatur variabel lingkungan mengubah nilai yang digunakan sampai akhir sesi prompt perintah saat ini, atau sampai Anda mengatur variabel ke nilai yang berbeda.

```
C:\> set HTTP_PROXY=http://10.15.20.25:1234
C:\> set HTTP_PROXY=http://proxy.example.com:1234
C:\> set HTTPS_PROXY=http://10.15.20.25:5678
C:\> set HTTPS_PROXY=http://proxy.example.com:5678
```

## Mengautentikasi ke proxy

### Note

AWS CLI tidak mendukung proxy NTLM. [Jika Anda menggunakan proxy protokol NTLM atau Kerberos, Anda mungkin dapat terhubung melalui proxy otentikasi seperti Cntlm.](#)

AWS CLI mendukung otentikasi HTTP Basic. Tentukan nama pengguna dan kata sandi di URL proxy, sebagai berikut.

Linux or macOS

```
$ export HTTP_PROXY=http://username:password@proxy.example.com:1234
$ export HTTPS_PROXY=http://username:password@proxy.example.com:5678
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx HTTP_PROXY http://username:password@proxy.example.com:1234
```

```
C:\> setx HTTPS_PROXY http://username:password@proxy.example.com:5678
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set HTTP_PROXY=http://username:password@proxy.example.com:1234
C:\> set HTTPS_PROXY=http://username:password@proxy.example.com:5678
```

## Menggunakan proxy di instans Amazon EC2

[Jika Anda mengonfigurasi proxy pada instans Amazon EC2 yang diluncurkan dengan peran IAM terlampir, pastikan Anda mengecualikan alamat yang digunakan untuk mengakses metadata instans.](#)

Untuk melakukan ini, atur variabel NO\_PROXY lingkungan ke alamat IP layanan metadata instance, 169.254.169.254. Alamat ini tidak berbeda.

Linux or macOS

```
$ export NO_PROXY=169.254.169.254
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx NO_PROXY 169.254.169.254
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set NO_PROXY=169.254.169.254
```

## Pemecahan Masalah

Jika Anda menemukan masalah dengan AWS CLI, lihat [Memecahkan masalah kesalahan](#) untuk langkah-langkah pemecahan masalah. Untuk langkah pemecahan masalah yang paling relevan, lihat [the section called “Kesalahan sertifikat SSL”](#)

## Gunakan titik akhir di AWS CLI

Untuk terhubung secara terprogram ke sebuah Layanan AWS, Anda menggunakan endpoint.

Endpoint adalah URL titik masuk untuk layanan AWS web. The AWS Command Line Interface (AWS

CLI) secara otomatis menggunakan titik akhir default untuk setiap layanan Wilayah AWS, tetapi Anda dapat menentukan titik akhir alternatif untuk permintaan API Anda.

Topik titik akhir

- [Tetapkan titik akhir untuk satu perintah](#)
- [Tetapkan titik akhir global untuk semua Layanan AWS](#)
- [Setel untuk menggunakan titik akhir FIP untuk semua Layanan AWS](#)
- [Setel untuk menggunakan titik akhir dual-stack untuk semua Layanan AWS](#)
- [Tetapkan titik akhir khusus layanan](#)
  - [Titik akhir khusus layanan: Variabel lingkungan](#)
  - [Titik akhir khusus layanan: File bersama config](#)
  - [Titik akhir khusus layanan: Daftar pengidentifikasi khusus layanan](#)
- [Konfigurasi titik akhir dan pengaturan diutamakan](#)

## Tetapkan titik akhir untuk satu perintah

Untuk mengganti pengaturan titik akhir atau variabel lingkungan untuk satu perintah, gunakan opsi baris `--endpoint-url` perintah. Contoh perintah berikut menggunakan URL endpoint Amazon S3 kustom.

```
$ aws s3 ls --endpoint-url http://localhost:4567
```

## Tetapkan titik akhir global untuk semua Layanan AWS

Untuk merutekan permintaan semua layanan ke URL titik akhir kustom, gunakan salah satu pengaturan berikut:

- Variabel lingkungan:
  - [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#)- Abaikan URL titik akhir yang dikonfigurasi.

Linux or macOS

```
$ export AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_IGNORE_CONFIGURED_ENDPOINT_URLS true
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

### PowerShell

```
PS C:\> $Env:AWS_IGNORE_CONFIGURED_ENDPOINT_URLS="true"
```

- [AWS\\_ENDPOINT\\_URL](#) - Tetapkan URL titik akhir global.

Linux or macOS

```
$ export AWS_ENDPOINT_URL=http://localhost:4567
```

### Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_ENDPOINT_URL http://localhost:4567
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set AWS_ENDPOINT_URL=http://localhost:4567
```

### PowerShell

```
PS C:\> $Env:AWS_ENDPOINT_URL="http://localhost:4567"
```

- `configBerkas`:
  - [ignore\\_configure\\_endpoint\\_urls](#) - Abaikan URL titik akhir yang dikonfigurasi.

```
ignore_configure_endpoint_urls = true
```

- [endpoint\\_url](#) - Tetapkan URL titik akhir global.

```
endpoint_url = http://localhost:4567
```

Titik akhir khusus layanan dan opsi baris `--endpoint-url` perintah mengganti titik akhir global apa pun.

## Setel untuk menggunakan titik akhir FIP untuk semua Layanan AWS

Untuk merutekan permintaan semua layanan agar menggunakan titik akhir FIP, gunakan salah satu dari berikut ini:

- [AWS\\_USE\\_FIPS\\_ENDPOINT](#) variabel lingkungan.

Linux or macOS

```
$ export AWS_USE_FIPS_ENDPOINT=true
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_USE_FIPS_ENDPOINT true
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set AWS_USE_FIPS_ENDPOINT=true
```

PowerShell

```
PS C:\> $Env:AWS_USE_FIPS_ENDPOINT="true"
```

- [use\\_fips\\_endpoint](#) pengaturan file.

```
use_fips_endpoint = true
```

Beberapa AWS layanan menawarkan titik akhir yang mendukung [Federal Information Processing Standard \(FIPS\) 140-2](#) di beberapa Wilayah AWS. Ketika AWS layanan mendukung FIPS, pengaturan ini menentukan titik akhir FIPS apa yang harus digunakan. AWS CLI Tidak seperti AWS endpoint standar, endpoint FIPS menggunakan pustaka perangkat lunak TLS yang sesuai dengan FIPS 140-2. Titik akhir ini mungkin diperlukan oleh korporasi yang berinteraksi dengan pemerintah Amerika Serikat.



Jika pengaturan ini diaktifkan, tetapi titik akhir FIPS tidak ada untuk layanan di Anda Wilayah AWS, AWS perintah mungkin gagal. Dalam hal ini, tentukan secara manual titik akhir yang akan digunakan dalam perintah menggunakan `--endpoint-url` opsi atau gunakan titik akhir [khusus layanan](#).

Untuk informasi selengkapnya tentang menentukan titik akhir FIPS menurut Wilayah AWS, lihat Titik Akhir [FIPS](#) menurut Layanan.

## Setel untuk menggunakan titik akhir dual-stack untuk semua Layanan AWS

Untuk merutekan permintaan semua layanan agar menggunakan titik akhir dual-stack bila tersedia, gunakan salah satu pengaturan berikut:

- [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#) variabel lingkungan.

Linux or macOS

```
$ export AWS_USE_DUALSTACK_ENDPOINT=true
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_USE_DUALSTACK_ENDPOINT true
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set AWS_USE_DUALSTACK_ENDPOINT=true
```

PowerShell

```
PS C:\> $Env:AWS_USE_DUALSTACK_ENDPOINT="true"
```

- [use\\_dualstack\\_endpoint](#) pengaturan file.

```
use_dualstack_endpoint = true
```

Mengaktifkan penggunaan titik akhir dual-stack untuk mengirim permintaan. AWS Untuk mempelajari lebih lanjut tentang titik akhir dual-stack, yang mendukung lalu lintas IPv4 dan IPv6, lihat Menggunakan titik akhir [tumpukan ganda Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan

Sederhana Amazon. Titik akhir dual-stack tersedia untuk beberapa layanan di beberapa wilayah. Jika titik akhir tumpukan ganda tidak ada untuk layanan atau Wilayah AWS, permintaan gagal. Ini dinonaktifkan secara default.

## Tetapkan titik akhir khusus layanan

Konfigurasi titik akhir khusus layanan menyediakan opsi untuk menggunakan titik akhir persisten yang Anda pilih untuk permintaan. AWS CLI Pengaturan ini memberikan fleksibilitas untuk mendukung titik akhir lokal, titik akhir VPC, dan lingkungan pengembangan lokal pihak ketiga AWS. Titik akhir yang berbeda dapat digunakan untuk lingkungan pengujian dan produksi. Anda dapat menentukan URL endpoint untuk individu Layanan AWS.

Titik akhir khusus layanan dapat ditentukan dengan cara berikut:

- Opsi baris perintah [--endpoint-url](#) untuk satu perintah.
- Variabel lingkungan:
  - [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#)- Abaikan semua URL endpoint yang dikonfigurasi, kecuali ditentukan pada baris perintah.
  - [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#)- Menentukan endpoint kustom yang digunakan untuk layanan tertentu, di mana <SERVICE> mengganti dengan identifier. Layanan AWS Untuk semua variabel khusus layanan, lihat. [the section called “Daftar pengidentifikasi khusus layanan”](#)
- configberkas:
  - [ignore\\_configure\\_endpoint\\_urls](#)- Abaikan semua URL endpoint yang dikonfigurasi, kecuali ditentukan menggunakan variabel lingkungan atau pada baris perintah.
  - [services](#)Bagian config file dikombinasikan dengan pengaturan [endpoint\\_url](#) file.

Topik titik akhir khusus layanan:

- [Titik akhir khusus layanan: Variabel lingkungan](#)
- [Titik akhir khusus layanan: File bersama config](#)
- [Titik akhir khusus layanan: Daftar pengidentifikasi khusus layanan](#)

### Titik akhir khusus layanan: Variabel lingkungan

Variabel lingkungan mengganti pengaturan dalam file konfigurasi Anda, tetapi jangan mengganti opsi yang ditentukan pada baris perintah. Gunakan variabel lingkungan jika Anda ingin semua profil menggunakan titik akhir yang sama di perangkat Anda.

Berikut ini adalah variabel lingkungan khusus layanan:

- [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#)- Abaikan semua URL endpoint yang dikonfigurasi, kecuali ditentukan pada baris perintah.

Linux or macOS

```
$ export AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_IGNORE_CONFIGURED_ENDPOINT_URLS true
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

PowerShell

```
PS C:\> $Env:AWS_IGNORE_CONFIGURED_ENDPOINT_URLS="true"
```

- [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#)- Menentukan endpoint kustom yang digunakan untuk layanan tertentu, di mana <SERVICE> diganti dengan identifier. Layanan AWS Untuk semua variabel khusus layanan, lihat. [the section called “Daftar pengidentifikasi khusus layanan”](#)

Contoh variabel lingkungan berikut menetapkan titik akhir untuk AWS Elastic Beanstalk:

Linux or macOS

```
$ export AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:4567
```

Windows Command Prompt

Untuk mengatur semua sesi

```
C:\> setx AWS_ENDPOINT_URL_ELASTIC_BEANSTALK http://localhost:4567
```

Untuk mengatur untuk sesi saat ini saja

```
C:\> set AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:4567
```

## PowerShell

```
PS C:\> $Env:AWS_ENDPOINT_URL_ELASTIC_BEANSTALK="http://localhost:4567"
```

Untuk informasi selengkapnya tentang pengaturan variabel lingkungan, lihat [the section called “Variabel lingkungan”](#).

## Titik akhir khusus layanan: File bersama **config**

Dalam config file bersama, `endpoint_url` digunakan dalam beberapa bagian. Untuk menyetel titik akhir khusus layanan, gunakan `endpoint_url` setelan yang bersarang di bawah kunci pengenalan layanan di dalam bagian. `services` Untuk detail tentang mendefinisikan `services` bagian dalam config file bersama Anda, lihat [the section called “services”](#).

Contoh berikut menggunakan `services` bagian untuk mengonfigurasi URL titik akhir khusus layanan untuk Amazon S3 dan titik akhir global khusus yang digunakan untuk semua layanan lainnya:

```
[profile dev1]  
endpoint_url = http://localhost:1234  
services = s3-specific  
  
[services testing-s3]  
s3 =  
  endpoint_url = http://localhost:4567
```

Satu profil dapat mengonfigurasi titik akhir untuk beberapa layanan. Contoh berikut menetapkan URL titik akhir khusus layanan untuk Amazon S3 dan di profil yang sama. AWS Elastic Beanstalk

Untuk daftar semua kunci pengenalan layanan yang akan digunakan di `services` bagian ini, lihat [Daftar pengidentifikasi khusus layanan](#).

```
[profile dev1]  
services = testing-s3-and-eb  
  
[services testing-s3-and-eb]  
s3 =
```

```
endpoint_url = http://localhost:4567
elastic_beanstalk =
  endpoint_url = http://localhost:8000
```

Bagian konfigurasi layanan dapat digunakan di beberapa profil. Contoh berikut memiliki dua profil menggunakan `services` definisi yang sama:

```
[profile dev1]
output = json
services = testing-s3

[profile dev2]
output = text
services = testing-s3

[services testing-s3]
s3 =
  endpoint_url = https://localhost:4567
```

## Titik akhir khusus layanan: Daftar pengidentifikasi khusus layanan

Layanan AWS Pengenal didasarkan pada model API `serviceId` dengan mengganti semua spasi dengan garis bawah dan huruf kecil semua huruf.

Contoh pengenal layanan berikut menggunakan AWS Elastic Beanstalk. AWS Elastic Beanstalk memiliki `serviceId` dari [Elastic Beanstalk](#), oleh karena itu kunci pengenal layanan adalah `elastic_beanstalk`.

Tabel berikut mencantumkan semua pengidentifikasi khusus layanan, kunci `config` file, dan variabel lingkungan.

## Konfigurasi titik akhir dan pengaturan diutamakan

Pengaturan konfigurasi titik akhir terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau dinyatakan secara eksplisit pada baris perintah sebagai parameter. Pengaturan konfigurasi AWS CLI titik akhir diutamakan dalam urutan sebagai berikut:

1. Opsi baris `--endpoint-url` perintah.
2. Jika diaktifkan, variabel lingkungan endpoint [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global atau pengaturan profil [ignore\\_configure\\_endpoint\\_urls](#) untuk mengabaikan endpoint kustom.

3. Nilai yang diberikan oleh variabel lingkungan khusus layanan [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), seperti `AWS_ENDPOINT_URL_DYNAMODB`
4. Nilai yang disediakan oleh [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#), dan variabel [AWS\\_ENDPOINT\\_URL](#) lingkungan.
5. Nilai endpoint khusus layanan yang disediakan oleh [endpoint\\_url](#) pengaturan dalam `services` bagian file bersama. `config`
6. Nilai yang diberikan oleh [endpoint\\_url](#) pengaturan `profile` dalam `config` file bersama.
7. [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#), dan [endpoint\\_url](#) pengaturan.
8. URL endpoint default untuk masing-masing Layanan AWS digunakan terakhir. Untuk daftar titik akhir layanan standar yang tersedia di setiap Wilayah, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services

## Otentikasi dan akses kredensial

Anda harus menetapkan bagaimana AWS CLI otentikasi dengan AWS ketika Anda mengembangkan dengan AWS layanan. Untuk mengonfigurasi kredensi untuk akses terprogram untuk AWS CLI, pilih salah satu opsi berikut. Pilihannya adalah dalam urutan rekomendasi.

Pengguna mana yang membutuhkan akses programatis?	Tujuan	Petunjuk
Identitas tenaga kerja (AWS IAM Identity Center pengguna)	(Disarankan) Gunakan kredensial jangka pendek.	<a href="#">the section called “Autentikasi Pusat Identitas IAM”</a>
IAM	Gunakan kredensi jangka pendek.	<a href="#">the section called “Kredensi jangka pendek”</a>
IAM atau Identitas tenaga kerja (AWS IAM Identity Center pengguna)	Gunakan metadata instans Amazon EC2 untuk kredensial.	<a href="#">the section called “Menggunakan kredensial untuk metadata instans Amazon EC2”</a>
IAM atau Identitas tenaga kerja (AWS IAM Identity Center pengguna)	Pasangkan metode kredensi lain dan ambil peran untuk izin.	<a href="#">the section called “Peran IAM ”</a>
IAM	(Tidak disarankan) Gunakan kredensi jangka panjang.	<a href="#">the section called “Pengguna IAM”</a>
IAM atau Identitas tenaga kerja (AWS IAM Identity Center pengguna)	(Tidak disarankan) Pasangkan metode kredensi lain tetapi gunakan nilai kredensi yang disimpan di lokasi di luar AWS CLI	<a href="#">the section called “Kredensial eksternal”</a>

## Konfigurasi dan prioritas kredensial

Kredensi dan pengaturan konfigurasi terletak di beberapa tempat, seperti variabel sistem atau lingkungan pengguna, file AWS konfigurasi lokal, atau secara eksplisit dideklarasikan pada baris perintah sebagai parameter. Otentikasi tertentu lebih diutamakan daripada yang lain. Pengaturan AWS CLI otentikasi diutamakan dalam urutan sebagai berikut:

1. [Opsi baris perintah](#) — Mengganti pengaturan di lokasi lain, seperti, `--region--output`, dan `--profile` parameter.
2. [Variabel lingkungan](#) — Anda dapat menyimpan nilai dalam variabel lingkungan sistem Anda.
3. [Asumsikan peran](#) - Asumsikan izin peran IAM melalui konfigurasi atau perintah. [aws sts assume-role](#)
4. [Asumsikan peran dengan identitas web](#) - Asumsikan izin peran IAM menggunakan identitas web melalui konfigurasi atau perintah. [aws sts assume-role](#)
5. [AWS IAM Identity Center](#)— Pengaturan konfigurasi Pusat Identitas IAM yang disimpan dalam config file diperbarui saat Anda menjalankan `aws configure sso` perintah. Kredensi kemudian diautentikasi ketika Anda menjalankan perintah. `aws sso login configFile` ini terletak `~/.aws/config` di Linux atau macOS, atau `C:\Users\USERNAME\.aws\config` di Windows.
6. [File kredensial](#) — config File credentials dan diperbarui saat Anda menjalankan perintah. `aws configure credentialsFile` ini terletak `~/.aws/credentials` di Linux atau macOS, atau `C:\Users\USERNAME\.aws\credentials` di Windows.
7. [Proses kustom](#) - Dapatkan kredensial Anda dari sumber eksternal.
8. [File konfigurasi](#) — config File credentials dan diperbarui saat Anda menjalankan perintah `aws configure`. configFile ini terletak `~/.aws/config` di Linux atau macOS, atau `C:\Users\USERNAME\.aws\config` di Windows.
9. [Kredensi kontainer](#) - Anda dapat mengaitkan peran IAM dengan masing-masing definisi tugas Amazon Elastic Container Service (Amazon ECS). Kredensi sementara untuk peran itu kemudian tersedia untuk wadah tugas itu. Untuk informasi lebih lanjut, lihat [Peran IAM untuk Tugas](#) dalam Panduan Developer Amazon Elastic Container Service.
10. [Kredensial profil instans Amazon EC2](#) — Anda dapat mengaitkan peran IAM dengan setiap instans Amazon Elastic Compute Cloud (Amazon EC2). Kredensi sementara untuk peran itu kemudian tersedia untuk kode yang berjalan dalam instance. Kredensialnya dikirimkan melalui layanan metadata Amazon EC2. Untuk informasi selengkapnya, lihat [Peran IAM untuk Amazon EC2](#) di Panduan Pengguna Amazon EC2 [dan Menggunakan Profil Instance](#) di Panduan Pengguna IAM.



## Topik tambahan di bagian ini

- [the section called “Autentikasi Pusat Identitas IAM”](#)
- [the section called “Kredensi jangka pendek”](#)
- [the section called “Peran IAM ”](#)
- [the section called “Pengguna IAM”](#)
- [the section called “Menggunakan kredensyal untuk metadata instans Amazon EC2”](#)
- [the section called “Kredensyal eksternal”](#)

## Konfigurasi AWS CLI yang akan digunakan AWS IAM Identity Center

Terutama ada dua cara untuk mengautentikasi pengguna dengan AWS IAM Identity Center (IAM Identity Center) untuk mendapatkan kredensi untuk menjalankan AWS Command Line Interface (AWS CLI) perintah melalui file: `config`

- (Disarankan) [Konfigurasi penyedia token SSO](#). Konfigurasi penyedia token SSO, AWS SDK, atau alat Anda dapat secara otomatis mengambil token otentikasi yang diperbarui.
- [Konfigurasi lama yang tidak dapat disegarkan](#). Saat menggunakan konfigurasi lama yang tidak dapat disegarkan, Anda perlu menyegarkan token secara manual karena kedaluwarsa secara berkala.

Saat menggunakan IAM Identity Center, Anda dapat masuk ke Active Directory, direktori IAM Identity Center bawaan, atau [IDP lain yang terhubung ke](#) IAM Identity Center. Anda dapat memetakan kredensial ini ke peran AWS Identity and Access Management (IAM) agar Anda dapat menjalankan perintah. AWS CLI

Terlepas dari IDP mana yang Anda gunakan, IAM Identity Center mengabstraksi perbedaan tersebut. Misalnya, Anda dapat menghubungkan Microsoft Azure AD seperti yang dijelaskan dalam artikel blog [Evolusi Berikutnya di Pusat Identitas IAM](#).

**Note**

Untuk informasi tentang penggunaan autentikasi pembawa, yang tidak menggunakan ID akun dan peran, lihat [Menyiapkan untuk menggunakan AWS CLI dengan CodeCatalyst di CodeCatalyst](#) Panduan Pengguna Amazon.

Topik di bagian ini

- [AWS CLI Konfigurasi kredensial penyedia token IAM Identity Center untuk menggunakan penyegaran autentikasi otomatis](#)
- [Konfigurasi lama yang tidak dapat disegarkan untuk AWS IAM Identity Center](#)
- [Menggunakan profil bernama Pusat Identitas IAM](#)

## AWS CLI Konfigurasi kredensial penyedia token IAM Identity Center untuk menggunakan penyegaran autentikasi otomatis

Topik ini menjelaskan cara mengonfigurasi AWS CLI untuk mengautentikasi pengguna dengan konfigurasi penyedia token AWS IAM Identity Center (IAM Identity Center). Dengan menggunakan konfigurasi penyedia token SSO ini, AWS SDK atau alat Anda dapat secara otomatis mengambil token otentikasi yang diperbarui.

Saat menggunakan IAM Identity Center, Anda dapat masuk ke Active Directory, direktori IAM Identity Center bawaan, atau [IDP lain yang terhubung ke](#) IAM Identity Center. Anda dapat memetakan kredensial ini ke peran AWS Identity and Access Management (IAM) agar Anda dapat menjalankan perintah. AWS CLI

Terlepas dari IDP mana yang Anda gunakan, IAM Identity Center mengabstraksikan perbedaan tersebut. Misalnya, Anda dapat menghubungkan Microsoft Azure AD seperti yang dijelaskan dalam artikel blog [Evolusi Berikutnya di Pusat Identitas IAM](#).

**Note**

Untuk informasi tentang penggunaan autentikasi pembawa, yang tidak menggunakan ID akun dan peran, lihat [Menyiapkan untuk menggunakan AWS CLI dengan CodeCatalyst di CodeCatalyst](#) Panduan Pengguna Amazon.

Anda dapat menggunakan konfigurasi penyedia token SSO untuk menyegarkan token autentikasi secara otomatis sesuai kebutuhan untuk aplikasi Anda, dan menggunakan opsi [durasi sesi](#) yang diperpanjang. Anda dapat mengonfigurasi ini dengan cara berikut:

- Secara otomatis, menggunakan `aws configure sso-session` perintah `aws configure sso` dan. Perintah berikut adalah wizard yang memandu Anda melalui konfigurasi profil Anda dan `sso-session` informasi adalah sebagai berikut:
  - Gunakan [aws configure sso](#) untuk membuat atau mengedit `config` profil dan `sso-session` bagian Anda.
  - Gunakan [aws configure sso-session](#) untuk membuat atau mengedit hanya `sso-session` bagian.
- [Secara manual](#), dengan mengedit `config` file yang menyimpan profil bernama.

## Prasyarat

- Instal AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#).
- Anda harus terlebih dahulu memiliki akses ke otentikasi SSO dalam IAM Identity Center. Pilih salah satu metode berikut untuk mengakses AWS kredensial Anda.

Saya tidak memiliki akses melalui IAM Identity Center

Ikuti petunjuk di [Memulai](#) di Panduan AWS IAM Identity Center Pengguna. Proses ini mengaktifkan IAM Identity Center, membuat pengguna administratif, dan menambahkan set izin hak istimewa yang paling tidak sesuai.

### Note

Buat set izin yang menerapkan izin hak istimewa paling sedikit. Sebaiknya gunakan set `PowerUserAccess` izin yang telah ditentukan sebelumnya, kecuali majikan Anda telah membuat set izin khusus untuk tujuan ini.

Keluar dari portal dan masuk lagi untuk melihat opsi Anda Akun AWS dan untuk Administrator atau `PowerUserAccess`. Pilih `PowerUserAccess` saat bekerja dengan SDK. Ini juga membantu Anda menemukan detail tentang akses terprogram.

Saya sudah memiliki akses AWS melalui penyedia identitas federasi yang dikelola oleh majikan saya (seperti Azure AD atau Okta)

Masuk AWS melalui portal penyedia identitas Anda. Jika Administrator Cloud Anda telah memberi Anda izin `PowerUserAccess` (pengembang), Anda akan melihat Akun AWS bahwa Anda memiliki akses ke dan izin Anda ditetapkan. Di samping nama set izin Anda, Anda melihat opsi untuk mengakses akun secara manual atau terprogram menggunakan set izin tersebut.

Implementasi kustom dapat menghasilkan pengalaman yang berbeda, seperti nama set izin yang berbeda. Jika Anda tidak yakin izin mana yang disetel untuk digunakan, hubungi tim TI Anda untuk mendapatkan bantuan.

Saya sudah memiliki akses AWS melalui portal AWS akses yang dikelola oleh majikan saya

Masuk AWS melalui portal AWS akses. Jika Administrator Cloud Anda telah memberi Anda izin `PowerUserAccess` (pengembang), Anda akan melihat Akun AWS bahwa Anda memiliki akses ke dan izin Anda ditetapkan. Di samping nama set izin Anda, Anda melihat opsi untuk mengakses akun secara manual atau terprogram menggunakan set izin tersebut.

Saya sudah memiliki akses AWS melalui penyedia identitas kustom federasi yang dikelola oleh majikan saya

Hubungi tim TI Anda untuk bantuan.

## Konfigurasi profil Anda dengan `aws configure sso` wizard

Untuk mengonfigurasi profil Pusat Identitas IAM dan `sso-session` ke profil Anda AWS CLI

1. Kumpulkan informasi Pusat Identitas IAM Anda dengan melakukan hal berikut:
  1. Di portal AWS akses Anda, pilih set izin yang Anda gunakan untuk pengembangan, dan pilih tautan Kunci akses.
  2. Di kotak dialog Dapatkan kredensi, pilih tab yang cocok dengan sistem operasi Anda.
  3. Pilih metode kredensial Pusat Identitas IAM untuk mendapatkan SSO Start URL dan SSO Region nilai yang perlu Anda jalankan. `aws configure sso`
  4. Untuk informasi tentang nilai cakupan mana yang akan didaftarkan, lihat [cakupan Akses OAuth 2.0](#) di Panduan Pengguna Pusat Identitas IAM.
2. Di terminal pilihan Anda, jalankan `aws configure sso` perintah dan berikan URL awal Pusat Identitas IAM Anda dan AWS Wilayah yang menghosting direktori Pusat Identitas.

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

3. AWS CLI Upaya untuk membuka browser default Anda dan memulai proses login untuk akun IAM Identity Center Anda.

```
Attempting to automatically open the SSO authorization page in your default browser.
```

Jika AWS CLI tidak dapat membuka browser, pesan berikut muncul dengan instruksi tentang cara memulai proses login secara manual.

```
If the browser does not open or you wish to use a different device to authorize this request, open the following URL:
```

```
https://device.sso.us-west-2.amazonaws.com/
```

```
Then enter the code:
```

```
QCFK-N451
```

IAM Identity Center menggunakan kode untuk mengaitkan sesi IAM Identity Center dengan sesi Anda saat ini AWS CLI . Halaman browser IAM Identity Center meminta Anda untuk masuk dengan kredensial Pusat Identitas IAM Anda. Ini memberikan izin AWS CLI untuk mengambil dan menampilkan AWS akun dan peran yang Anda berwenang untuk digunakan dengan IAM Identity Center.

#### Note

Proses masuk dapat meminta Anda untuk mengizinkan AWS CLI akses ke data Anda. Karena AWS CLI dibangun di atas SDK untuk Python, pesan izin mungkin berisi variasi nama. `botocore`

4. Ini AWS CLI menampilkan AWS akun yang tersedia untuk Anda gunakan. Jika Anda berwenang untuk hanya menggunakan satu akun, maka akan AWS CLI memilih akun tersebut untuk

Anda secara otomatis dan melewati prompt. AWS Akun yang tersedia untuk Anda gunakan ditentukan oleh konfigurasi pengguna Anda di Pusat Identitas IAM.

```
There are 2 AWS accounts available to you.  
> DeveloperAccount, developer-account-admin@example.com (123456789011)  
   ProductionAccount, production-account-admin@example.com (123456789022)
```

Gunakan tombol panah untuk memilih akun yang ingin Anda gunakan. Karakter ">" di sebelah kiri menunjuk ke pilihan saat ini. Tekan ENTER untuk membuat pilihan Anda.

5. AWS CLI Konfirmasi pilihan akun Anda, dan menampilkan peran IAM yang tersedia untuk Anda di akun yang dipilih. Jika akun yang dipilih hanya mencantumkan satu peran, maka akan AWS CLI memilih peran tersebut untuk Anda secara otomatis dan melewati prompt. Peran yang tersedia untuk Anda gunakan ditentukan oleh konfigurasi pengguna Anda di Pusat Identitas IAM.

```
Using the account ID 123456789011  
There are 2 roles available to you.  
> ReadOnly  
   FullAccess
```

<ENTER>Gunakan tombol panah untuk memilih peran IAM yang ingin Anda gunakan dan tekan.

6. Tentukan [format output default](#), [default Wilayah AWS](#) untuk mengirim perintah, dan memberikan [nama untuk profil](#) sehingga Anda dapat mereferensikan profil ini dari antara semua yang ditentukan di komputer lokal. Dalam contoh berikut, pengguna memasukkan Wilayah default, format output default, dan nama profil. Jika Anda memiliki konfigurasi yang sudah ada sebelumnya, Anda dapat menekan <ENTER> untuk memilih nilai default apa pun yang ditampilkan di antara tanda kurung siku. Nama profil yang disarankan adalah nomor ID akun diikuti dengan garis bawah diikuti dengan nama peran.

```
CLI default client Region [None]: us-west-2<ENTER>  
CLI default output format [None]: json<ENTER>  
CLI profile name [123456789011_ReadOnly]: my-dev-profile<ENTER>
```

#### Note

Jika Anda menentukan default sebagai nama profil, profil ini menjadi yang digunakan setiap kali Anda menjalankan AWS CLI perintah dan tidak menentukan nama profil.

7. Pesan terakhir menjelaskan konfigurasi profil yang telah selesai.

```
To use this profile, specify the profile name using --profile, as shown:
```

```
aws s3 ls --profile my-dev-profile
```

8. Ini menghasilkan pembuatan `sso-session` bagian dan profil bernama di `~/.aws/config` yang terlihat seperti berikut:

```
[profile my-dev-profile]  
sso_session = my-sso  
sso_account_id = 123456789011  
sso_role_name = readOnly  
region = us-west-2  
output = json  
  
[sso-session my-sso]  
sso_region = us-east-1  
sso_start_url = https://my-sso-portal.awsapps.com/start  
sso_registration_scopes = sso:account:access
```

Anda sekarang dapat menggunakan ini `sso-session` dan profil untuk meminta kredensial yang diperbarui. Gunakan `aws sso login` perintah untuk meminta dan mengambil kredensial yang diperlukan untuk menjalankan perintah. Untuk petunjuk, lihat [Menggunakan profil bernama Pusat Identitas IAM](#).

## Konfigurasikan hanya `sso-session` bagian Anda dengan `aws configure sso-session` wizard

`aws configure sso-session` Perintah hanya memperbarui `sso-session` bagian dalam `~/.aws/config` file. Perintah ini dapat digunakan untuk membuat atau memperbarui sesi Anda. Ini berguna jika Anda sudah memiliki pengaturan konfigurasi yang ada dan ingin membuat `sso-session` konfigurasi baru atau mengedit yang ada.

Jalankan `aws configure sso-session` perintah dan berikan URL awal Pusat Identitas IAM Anda dan AWS Wilayah yang menghosting direktori Pusat Identitas.

```
$ aws configure sso-session  
SSO session name: my-sso  
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
```

```
SSO region [None]: us-east-1  
SSO registration scopes [None]: sso:account:access
```

Setelah memasukkan informasi Anda, pesan menjelaskan konfigurasi profil yang telah selesai.

```
Completed configuring SSO session: my-ss0  
Run the following to login and refresh access token for this session:
```

```
aws sso login --sso-session my-ss0
```

### Note

Jika Anda masuk ke pembaruan yang `sso-session` Anda perbarui, segarkan token Anda dengan menjalankan `aws sso login` perintah.

## Konfigurasi manual menggunakan **config** file

`sso-session` Bagian dari `config` file ini digunakan untuk mengelompokkan variabel konfigurasi untuk memperoleh token akses SSO, yang kemudian dapat digunakan untuk memperoleh AWS kredensial. Pengaturan berikut digunakan:

- (Diperlukan) [sso\\_start\\_url](#)
- (Diperlukan) [sso\\_region](#)
- [sso\\_account\\_id](#)
- [sso\\_role\\_name](#)
- [sso\\_registration\\_scopes](#)

Anda menentukan `sso-session` bagian dan mengaitkannya ke profil. `sso_region` dan `sso_start_url` harus diatur dalam `sso-session` bagian. Biasanya, `sso_account_id` dan `sso_role_name` harus diatur di `profile` bagian sehingga SDK dapat meminta kredensial SSO.

Contoh berikut mengonfigurasi SDK untuk meminta kredensial SSO dan mendukung penyegaran token otomatis:

```
[profile dev]  
sso_session = my-ss0  
sso_account_id = 111122223333
```



```
sso_role_name = SampleRole

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Ini juga memungkinkan sso-session konfigurasi untuk digunakan kembali di beberapa profil:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[profile prod]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole2

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Namun, `sso_account_id` dan `sso_role_name` tidak diperlukan untuk semua skenario konfigurasi token SSO. Jika aplikasi Anda hanya menggunakan AWS layanan yang mendukung otentikasi pembawa, maka AWS kredensi tradisional tidak diperlukan. Otentikasi pembawa adalah skema otentikasi HTTP yang menggunakan token keamanan yang disebut token pembawa. Dalam skenario ini, `sso_account_id` dan `sso_role_name` tidak diperlukan. Lihat panduan individual untuk AWS layanan Anda untuk menentukan apakah itu mendukung otorisasi token pembawa.

Selain itu, cakupan pendaftaran dapat dikonfigurasi sebagai bagian dari file. `sso-session` Cakupan adalah mekanisme di OAuth 2.0 untuk membatasi akses aplikasi ke akun pengguna. Aplikasi dapat meminta satu atau lebih cakupan, dan token akses yang dikeluarkan untuk aplikasi akan terbatas pada cakupan yang diberikan. Cakupan ini menentukan izin yang diminta untuk diotorisasi untuk klien OIDC terdaftar dan token akses yang diambil oleh klien. Contoh berikut menetapkan `sso_registration_scopes` untuk menyediakan akses untuk daftar akun/peran:

```
[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

Token otentikasi di-cache ke disk di bawah `~/ .aws/ssr/cache` direktori dengan nama file berdasarkan nama sesi.

## Konfigurasi lama yang tidak dapat disegarkan untuk AWS IAM Identity Center

Topik ini menjelaskan cara mengonfigurasi untuk mengautentikasi pengguna dengan AWS IAM Identity Center (IAM Identity Center) untuk mendapatkan kredensial untuk menjalankan AWS CLI perintah menggunakan metode lama. AWS CLI Saat menggunakan konfigurasi lama yang tidak dapat disegarkan, Anda perlu menyegarkan token secara manual karena kedaluwarsa secara berkala.

Saat menggunakan IAM Identity Center, Anda dapat masuk ke Active Directory, direktori IAM Identity Center bawaan, atau [IDP lain yang terhubung ke](#) IAM Identity Center. Anda dapat memetakan kredensial ini ke peran AWS Identity and Access Management (IAM) di mana Anda dapat menjalankan perintah. AWS CLI

Terlepas dari IDP mana yang Anda gunakan, IAM Identity Center mengabstraksikan perbedaan tersebut. Misalnya, Anda dapat menghubungkan Microsoft Azure AD seperti yang dijelaskan dalam artikel blog [Evolusi Berikutnya di Pusat Identitas IAM](#).

### Note

Untuk informasi tentang penggunaan autentikasi pembawa, yang tidak menggunakan ID akun dan peran, lihat [Menyiapkan untuk menggunakan AWS CLI dengan CodeCatalyst di CodeCatalyst](#) Panduan Pengguna Amazon.

Anda dapat mengonfigurasi satu atau beberapa [profil AWS CLI bernama](#) Anda untuk menggunakan peran dari Pusat Identitas IAM lama dengan cara berikut:

- [Secara otomatis](#), menggunakan `aws configure sso` perintah.
- [Secara manual](#), dengan mengedit `config` file yang menyimpan profil bernama.


## Prasyarat

- Instal AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#).

- Anda harus terlebih dahulu memiliki akses ke otentikasi SSO dalam IAM Identity Center. Pilih salah satu metode berikut untuk mengakses AWS kredensial Anda.

Saya tidak memiliki akses melalui IAM Identity Center

Ikuti petunjuk di [Memulai](#) di Panduan AWS IAM Identity Center Pengguna. Proses ini mengaktifkan IAM Identity Center, membuat pengguna administratif, dan menambahkan set izin hak istimewa yang paling tidak sesuai.

 Note

Buat set izin yang menerapkan izin hak istimewa paling sedikit. Sebaiknya gunakan set `PowerUserAccess` izin yang telah ditentukan sebelumnya, kecuali majikan Anda telah membuat set izin khusus untuk tujuan ini.

Keluar dari portal dan masuk lagi untuk melihat opsi Anda Akun AWS dan untuk Administrator atau `PowerUserAccess`. Pilih `PowerUserAccess` saat bekerja dengan SDK. Ini juga membantu Anda menemukan detail tentang akses terprogram.

Saya sudah memiliki akses AWS melalui penyedia identitas federasi yang dikelola oleh majikan saya (seperti Azure AD atau Okta)

Masuk AWS melalui portal penyedia identitas Anda. Jika Administrator Cloud Anda telah memberi Anda izin `PowerUserAccess` (pengembang), Anda akan melihat Akun AWS bahwa Anda memiliki akses ke dan izin Anda ditetapkan. Di samping nama set izin Anda, Anda melihat opsi untuk mengakses akun secara manual atau terprogram menggunakan set izin tersebut.

Implementasi kustom dapat menghasilkan pengalaman yang berbeda, seperti nama set izin yang berbeda. Jika Anda tidak yakin izin mana yang disetel untuk digunakan, hubungi tim TI Anda untuk mendapatkan bantuan.

Saya sudah memiliki akses AWS melalui portal AWS akses yang dikelola oleh majikan saya

Masuk AWS melalui portal AWS akses. Jika Administrator Cloud Anda telah memberi Anda izin `PowerUserAccess` (pengembang), Anda akan melihat Akun AWS bahwa Anda memiliki akses ke dan izin Anda ditetapkan. Di samping nama set izin Anda, Anda melihat opsi untuk mengakses akun secara manual atau terprogram menggunakan set izin tersebut.

Saya sudah memiliki akses AWS melalui penyedia identitas kustom federasi yang dikelola oleh majikan saya

Hubungi tim TI Anda untuk bantuan.

## Konfigurasi otomatis untuk konfigurasi lama

Untuk mengonfigurasi profil Pusat Identitas IAM ke AWS CLI

1. Jalankan `aws configure sso` perintah dan berikan URL awal Pusat Identitas IAM Anda dan AWS Wilayah yang menghosting direktori Pusat Identitas.

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1
```

2. AWS CLI Upaya untuk membuka browser default Anda dan memulai proses login untuk akun IAM Identity Center Anda.

```
SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.
```

Jika AWS CLI tidak dapat membuka browser, pesan berikut muncul dengan instruksi tentang cara memulai proses login secara manual.

```
Using a browser, open the following URL:
```

```
https://device.sso.us-west-2.amazonaws.com/
```

```
and enter the following code:
```

```
QCFK-N451
```

IAM Identity Center menggunakan kode untuk mengaitkan sesi IAM Identity Center dengan sesi Anda saat ini AWS CLI . Halaman browser IAM Identity Center meminta Anda untuk masuk dengan kredensial Pusat Identitas IAM Anda. Ini memberikan izin AWS CLI untuk mengambil dan menampilkan AWS akun dan peran yang Anda berwenang untuk digunakan dengan IAM Identity Center.

3. Selanjutnya, AWS CLI menampilkan AWS akun yang tersedia untuk Anda gunakan. Jika Anda berwenang untuk hanya menggunakan satu akun, maka akan AWS CLI memilih akun tersebut

untuk Anda secara otomatis dan melewati prompt. AWS Akun yang tersedia untuk Anda digunakan ditentukan oleh konfigurasi pengguna Anda di Pusat Identitas IAM.

```
There are 2 AWS accounts available to you.  
> DeveloperAccount, developer-account-admin@example.com (123456789011)  
   ProductionAccount, production-account-admin@example.com (123456789022)
```

Gunakan tombol panah untuk memilih akun yang ingin Anda gunakan dengan profil ini. Karakter “>” di sebelah kiri menunjuk ke pilihan saat ini. Tekan ENTER untuk membuat pilihan Anda.

4. Selanjutnya, AWS CLI konfirmasi pilihan akun Anda, dan menampilkan peran IAM yang tersedia untuk Anda di akun yang dipilih. Jika akun yang dipilih hanya mencantumkan satu peran, maka akan AWS CLI memilih peran tersebut untuk Anda secara otomatis dan melewati prompt. Peran yang tersedia untuk Anda gunakan ditentukan oleh konfigurasi pengguna Anda di Pusat Identitas IAM.

```
Using the account ID 123456789011  
There are 2 roles available to you.  
> ReadOnly  
   FullAccess
```

<ENTER>Gunakan tombol panah untuk memilih peran IAM yang ingin Anda gunakan dengan profil ini dan tekan.

5. AWS CLI Konfirmasi pemilihan peran Anda.

```
Using the role name "ReadOnly"
```

6. Selesaikan konfigurasi profil Anda dengan menentukan format output default, default Wilayah AWS untuk mengirim perintah, dan memberikan [nama untuk profil](#) sehingga Anda dapat mereferensikan profil ini dari antara semua yang ditentukan di komputer lokal. Dalam contoh berikut, pengguna memasukkan Wilayah default, format output default, dan nama profil. Anda dapat menekan <ENTER> untuk memilih nilai default apa pun yang ditampilkan di antara tanda kurung siku. Nama profil yang disarankan adalah nomor ID akun diikuti dengan garis bawah diikuti dengan nama peran.

```
CLI default client Region [None]: us-west-2<ENTER>  
CLI default output format [None]: json<ENTER>  
CLI profile name [123456789011_ReadOnly]: my-dev-profile<ENTER>
```

**Note**

Jika Anda menentukan default sebagai nama profil, profil ini menjadi yang digunakan setiap kali Anda menjalankan AWS CLI perintah dan tidak menentukan nama profil.

7. Pesan terakhir menjelaskan konfigurasi profil yang telah selesai.

Untuk menggunakan profil ini, tentukan nama profil menggunakan `--profile`, seperti yang ditunjukkan:

```
aws s3 ls --profile my-dev-profile
```

8. Entri contoh sebelumnya akan menghasilkan profil bernama `~/.aws/config` yang terlihat seperti contoh berikut.

```
[profile my-dev-profile]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
sso_account_id = 123456789011
sso_role_name = readOnly
region = us-west-2
output = json
```

Pada titik ini, Anda memiliki profil yang dapat Anda gunakan untuk meminta kredensial sementara. Anda harus menggunakan `aws sso login` perintah untuk benar-benar meminta dan mengambil kredensial sementara yang diperlukan untuk menjalankan perintah. Untuk petunjuk, lihat [Menggunakan profil bernama Pusat Identitas IAM](#).

## Konfigurasi manual untuk konfigurasi lama

Penyegaran token otomatis tidak didukung menggunakan konfigurasi lama yang tidak dapat disegarkan. Sebaiknya gunakan konfigurasi token SSO.

Untuk menambahkan dukungan Pusat Identitas IAM secara manual ke profil bernama, Anda harus menambahkan kunci dan nilai berikut ke definisi profil dalam file `~/.aws/config` (Linux atau macOS) `%USERPROFILE%/.aws/config` atau (Windows).

- [sso\\_start\\_url](#)

- [sso\\_region](#)
- [sso\\_account\\_id](#)
- [sso\\_role\\_name](#)

Anda dapat menyertakan kunci dan nilai lain yang valid dalam `.aws/config` file, seperti [region](#), [output](#), atau [s3](#). Untuk mencegah kesalahan, jangan sertakan nilai terkait kredensi apa pun, seperti [role\\_arn](#) atau [aws\\_secret\\_access\\_key](#).

Berikut ini adalah contoh profil IAM Identity Center di `.aws/config`:

```
[profile my-sso-profile]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-west-2
sso_account_id = 111122223333
sso_role_name = SSOReadOnlyRole
region = us-west-2
output = json
```

Profil Anda untuk kredensi sementara sudah lengkap.

Untuk menjalankan perintah, Anda harus terlebih dahulu menggunakan `aws sso login` perintah untuk meminta dan mengambil kredensi sementara Anda. Untuk petunjuk, lihat bagian selanjutnya, [Menggunakan profil bernama Pusat Identitas IAM](#). Token otentikasi di-cache ke disk di bawah `~/.aws/sso/cache` direktori dengan nama file berdasarkan file `sso_start_url`

## Menggunakan profil bernama Pusat Identitas IAM

Topik ini menjelaskan cara menggunakan untuk mengautentikasi pengguna dengan AWS IAM Identity Center (IAM Identity Center) untuk mendapatkan kredensi untuk menjalankan perintah. AWS CLI AWS CLI

### Note

Apakah kredensi Anda bersifat sementara atau secara otomatis menyegarkan tergantung pada cara Anda mengonfigurasi profil Anda sebelumnya.

Topik

- [Prasyarat](#)

- [Masuk dan mendapatkan kredensi](#)
- [Menjalankan perintah dengan profil IAM Identity Center](#)
- [Keluar dari sesi IAM Identity Center](#)

## Prasyarat

Anda telah mengonfigurasi profil Pusat Identitas IAM. Lihat [the section called “Konfigurasi penyesuaian token otomatis”](#) dan [the section called “Konfigurasi lawas yang tidak dapat disegarkan”](#) untuk informasi lebih lanjut.

## Masuk dan mendapatkan kredensi

### Note

Proses masuk dapat meminta Anda untuk mengizinkan AWS CLI akses ke data Anda. Karena AWS CLI dibangun di atas SDK untuk Python, pesan izin mungkin berisi variasi nama. `botocore`

Setelah mengonfigurasi profil bernama, Anda dapat memanggilnya untuk meminta kredensialnya. AWS Sebelum Anda dapat menjalankan perintah AWS CLI layanan, Anda harus mengambil dan menyimpan satu set kredensi. Untuk mendapatkan kredensi ini, jalankan perintah berikut.

```
$ aws sso login --profile my-dev-profile
```

Ini AWS CLI membuka browser default Anda dan memverifikasi log masuk Pusat Identitas IAM Anda.

```
SSO authorization page has automatically been opened in your default browser.  
Follow the instructions in the browser to complete this authorization request.  
Successfully logged into Start URL: https://my-sso-portal.awsapps.com/start
```

Jika saat ini Anda tidak masuk ke Pusat Identitas IAM, Anda harus memberikan kredensial Pusat Identitas IAM Anda.

Jika tidak AWS CLI dapat membuka browser Anda, itu meminta Anda untuk membukanya sendiri dan memasukkan kode yang ditentukan.

```
$ aws sso login --profile my-dev-profile
```



Using a browser, open the following URL:

```
https://device.sso.us-west-2.amazonaws.com/
```

and enter the following code:

```
QCFK-N451
```

AWS CLIMembuka browser default Anda (atau Anda secara manual membuka browser pilihan Anda) ke halaman yang ditentukan, dan masukkan kode yang disediakan. Halaman web kemudian meminta Anda untuk kredensial Pusat Identitas IAM Anda.

Kredensial sesi Pusat Identitas IAM Anda di-cache. Jika kredensial ini bersifat sementara, itu termasuk stempel waktu kedaluwarsa dan ketika mereka kedaluwarsa, AWS CLI meminta Anda untuk masuk ke IAM Identity Center lagi.

Jika kredensial Pusat Identitas IAM Anda valid, Anda akan AWS CLI menggunakannya untuk mengambil AWS kredensial secara aman untuk peran IAM yang ditentukan dalam profil.

```
Welcome, you have successfully signed-in to the AWS-CLI.
```

Anda juga dapat menentukan `sso-session` profil mana yang akan digunakan saat masuk menggunakan `--sso-session` parameter `aws sso login` perintah.

```
$ aws sso login --sso-session my-dev-session
```

```
Attempting to automatically open the SSO authorization page in your default browser.  
If the browser does not open or you wish to use a different device to authorize this  
request, open the following URL:
```

```
https://device.sso.us-west-2.amazonaws.com/
```

```
and enter the following code:
```

```
QCFK-N451
```

```
Successfully logged into Start URL: https://cli-reinvent.awsapps.com/start
```

## Menjalankan perintah dengan profil IAM Identity Center

Anda dapat menggunakan kredensial ini untuk menjalankan AWS CLI perintah dengan profil bernama terkait. Contoh berikut menunjukkan bahwa perintah dijalankan di bawah peran yang diasumsikan yang merupakan bagian dari akun yang ditentukan.

```
$ aws sts get-caller-identity --profile my-dev-profile
{
  "UserId": "AROA12345678901234567:test-user@example.com",
  "Account": "123456789011",
  "Arn": "arn:aws:sts::123456789011:assumed-role/
AWSPeregrine_readOnly_12321abc454d123/test-user@example.com"
}
```

Selama Anda masuk ke IAM Identity Center dan kredensial cache tersebut tidak kedaluwarsa, kredensial yang kedaluwarsa akan diperbarui AWS CLI secara otomatis bila diperlukan. Namun, jika kredensial Pusat Identitas IAM Anda kedaluwarsa, Anda harus memperbaruinya secara eksplisit dengan masuk ke akun Pusat Identitas IAM Anda lagi.

```
$ aws s3 ls --profile my-sso-profile
Your short-term credentials have expired. Please sign-in to renew your credentials
SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.
```

## Keluar dari sesi IAM Identity Center

Ketika Anda selesai menggunakan profil Pusat Identitas IAM Anda, Anda dapat memilih untuk tidak melakukan apa pun dan membiarkan kredensial AWS sementara dan kredensial Pusat Identitas IAM Anda kedaluwarsa. Namun, Anda juga dapat memilih untuk menjalankan perintah berikut untuk segera menghapus semua kredensial yang di-cache di folder cache kredensial SSO dan semua kredensial AWS sementara yang didasarkan pada kredensial Pusat Identitas IAM. Ini membuat kredensial tersebut tidak tersedia untuk digunakan untuk perintah future apa pun.

```
$ aws sso logout
Successfully signed out of all SSO profiles.
```

Jika nanti Anda ingin menjalankan perintah dengan salah satu profil Pusat Identitas IAM Anda, Anda harus menjalankan kembali `aws sso login` perintah (lihat bagian sebelumnya) dan tentukan profil yang akan digunakan.

## Otentikasi dengan kredensial jangka pendek

Sebaiknya konfigurasi SDK atau alat Anda untuk menggunakan [otentikasi IAM Identity Center](#) dengan opsi durasi sesi yang diperpanjang. Namun, Anda dapat menyalin dan menggunakan



# Gunakan peran IAM dalam AWS CLI

[Peran AWS Identity and Access Management \(IAM\)](#) adalah alat otorisasi yang memungkinkan pengguna mendapatkan izin tambahan (atau berbeda), atau mendapatkan izin untuk melakukan tindakan di akun yang berbeda. AWS

## Topik

- [Prasyarat](#)
- [Ikhtisar penggunaan peran IAM](#)
- [Mengkonfigurasi dan menggunakan peran](#)
- [Menggunakan otentikasi multi-faktor](#)
- [Peran lintas akun dan ID eksternal](#)
- [Menentukan nama sesi peran untuk audit yang lebih mudah](#)
- [Asumsikan peran dengan identitas web](#)
- [Membersihkan kredensial yang di-cache](#)

## Prasyarat

Untuk menjalankan `iam` perintah, Anda perlu menginstal dan mengkonfigurasi AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#).

## Ikhtisar penggunaan peran IAM

Anda dapat mengonfigurasi AWS Command Line Interface (AWS CLI) untuk menggunakan peran IAM dengan menentukan profil untuk peran dalam file. `~/.aws/config`

Contoh berikut menunjukkan profil peran bernama `marketingadmin`. Jika Anda menjalankan perintah dengan `--profile marketingadmin` (atau menentukannya dengan [variabel lingkungan `AWS\_PROFILE`](#)), akan AWS CLI menggunakan kredensial yang ditentukan dalam profil terpisah `user1` untuk mengambil peran dengan Amazon Resource Name (ARN). `arn:aws:iam::123456789012:role/marketingadminrole` Anda dapat menjalankan operasi apa pun yang diizinkan oleh izin yang ditetapkan untuk peran tersebut.

```
[profile marketingadmin]  
role_arn = arn:aws:iam::123456789012:role/marketingadminrole  
source_profile = user1
```

Anda kemudian dapat menentukan `source_profile` yang menunjuk ke profil bernama terpisah yang berisi kredensial pengguna dengan izin untuk menggunakan peran tersebut. Pada contoh sebelumnya, `marketingadmin` profil menggunakan kredensial di profil. `user1` Ketika Anda menentukan bahwa AWS CLI perintah adalah untuk menggunakan profil `marketingadmin`, AWS CLI secara otomatis mencari kredensial untuk `user1` profil tertaut dan menggunakannya untuk meminta kredensial sementara untuk peran IAM yang ditentukan. CLI menggunakan `AssumeRole` operasi [sts:](#) di latar belakang untuk mencapai ini. Kredensial sementara tersebut kemudian digunakan untuk menjalankan perintah yang diminta AWS CLI. Peran yang ditentukan harus telah melampirkan kebijakan izin IAM yang memungkinkan AWS CLI perintah yang diminta untuk dijalankan.

Untuk menjalankan AWS CLI perintah dari dalam instans Amazon Elastic Compute Cloud (Amazon EC2) atau container Amazon Elastic Container Service (Amazon ECS), Anda dapat menggunakan peran IAM yang dilampirkan ke profil instance atau container. Jika Anda menentukan tidak ada profil atau tidak menyetel variabel lingkungan, peran tersebut akan digunakan secara langsung. Hal ini memungkinkan Anda untuk menghindari menyimpan kunci akses berumur panjang pada instans Anda. Anda juga dapat menggunakan peran instance atau kontainer tersebut hanya untuk mendapatkan kredensial untuk peran lain. Untuk melakukan ini, Anda menggunakan `credential_source` (bukan `source_profile`) untuk menentukan cara menemukan kredensialnya. `credential_source` Atribut mendukung nilai-nilai berikut:

- `Environment`— Mengambil kredensi sumber dari variabel lingkungan.
- `Ec2InstanceMetadata`— Menggunakan peran IAM yang dilampirkan ke profil instans Amazon EC2.
- `EcsContainer`— Menggunakan peran IAM yang dilampirkan ke wadah Amazon ECS.

Contoh berikut menunjukkan `marketingadminrole` peran yang sama yang digunakan dengan mereferensikan profil instans Amazon EC2.

```
[profile marketingadmin]
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
credential_source = Ec2InstanceMetadata
```

Ketika Anda memanggil peran, Anda memiliki opsi tambahan yang dapat Anda perlukan, seperti penggunaan otentikasi multi-faktor dan ID Eksternal (digunakan oleh perusahaan pihak ketiga untuk mengakses sumber daya klien mereka). Anda juga dapat menentukan nama sesi peran unik yang dapat lebih mudah diaudit di AWS CloudTrail log.

## Mengkonfigurasi dan menggunakan peran

Saat Anda menjalankan perintah menggunakan profil yang menentukan peran IAM, akan AWS CLI menggunakan kredensial profil sumber untuk memanggil AWS Security Token Service (AWS STS) dan meminta kredensial sementara untuk peran yang ditentukan. Pengguna di profil sumber harus memiliki izin `sts:assume-role` untuk memanggil peran dalam profil yang ditentukan. Peran harus memiliki hubungan kepercayaan yang memungkinkan pengguna di profil sumber untuk menggunakan peran tersebut. Proses pengambilan dan kemudian menggunakan kredensial sementara untuk suatu peran sering disebut sebagai asumsi peran.

Anda dapat membuat peran di IAM dengan izin yang ingin diasumsikan pengguna dengan mengikuti prosedur di bawah [Membuat Peran untuk Mendelegasikan Izin ke pengguna IAM di Panduan Pengguna](#). AWS Identity and Access Management Jika peran dan pengguna profil sumber berada di akun yang sama, Anda dapat memasukkan ID akun Anda sendiri saat mengonfigurasi hubungan kepercayaan peran.

Setelah membuat peran, modifikasi hubungan kepercayaan untuk memungkinkan pengguna untuk mengasumsikan itu.

Contoh berikut menunjukkan kebijakan kepercayaan yang dapat Anda lampirkan ke peran. Kebijakan ini memungkinkan peran diasumsikan oleh setiap pengguna di akun 123456789012, jika administrator akun tersebut secara eksplisit memberikan izin kepada pengguna. `sts:AssumeRole`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Kebijakan kepercayaan sebenarnya tidak memberikan izin. Administrator akun harus mendelegasikan izin untuk mengambil peran tersebut kepada pengguna individu dengan melampirkan kebijakan dengan izin yang sesuai. Contoh berikut menunjukkan kebijakan yang dapat Anda lampirkan ke pengguna yang memungkinkan pengguna untuk mengambil

`marketingadminrole` peran saja. Untuk informasi selengkapnya tentang pemberian akses pengguna untuk mengambil peran, lihat [Memberikan Izin Pengguna untuk Beralih Peran](#) di Panduan Pengguna IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::123456789012:role/marketingadminrole"
    }
  ]
}
```

Pengguna tidak perlu memiliki izin tambahan untuk menjalankan AWS CLI perintah menggunakan profil peran. Sebaliknya, izin untuk menjalankan perintah berasal dari yang dilampirkan pada peran. Anda melampirkan kebijakan izin ke peran untuk menentukan tindakan mana yang dapat dilakukan terhadap AWS sumber daya yang mana. Untuk informasi selengkapnya tentang melampirkan izin ke peran (yang bekerja secara identik dengan pengguna), lihat [Mengubah Izin untuk pengguna IAM di Panduan Pengguna IAM](#).

Sekarang setelah Anda memiliki profil peran, izin peran, hubungan kepercayaan peran, dan izin pengguna yang dikonfigurasi dengan benar, Anda dapat menggunakan peran di baris perintah dengan menjalankan opsi. `--profile` Misalnya, berikut ini memanggil `ls` perintah Amazon S3 menggunakan izin yang dilampirkan pada `marketingadmin` peran seperti yang ditentukan oleh contoh di awal topik ini.

```
$ aws s3 ls --profile marketingadmin
```

Untuk menggunakan peran untuk beberapa panggilan, Anda dapat mengatur variabel `AWS_PROFILE` lingkungan untuk sesi saat ini dari baris perintah. Sementara variabel lingkungan didefinisikan, Anda tidak perlu menentukan `--profile` opsi pada setiap perintah.

Linux atau macOS

```
$ export AWS_PROFILE=marketingadmin
```

Jendela

```
C:\> setx AWS_PROFILE marketingadmin
```

Untuk informasi selengkapnya tentang mengonfigurasi pengguna dan peran, lihat [Pengguna dan Grup](#) dan [Peran](#) di Panduan Pengguna IAM.

## Menggunakan otentikasi multi-faktor

Untuk keamanan tambahan, Anda dapat meminta pengguna menyediakan kunci satu kali yang dihasilkan dari perangkat multi-faktor otentikasi (MFA), perangkat U2F, atau aplikasi seluler saat mereka mencoba melakukan panggilan menggunakan profil peran.

Pertama, Anda dapat memilih untuk memodifikasi hubungan kepercayaan pada peran IAM untuk meminta MFA. Ini mencegah siapa pun menggunakan peran tanpa terlebih dahulu mengotentikasi dengan menggunakan MFA. Sebagai contoh, lihat `Condition` baris dalam contoh berikut. Kebijakan ini memungkinkan pengguna yang diberi nama `anika` untuk mengambil peran yang dilampirkan kebijakan, tetapi hanya jika mereka mengotentikasi dengan menggunakan MFA.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": { "AWS": "arn:aws:iam::123456789012:user/anika" },
      "Action": "sts:AssumeRole",
      "Condition": { "Bool": { "aws:multifactorAuthPresent": true } }
    }
  ]
}
```

Selanjutnya, tambahkan baris ke profil peran yang menentukan ARN perangkat MFA pengguna. Entri config file contoh berikut menunjukkan dua profil peran yang keduanya menggunakan kunci akses bagi pengguna `anika` untuk meminta kredensial sementara untuk peran tersebut. `cli-role` Pengguna `anika` memiliki izin untuk mengambil peran, yang diberikan oleh kebijakan kepercayaan peran.

```
[profile role-without-mfa]
region = us-west-2
role_arn= arn:aws:iam::128716708097:role/cli-role
source_profile=cli-user
```



```
[profile role-with-mfa]
region = us-west-2
role_arn= arn:aws:iam::128716708097:role/cli-role
source_profile = cli-user
mfa_serial = arn:aws:iam::128716708097:mfa/cli-user

[profile cli-user]
region = us-west-2
output = json
```

`mfa_serial` Pengaturan dapat mengambil ARN, seperti yang ditunjukkan, atau nomor seri token MFA perangkat keras.

Profil pertama, `role-without-mfa`, tidak memerlukan MFA. Namun, karena contoh kebijakan kepercayaan sebelumnya yang dilampirkan pada peran memerlukan MFA, setiap upaya untuk menjalankan perintah dengan profil ini gagal.

```
$ aws iam list-users --profile role-without-mfa
```

```
An error occurred (AccessDenied) when calling the AssumeRole operation: Access denied
```

Entri profil kedua, `role-with-mfa`, mengidentifikasi perangkat MFA untuk digunakan. Ketika pengguna mencoba menjalankan AWS CLI perintah dengan profil ini, pengguna AWS CLI meminta pengguna untuk memasukkan kata sandi satu kali (OTP) yang disediakan perangkat MFA. Jika otentikasi MFA berhasil, perintah melakukan operasi yang diminta. OTP tidak ditampilkan di layar.

```
$ aws iam list-users --profile role-with-mfa
Enter MFA code for arn:aws:iam::123456789012:mfa/cli-user:
{
  "Users": [
    {
      ...
```

## Peran lintas akun dan ID eksternal

Anda dapat mengaktifkan pengguna untuk menggunakan peran yang dimiliki oleh akun yang berbeda dengan mengonfigurasi peran sebagai peran lintas akun. Selama pembuatan peran, setel tipe peran ke `AWS` akun lain, seperti yang dijelaskan dalam [Membuat Peran untuk Mendelegasikan Izin ke pengguna IAM](#). Secara opsional, pilih `Memerlukan MFA`. `Memerlukan MFA` mengkonfigurasi kondisi

yang sesuai dalam hubungan kepercayaan, seperti yang dijelaskan dalam [Menggunakan otentikasi multi-faktor](#)

Jika Anda menggunakan [ID eksternal](#) untuk memberikan kontrol tambahan atas siapa yang dapat menggunakan peran di seluruh akun, Anda juga harus menambahkan `external_id` parameter ke profil peran. Anda biasanya menggunakan ini hanya ketika akun lain dikendalikan oleh seseorang di luar perusahaan atau organisasi Anda.

```
[profile crossaccountrole]
role_arn = arn:aws:iam::234567890123:role/SomeRole
source_profile = default
mfa_serial = arn:aws:iam::123456789012:mfa/saanvi
external_id = 123456
```

## Menentukan nama sesi peran untuk audit yang lebih mudah

Ketika banyak individu berbagi peran, audit menjadi lebih dari sebuah tantangan. Anda ingin mengaitkan setiap operasi yang dipanggil dengan individu yang memanggil tindakan. Namun, ketika individu menggunakan peran, asumsi peran oleh individu adalah tindakan terpisah dari pemanggilan operasi, dan Anda harus mengkorelasikan keduanya secara manual.

Anda dapat menyederhanakan ini dengan menentukan nama sesi peran unik saat pengguna mengambil peran. Anda melakukan ini dengan menambahkan `role_session_name` parameter ke setiap profil bernama dalam `config` file yang menentukan peran. `role_session_name` Nilai diteruskan ke `AssumeRole` operasi dan menjadi bagian dari ARN untuk sesi peran. Ini juga termasuk dalam AWS CloudTrail log untuk semua operasi yang dicatat.

Misalnya, Anda dapat membuat profil berbasis peran sebagai berikut.

```
[profile namedsessionrole]
role_arn = arn:aws:iam::234567890123:role/SomeRole
source_profile = default
role_session_name = Session_Maria_Garcia
```

Hal ini mengakibatkan sesi peran memiliki ARN berikut.

```
arn:aws:iam::234567890123:assumed-role/SomeRole/Session_Maria_Garcia
```

Selain itu, semua AWS CloudTrail log menyertakan nama sesi peran dalam informasi yang diambil untuk setiap operasi.

## Asumsikan peran dengan identitas web

Anda dapat mengonfigurasi profil untuk menunjukkan bahwa AWS CLI harus mengambil peran menggunakan [federasi identitas web dan Open ID Connect \(OIDC\)](#). Saat Anda menentukan ini di profil, secara otomatis AWS CLI membuat AWS STS `AssumeRoleWithWebIdentity` panggilan yang sesuai untuk Anda.

### Note

Saat Anda menentukan profil yang menggunakan peran IAM, akan AWS CLI membuat panggilan yang sesuai untuk mengambil kredensial sementara. Kredensial ini disimpan di `~/.aws/cli/cache` AWS CLI Perintah selanjutnya yang menentukan profil yang sama menggunakan kredensial sementara yang di-cache hingga kedaluwarsa. Pada saat itu, secara otomatis AWS CLI menyegarkan kredensialnya.

Untuk mengambil dan menggunakan kredensial sementara menggunakan federasi identitas web, Anda dapat menentukan nilai konfigurasi berikut di profil bersama.

### [role\\_arn](#)

Menentukan ARN dari peran yang akan diasumsikan.

### [web\\_identity\\_token\\_file](#)

Menentukan path ke file yang berisi token akses OAuth 2.0 atau token OpenID Connect ID yang disediakan oleh penyedia identitas. AWS CLI Memuat file ini dan meneruskan kontennya sebagai `WebIdentityToken` argumen `AssumeRoleWithWebIdentity` operasi.

### [role\\_session\\_name](#)

Menentukan nama opsional diterapkan untuk sesi `assume-role` ini.

Berikut ini adalah contoh dari jumlah minimal konfigurasi yang diperlukan untuk mengkonfigurasi peran asumsi dengan profil identitas web.

```
# In ~/.aws/config

[profile web-identity]
role_arn=arn:aws:iam:123456789012:role/RoLeNameToAssume
```

```
web_identity_token_file=/path/to/a/token
```

Anda juga dapat menyediakan konfigurasi ini dengan menggunakan [variabel lingkungan](#).

#### AWS\_ROLE\_ARN

ARN dari peran yang harus diasumsikan.

#### AWS\_WEB\_IDENTITY\_TOKEN\_FILE

Jalur ke file token identitas web.

#### AWS\_ROLE\_SESSION\_NAME

Nama yang diterapkan pada sesi peran asumsi ini.

#### Note

Variabel lingkungan ini saat ini hanya berlaku untuk peran asumsi dengan penyedia identitas web. Mereka tidak berlaku untuk konfigurasi penyedia peran asumsi umum.

## Membersihkan kredensial yang di-cache

Saat Anda menggunakan peran, kredensial sementara akan disimpan di AWS CLI cache secara lokal hingga mereka kedaluwarsa. Lain kali Anda mencoba menggunakannya, AWS CLI upaya untuk memperbaruinya atas nama Anda.

Jika kredensial sementara peran Anda [dicabut](#), kredensialnya tidak diperbarui secara otomatis, dan upaya untuk menggunakannya gagal. Namun, Anda dapat menghapus cache untuk memaksa AWS CLI untuk mengambil kredensial baru.

#### Linux atau macOS

```
$ rm -r ~/.aws/cli/cache
```

#### Jendela

```
C:\> del /s /q %UserProfile%\aws\cli\cache
```

# Otentikasi dengan kredensi pengguna IAM

## Warning

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti [AWS IAM Identity Center](#).

Bagian ini menjelaskan cara mengkonfigurasi pengaturan dasar dengan pengguna IAM. Ini termasuk kredensi keamanan Anda menggunakan `config` dan `credentials` file. Untuk melihat instruksi konfigurasi AWS IAM Identity Center, lihat [the section called “Autentikasi Pusat Identitas IAM”](#).

## Topik

- [Langkah 1: Buat pengguna IAM Anda](#)
- [Langkah 2: Dapatkan kunci akses Anda](#)
- [Konfigurasi AWS CLI](#)
  - [Menggunakan `aws configure`](#)
  - [Mengimpor kunci akses melalui file CSV](#)
  - [Langsung mengedit `config` dan `credentials` file](#)

## Langkah 1: Buat pengguna IAM Anda

Buat pengguna IAM Anda dengan mengikuti prosedur [Membuat pengguna IAM \(konsol\)](#) di Panduan Pengguna IAM.

- Untuk opsi Izin, pilih Lampirkan kebijakan secara langsung untuk mengetahui cara Anda ingin menetapkan izin kepada pengguna ini.
- Sebagian besar tutorial SDK “Memulai” menggunakan layanan Amazon S3 sebagai contoh. Untuk menyediakan aplikasi Anda dengan akses penuh ke Amazon S3, pilih `AmazonS3FullAccess` kebijakan untuk melampirkan ke pengguna ini.

## Langkah 2: Dapatkan kunci akses Anda

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi konsol IAM, pilih Pengguna dan kemudian pilih pengguna **User** `name` yang Anda buat sebelumnya.
3. Pada halaman pengguna, pilih halaman Security credentials. Kemudian, di bawah tombol Access, pilih Create Access Key.
4. Untuk Buat tombol akses Langkah 1, pilih Antarmuka Baris Perintah (CLI).
5. Untuk Buat tombol akses Langkah 2, masukkan tag opsional dan pilih Berikutnya.
6. Untuk Buat kunci akses Langkah 3, pilih Unduh file.csv untuk menyimpan .csv file dengan kunci akses pengguna IAM dan kunci akses rahasia Anda. Anda memerlukan informasi ini untuk nanti.
7. Pilih Selesai.

## Konfigurasi AWS CLI

Untuk penggunaan umum, AWS CLI kebutuhan informasi berikut:

- ID kunci akses
- Kunci akses rahasia
- AWS Wilayah
- Format output

AWS CLI Menyimpan informasi ini dalam profil (kumpulan pengaturan) yang disebutkan default dalam `credentials` file. Secara default, informasi dalam profil ini digunakan saat Anda menjalankan AWS CLI perintah yang tidak secara eksplisit menentukan profil yang akan digunakan. Untuk informasi lebih lanjut tentang file `credentials`, lihat [Konfigurasi dan pengaturan file kredensi](#).

Untuk mengkonfigurasi AWS CLI, gunakan salah satu prosedur berikut:

Topik

- [Menggunakan `aws configure`](#)
- [Mengimpor kunci akses melalui file.CSV](#)
- [Langsung mengedit `config` dan `credentials` file](#)

## Menggunakan `aws configure`

Untuk penggunaan umum, `aws configure` perintah adalah cara tercepat untuk mengatur AWS CLI instalasi Anda. Panduan konfigurasi ini meminta Anda untuk setiap informasi yang Anda butuhkan untuk memulai. Kecuali ditentukan lain dengan menggunakan `--profile` opsi, AWS CLI menyimpan informasi ini di default profil.

Contoh berikut mengkonfigurasi default profil menggunakan nilai sampel. Ganti dengan nilai Anda sendiri seperti yang dijelaskan di bagian berikut.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

Contoh berikut mengkonfigurasi profil bernama `userprod` menggunakan nilai sampel. Ganti dengan nilai Anda sendiri seperti yang dijelaskan di bagian berikut.

```
$ aws configure --profile userprod
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

## Mengimpor kunci akses melalui file.CSV

Alih-alih menggunakan `aws configure` untuk memasukkan kunci akses, Anda dapat mengimpor `.csv` file teks biasa yang Anda unduh setelah Anda membuat kunci akses Anda.

`.csv`File harus berisi header berikut.

- Nama Pengguna - Kolom ini harus ditambahkan ke kolom Anda `.csv`. Ini digunakan untuk membuat nama profil yang digunakan dalam `credentials` file config dan ketika Anda mengimpor.
- ID kunci akses
- Kunci akses rahasia

**Note**

Selama pembuatan kunci akses awal, setelah Anda menutup kotak dialog Unduh file.csv, Anda tidak dapat mengakses kunci akses rahasia Anda setelah Anda menutup kotak dialog. Jika Anda memerlukan .csv file, Anda harus membuatnya sendiri dengan header yang diperlukan dan informasi kunci akses yang Anda simpan. Jika Anda tidak memiliki akses ke informasi kunci akses Anda, Anda perlu membuat kunci akses baru.

Untuk mengimpor .csv file, gunakan `aws configure import` perintah dengan `--csv` opsi sebagai berikut:

```
$ aws configure import --csv file://credentials.csv
```

Untuk informasi selengkapnya, lihat [aws\\_configure\\_import](#).

## Langsung mengedit `config` dan `credentials` file

Untuk langsung mengedit `config` dan `credentials` file, lakukan hal berikut.

1. Buat atau buka AWS `credentials` file bersama. File ini ada `~/.aws/credentials` di sistem Linux dan macOS, dan `%USERPROFILE%\aws\credentials` di Windows. Untuk informasi selengkapnya, lihat [the section called “Konfigurasi dan pengaturan file kredensi”](#).
2. Tambahkan teks berikut ke `credentials` file bersama. Ganti nilai sampel dalam .csv file yang Anda unduh sebelumnya dan simpan file.

```
[default]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

## Menggunakan kredensyal untuk metadata instans Amazon EC2

Saat menjalankan instans Amazon Elastic Compute Cloud (Amazon EC2) AWS CLI dari dalam instans Amazon Elastic Compute Cloud (Amazon EC2), Anda dapat menyederhanakan penyediaan kredensyal ke perintah Anda. Setiap instans Amazon EC2 berisi metadata yang AWS CLI dapat langsung kueri untuk kredensyal sementara. Ketika peran IAM dilampirkan ke instance, AWS CLI secara otomatis dan aman mengambil kredensyal dari metadata instance.



Untuk menonaktifkan layanan ini, gunakan variabel lingkungan [AWS\\_EC2\\_METADATA\\_DISABLED](#).

Topik

- [Prasyarat](#)
- [Mengonfigurasi profil untuk metadata Amazon EC2](#)

## Prasyarat

Untuk menggunakan kredensial Amazon EC2 dengan AWS CLI, Anda harus menyelesaikan yang berikut ini:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Anda memahami file konfigurasi dan profil bernama. Untuk informasi selengkapnya, lihat [Konfigurasi dan pengaturan file kredensi](#).
- Anda telah membuat peran AWS Identity and Access Management (IAM) yang memiliki akses ke sumber daya yang dibutuhkan, dan melampirkan peran tersebut ke instans Amazon EC2 saat meluncurkannya. Untuk informasi selengkapnya, lihat [kebijakan IAM untuk Amazon EC2](#) di Panduan Pengguna Amazon EC2 [dan Pemberian Aplikasi yang Berjalan di Instans Amazon EC2](#) Akses ke Sumber Daya di Panduan Pengguna IAM. AWS

## Mengonfigurasi profil untuk metadata Amazon EC2

Untuk menentukan bahwa Anda ingin menggunakan kredensial yang tersedia di profil instans Amazon EC2 hosting, gunakan sintaks berikut di profil bernama di file konfigurasi Anda. Lihat langkah-langkah berikut untuk petunjuk lebih lanjut.

```
[profile profilename]  
role_arn = arn:aws:iam::123456789012:role/rolename  
credential_source = Ec2InstanceMetadata  
region = region
```

1. Buat profil di file konfigurasi Anda.

```
[profile profilename]
```

2. Tambahkan peran ARN IAM Anda yang memiliki akses ke sumber daya yang dibutuhkan.

```
role_arn = arn:aws:iam::123456789012:role/rolename
```

3. Tentukan `Ec2InstanceMetadata` sebagai sumber kredensial Anda.

```
credential_source = Ec2InstanceMetadata
```

4. Atur Wilayah Anda.

```
region = region
```

## Contoh

Contoh berikut mengasumsikan *marketingadminrole* peran dan menggunakan *us-west-2* Wilayah dalam profil instans Amazon EC2 bernama *marketingadmin*

```
[profile marketingadmin]  
role_arn = arn:aws:iam::123456789012:role/marketingadminrole  
credential_source = Ec2InstanceMetadata  
region = us-west-2
```

## Kredensi sumber dengan proses eksternal

### Warning

Topik ini membahas sumber kredensial dari proses eksternal. Ini bisa menjadi risiko keamanan jika perintah untuk menghasilkan kredensial dapat diakses oleh proses atau pengguna yang tidak disetujui. Kami menyarankan Anda menggunakan alternatif yang didukung dan aman yang disediakan oleh AWS CLI dan AWS untuk mengurangi risiko mengorbankan kredensi Anda. Pastikan Anda mengamankan `config` file dan file dan alat pendukung apa pun untuk mencegah pengungkapan.

Pastikan alat kredensial kustom Anda tidak menulis informasi rahasia apa pun `StdErr` karena SDK dan AWS CLI dapat menangkap dan mencatat informasi tersebut, berpotensi mengeksposnya kepada pengguna yang tidak sah.

Jika Anda memiliki metode untuk menghasilkan atau mencari kredensial yang tidak didukung secara langsung oleh AWS CLI, Anda dapat mengonfigurasi AWS CLI untuk menggunakannya dengan mengonfigurasi `credential_process` pengaturan dalam file `.config`

Misalnya, Anda mungkin menyertakan entri yang mirip dengan yang berikut ini dalam `config` file.

```
[profile developer]
credential_process = /opt/bin/awscreds-custom --username helen
```

## Sintaksis

Untuk membuat string ini dengan cara yang kompatibel dengan sistem operasi apa pun, ikuti aturan ini:

- Jika jalur atau nama file berisi spasi, kelilingi jalur lengkap dan nama file dengan tanda kutip ganda (""). Jalur dan nama file hanya dapat terdiri dari karakter: A-Z a-z 0-9 - \_ . spasi
- Jika nama parameter atau nilai parameter berisi spasi, kelilingi elemen tersebut dengan tanda kutip ganda (""). Kelilingi hanya nama atau nilai, bukan pasangannya.
- Jangan sertakan variabel lingkungan apa pun dalam string. Misalnya, Anda tidak dapat memasukkan `$HOME` atau `%USERPROFILE%`.
- Jangan tentukan folder beranda sebagai `~`. Anda harus menentukan jalur lengkap.

## Contoh untuk Windows

```
credential_process = "C:\Path\To\credentials.cmd" parameterWithoutSpaces "parameter with spaces"
```

## Contoh untuk Linux atau macOS

```
credential_process = "/Users/Dave/path/to/credentials.sh" parameterWithoutSpaces "parameter with spaces"
```

## Output yang diharapkan dari program Credentials

AWS CLI menjalankan perintah seperti yang ditentukan dalam profil dan kemudian membaca data dari `STDOUT`. Perintah yang Anda tentukan harus menghasilkan output JSON `STDOUT` yang cocok dengan sintaks berikut.

```
{  
  "Version": 1,  
  "AccessKeyId": "an AWS access key",  
  "SecretAccessKey": "your AWS secret access key",  
  "SessionToken": "the AWS session token for temporary credentials",  
  "Expiration": "ISO8601 timestamp when the credentials expire"  
}
```

#### Note

Pada tulisan ini, `Version` kuncinya harus diatur ke 1. Ini mungkin meningkat seiring waktu seiring berkembangnya struktur.

`Expiration` kuncinya adalah stempel waktu [berformat ISO8601](#). Jika `Expiration` kunci tidak ada dalam output alat, CLI mengasumsikan bahwa kredensialnya adalah kredensial jangka panjang yang tidak disegarkan. Jika tidak, kredensial dianggap sebagai kredensial sementara dan disegarkan secara otomatis dengan menjalankan kembali perintah sebelum kedaluwarsa.

```
credential_process
```

#### Note

AWS CLI tidak menyimpan kredensial proses eksternal seperti yang dilakukannya kredensial peran asumsi. Jika caching diperlukan, Anda harus menerapkannya dalam proses eksternal.

Proses eksternal dapat mengembalikan kode pengembalian bukan nol untuk menunjukkan bahwa kesalahan terjadi saat mengambil kredensi.

# Gunakan AWS CLI

Bagian ini memberikan informasi tentang penggunaan umum, fitur umum, dan opsi yang tersedia di AWS Command Line Interface (AWS CLI), di luar apa yang tertulis di [the section called “Titik akhir”](#) bagian Konfigurasi. Instruksi mencakup cara menulis perintah, struktur dasar, pemformatan, pemfilteran, dan menemukan konten bantuan atau dokumentasi untuk sebuah perintah.

Untuk contoh Layanan AWS spesifik, lihat [Contoh kode](#) atau [referensi AWS CLI versi 2](#).

## Note

Secara default, AWS CLI mengirim permintaan ke AWS layanan dengan menggunakan HTTPS pada port TCP 443. AWS CLI Agar berhasil menggunakan, Anda harus dapat membuat koneksi keluar pada port TCP 443.

Topik dalam panduan ini

- [Dapatkan bantuan dengan AWS CLI](#)
- [Struktur perintah di AWS CLI](#)
- [Tentukan nilai parameter untuk AWS CLI](#)
- [AWS CLIMinta Anda meminta perintah](#)
- [Kontrol output perintah dari AWS CLI](#)
- [Mengembalikan kode dari AWS CLI](#)
- [Perintah interaktif menggunakan AWS CLI penyihir](#)
- [Buat dan gunakan pintasan AWS CLI perintah yang disebut alias](#)

## Dapatkan bantuan dengan AWS CLI

Topik ini menjelaskan cara mengakses konten bantuan untuk AWS Command Line Interface (AWS CLI).

Topik

- [Perintah AWS CLI bantuan bawaan](#)
- [AWS CLIpanduan referensi](#)

- [Dokumentasi API](#)
- [Memecahkan masalah kesalahan](#)
- [Bantuan tambahan](#)

## Perintah AWS CLI bantuan bawaan

Anda bisa mendapatkan bantuan dengan perintah apa pun saat menggunakan AWS Command Line Interface (AWS CLI). Untuk melakukannya, cukup ketik `help` di akhir nama perintah.

Misalnya, perintah berikut menampilkan bantuan untuk AWS CLI opsi umum dan perintah tingkat atas yang tersedia.

```
$ aws help
```

Perintah berikut menampilkan perintah khusus Amazon Elastic Compute Cloud (Amazon EC2) yang tersedia.

```
$ aws ec2 help
```

Contoh berikut menampilkan bantuan terperinci untuk operasi Amazon EC2 `DescribeInstances`. Bantuan tersebut mencakup deskripsi parameter inputnya, filter yang tersedia, dan apa yang disertakan sebagai output. Ini juga mencakup contoh yang menunjukkan cara mengetik variasi umum dari perintah.

```
$ aws ec2 describe-instances help
```

Bantuan untuk setiap perintah dibagi menjadi enam bagian:

### Nama

Nama perintah.

```
NAME
    describe-instances -
```

### Deskripsi

Deskripsi operasi API yang dipanggil perintah.

**DESCRIPTION**

Describes one or more of your instances.

If you specify one or more instance IDs, Amazon EC2 returns information for those instances. If you do not specify instance IDs, Amazon EC2 returns information for all relevant instances. If you specify an instance ID that is not valid, an error is returned. If you specify an instance that you do not own, it is not included in the returned results.

...

**Sinopsis**

Sintaks dasar untuk menggunakan perintah dan opsinya. Jika opsi ditampilkan dalam tanda kurung siku, itu opsional, memiliki nilai default, atau memiliki opsi alternatif yang dapat Anda gunakan.

**SYNOPSIS**

```
describe-instances
[--dry-run | --no-dry-run]
[--instance-ids <value>]
[--filters <value>]
[--cli-input-json <value>]
[--starting-token <value>]
[--page-size <value>]
[--max-items <value>]
[--generate-cli-skeleton]
```

Misalnya, `describe-instances` memiliki perilaku default yang menjelaskan semua instance di akun saat ini dan AWS Wilayah. Anda dapat secara opsional menentukan daftar `instance-ids` untuk menggambarkan satu atau beberapa contoh; `dry-run` adalah bendera Boolean opsional yang tidak mengambil nilai. Untuk menggunakan bendera Boolean, tentukan nilai yang ditampilkan, dalam kasus ini `--dry-run` atau `--no-dry-run`. Demikian juga, `--generate-cli-skeleton` tidak mengambil nilai. Jika ada kondisi pada penggunaan opsi, mereka dijelaskan di **OPTIONS** bagian, atau ditampilkan dalam contoh.

**Opsi**

Deskripsi masing-masing opsi yang ditunjukkan dalam sinopsis.

**OPTIONS**

```
--dry-run | --no-dry-run (boolean)
  Checks whether you have the required permissions for the action,
  without actually making the request, and provides an error response.
  If you have the required permissions, the error response is DryRun-
  Operation . Otherwise, it is UnauthorizedOperation .

--instance-ids (list)
  One or more instance IDs.

  Default: Describes all your instances.

...
```

## Contoh

Contoh yang menunjukkan penggunaan perintah dan opsinya. Jika tidak ada contoh yang tersedia untuk perintah atau kasus penggunaan yang Anda butuhkan, mintalah satu menggunakan tautan umpan balik di halaman ini, atau dalam referensi AWS CLI perintah di halaman bantuan untuk perintah tersebut.

### EXAMPLES

#### **To describe an Amazon EC2 instance**

Command:

```
aws ec2 describe-instances --instance-ids i-5203422c
```

#### **To describe all instances with the instance type m1.small**

Command:

```
aws ec2 describe-instances --filters "Name=instance-type,Values=m1.small"
```

#### **To describe all instances with an Owner tag**

Command:

```
aws ec2 describe-instances --filters "Name=tag-key,Values=Owner"
```

...

## Output

Deskripsi masing-masing bidang dan tipe data yang disertakan dalam respons dari AWS.



Untuk `describe-instances`, outputnya adalah daftar objek reservasi, yang masing-masing berisi beberapa bidang dan objek yang berisi informasi tentang instance yang terkait dengannya. Informasi ini berasal dari [dokumentasi API untuk tipe data reservasi](#) yang digunakan oleh Amazon EC2.

**OUTPUT**

```
Reservations -> (list)
  One or more reservations.

  (structure)
    Describes a reservation.

    ReservationId -> (string)
      The ID of the reservation.

    OwnerId -> (string)
      The ID of the AWS account that owns the reservation.

    RequesterId -> (string)
      The ID of the requester that launched the instances on your
      behalf (for example, AWS Management Console or Auto Scaling).

  Groups -> (list)
    One or more security groups.

    (structure)
      Describes a security group.

      GroupName -> (string)
        The name of the security group.

      GroupId -> (string)
        The ID of the security group.

  Instances -> (list)
    One or more instances.

    (structure)
      Describes an instance.

      InstanceId -> (string)
        The ID of the instance.
```

```
ImageId -> (string)
    The ID of the AMI used to launch the instance.

State -> (structure)
    The current state of the instance.

Code -> (integer)
    The low byte represents the state. The high byte
    is an opaque internal value and should be ignored.
```

...

Ketika AWS CLI merender output ke JSON, itu menjadi array objek reservasi, mirip dengan contoh berikut.

```
{
  "Reservations": [
    {
      "OwnerId": "012345678901",
      "ReservationId": "r-4c58f8a0",
      "Groups": [],
      "RequesterId": "012345678901",
      "Instances": [
        {
          "Monitoring": {
            "State": "disabled"
          },
          "PublicDnsName": "ec2-52-74-16-12.us-
west-2.compute.amazonaws.com",
          "State": {
            "Code": 16,
            "Name": "running"
          },
        },
      ],
    },
  ],
}
```

...

Setiap objek reservasi berisi bidang yang menjelaskan reservasi dan array objek instance, masing-masing dengan bidangnya sendiri (misalnya, `PublicDnsName`) dan objek (misalnya, `State`) yang mendeskripsikannya.

### Pengguna Windows

Anda dapat menyalurkan (|) output dari perintah bantuan ke more perintah untuk melihat file bantuan satu halaman pada satu waktu. Tekan spasi atau PgDn untuk melihat lebih banyak dokumen, dan q untuk berhenti.

```
C:\> aws ec2 describe-instances help | more
```

## AWS CLI panduan referensi

File bantuan berisi tautan yang tidak dapat dilihat atau dinavigasi dari baris perintah. Anda dapat melihat dan berinteraksi dengan tautan ini dengan menggunakan [referensi AWS CLI versi 2](#). Referensi juga berisi konten bantuan untuk semua AWS CLI perintah. Deskripsi disajikan untuk navigasi dan tampilan yang mudah di layar seluler, tablet, atau desktop.

## Dokumentasi API

Semua perintah dalam AWS CLI sesuai dengan permintaan yang dibuat ke API publik AWS layanan. Setiap layanan dengan API publik memiliki referensi API yang dapat ditemukan di halaman beranda layanan di [situs web AWS Dokumentasi](#). Konten untuk referensi API bervariasi berdasarkan bagaimana API dibangun dan protokol mana yang digunakan. Biasanya, referensi API berisi informasi terperinci tentang operasi yang didukung oleh API, data yang dikirim ke dan dari layanan, dan kondisi kesalahan apa pun yang dapat dilaporkan oleh layanan.

### Bagian Dokumentasi API

- **Tindakan** — Informasi terperinci tentang setiap operasi dan parameternya (termasuk kendala pada panjang atau konten, dan nilai default). Ini mencantumkan kesalahan yang dapat terjadi untuk operasi ini. Setiap operasi sesuai dengan subperintah di AWS CLI
- **Tipe Data** — Informasi terperinci tentang struktur yang mungkin diperlukan perintah sebagai parameter, atau dikembalikan sebagai respons terhadap permintaan.
- **Parameter Umum** — Informasi terperinci tentang parameter yang dibagikan oleh semua tindakan untuk layanan.
- **Kesalahan Umum** — Informasi terperinci tentang kesalahan yang dapat dikembalikan oleh salah satu operasi layanan.

Nama dan ketersediaan setiap bagian dapat bervariasi, tergantung pada layanan.

### CLI khusus layanan

Beberapa layanan memiliki CLI terpisah yang berasal dari sebelum satu AWS CLI dibuat untuk bekerja dengan semua layanan. CLI khusus layanan ini memiliki dokumentasi terpisah yang ditautkan dari halaman dokumentasi layanan. Dokumentasi untuk CLI khusus layanan tidak berlaku untuk AWS CLI

## Memecahkan masalah kesalahan

Untuk bantuan mendiagnosis dan memperbaiki AWS CLI kesalahan, lihat [Memecahkan masalah kesalahan](#).

## Bantuan tambahan

Untuk bantuan tambahan AWS CLI terkait masalah Anda, kunjungi [AWS CLI komunitas](#) di GitHub.

## Struktur perintah di AWS CLI

Topik ini mencakup bagaimana AWS Command Line Interface (AWS CLI) perintah terstruktur, dan bagaimana menggunakan perintah tunggal.

Topik

- [Struktur perintah](#)
- [Tunggu perintah](#)

## Struktur perintah

AWS CLIMenggunakan struktur multipart pada baris perintah yang harus ditentukan dalam urutan ini:

1. Panggilan dasar ke aws program.
2. Perintah tingkat atas, yang biasanya sesuai dengan AWS layanan yang didukung oleh AWS CLI
3. Subperintah yang menentukan operasi mana yang harus dilakukan.

4. AWS CLI Opsi umum atau parameter yang diperlukan oleh operasi. Anda dapat menentukan ini dalam urutan apa pun selama mereka mengikuti tiga bagian pertama. Jika parameter eksklusif ditentukan beberapa kali, hanya nilai terakhir yang berlaku.

```
$ aws <command> <subcommand> [options and parameters]
```

Parameter dapat mengambil berbagai jenis nilai input, seperti angka, string, daftar, peta, dan struktur JSON. Apa yang didukung tergantung pada perintah dan subperintah yang Anda tentukan.

Contoh

Amazon S3

Contoh berikut mencantumkan semua bucket Amazon S3 Anda.

```
$ aws s3 ls
2018-12-11 17:08:50 my-bucket
2018-12-14 14:55:44 my-bucket2
```

Untuk informasi selengkapnya tentang perintah Amazon S3, lihat di Referensi AWS CLI Perintah.

AWS CloudFormation

Contoh perintah berikut mengubah nama tumpukan cloudformation menjadi *my-change-set*

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set-name my-change-set
```

Untuk informasi selengkapnya tentang AWS CloudFormation perintah, lihat di Referensi AWS CLI Perintah.

## Tunggu perintah

Beberapa AWS layanan memiliki `wait` perintah yang tersedia. Setiap perintah yang menggunakan `aws wait` biasanya menunggu sampai perintah selesai sebelum pindah ke langkah berikutnya. Ini sangat berguna untuk perintah multibagian atau skrip, karena Anda dapat menggunakan perintah `wait` untuk mencegah pindah ke langkah berikutnya jika perintah `wait` gagal.

AWS CLI Menggunakan struktur multipart pada baris perintah untuk `wait` perintah yang harus ditentukan dalam urutan ini:

1. Panggilan dasar ke aws program.
2. Perintah tingkat atas, yang biasanya sesuai dengan AWS layanan yang didukung oleh. AWS CLI
3. waitPerintah.
4. Subperintah yang menentukan operasi mana yang harus dilakukan.
5. Opsi atau parameter CLI umum yang diperlukan oleh operasi. Anda dapat menentukan ini dalam urutan apa pun selama mereka mengikuti tiga bagian pertama. Jika parameter eksklusif ditentukan beberapa kali, hanya nilai terakhir yang berlaku.

```
$ aws <command> wait <subcommand> [options and parameters]
```

Parameter dapat mengambil berbagai jenis nilai input, seperti angka, string, daftar, peta, dan struktur JSON. Apa yang didukung tergantung pada perintah dan subperintah yang Anda tentukan.

#### Note

Tidak semua layanan AWS mendukung wait perintah. Lihat [versi 2 panduan referensi](#) untuk melihat apakah layanan Anda mendukung wait perintah.

#### Contoh

##### AWS CloudFormation

Contoh perintah berikut berhenti dan dilanjutkan hanya setelah dapat mengonfirmasi bahwa *my-change-set* perubahan yang disetel di *tumpukan tumpukan saya siap dijalankan*.

```
$ aws cloudformation wait change-set-create-complete --stack-name my-stack --change-set-name my-change-set
```

Untuk informasi selengkapnya tentang AWS CloudFormation wait perintah, lihat di Referensi AWS CLI Perintah.

##### AWS CodeDeploy

Contoh perintah berikut berhenti sampai penerapan *d-A1B2C3111 berhasil diselesaikan*.

```
$ aws deploy wait deployment-successful --deployment-id d-A1B2C3111
```

Untuk informasi selengkapnya tentang AWS CodeDeploy wait perintah, lihat di Referensi AWS CLI Perintah.

## Tentukan nilai parameter untuk AWS CLI

Banyak parameter yang digunakan dalam AWS Command Line Interface (AWS CLI) adalah string sederhana atau nilai numerik, seperti nama kunci-pasangan `my-key-pair` dalam contoh berikut.

```
$ aws ec2 create-key-pair --key-name my-key-pair
```

Pemformatan antar terminal dapat bervariasi. Misalnya, sebagian besar terminal peka huruf besar/kecil tetapi Powershell tidak peka huruf besar/kecil. Ini berarti dua contoh perintah berikut akan menghasilkan hasil yang berbeda untuk terminal peka huruf besar/kecil saat mereka melihat `MyFile*.txt` dan `myfile*.txt` sebagai parameter yang berbeda.

Namun, PowerShell akan memproses permintaan ini sama seperti yang terlihat `MyFile*.txt` dan `myfile*.txt` sebagai parameter yang sama.

```
$ aws s3 cp . s3://my-bucket/path --include "MyFile*.txt"  
$ aws s3 cp . s3://my-bucket/path --include "myfile*.txt"
```

Untuk informasi selengkapnya tentang PowerShell case insensitiv, lihat [About case-sensitivity](#) dalam dokumentasi. PowerShell

Terkadang Anda perlu menggunakan tanda kutip atau literal di sekitar string yang menyertakan karakter khusus atau spasi. Aturan seputar pemformatan ini juga dapat bervariasi antar terminal. Untuk informasi selengkapnya tentang menggunakan tanda kutip di sekitar parameter kompleks, lihat [Tanda kutip dengan string di AWS CLI](#).

Topik parameter

- [Jenis AWS CLI parameter umum](#)
- [Tanda kutip dengan string di AWS CLI](#)
- [Muat AWS CLI parameter dari file](#)
- [AWS CLI kerangka dan file masukan](#)
- [Gunakan sintaks singkatan dengan AWS CLI](#)

## Jenis AWS CLI parameter umum

Bagian ini menjelaskan beberapa jenis parameter umum dan format khas yang diperlukan.

Jika Anda mengalami masalah dalam memformat parameter untuk perintah tertentu, periksa bantuan dengan memasukkan **help** setelah nama perintah. Bantuan untuk setiap subperintah mencakup nama dan deskripsi opsi. Jenis parameter opsi tercantum dalam tanda kurung. Untuk informasi selengkapnya tentang melihat bantuan, lihat [the section called “Dapatkan Bantuan”](#).

Jenis parameter meliputi:

- [String](#)
- [Timestamp](#)
- [Daftar](#)
- [Boolean](#)
- [Bulat](#)
- [Biner/gumpalan \(objek besar biner\) dan gumpalan streaming](#)
- [Map](#)
- [Dokumen](#)

### String

Parameter string dapat berisi karakter alfanumerik, simbol, dan spasi putih dari set karakter [ASCII](#). String yang berisi spasi putih harus dikelilingi oleh tanda kutip. Kami menyarankan Anda untuk tidak menggunakan simbol atau spasi putih selain karakter spasi standar dan untuk mematuhi [aturan kutipan](#) terminal Anda untuk mencegah hasil yang tidak terduga.

Beberapa parameter string dapat menerima data biner dari file. Lihat [File biner](#) sebagai contoh.

### Timestamp

Stempel waktu diformat sesuai dengan [standar ISO](#) 8601. Ini sering disebut sebagai parameter "DateTime" atau Date "".

```
$ aws ec2 describe-spot-price-history --start-time 2014-10-13T19:00:00Z
```

Format yang dapat diterima meliputi:



- `YYYY-MM-DD T hh:mm:ss.sss TZD (UTC)`, misalnya, `2014-10-01T20:30:00.000 Z`
- `YYYY-MM-DD T hh:mm:ss.sss TZD (dengan offset)`, misalnya, `2014-10-01T12:30:00.000-08:00`
- `YYYY-MM-DD`, misalnya, `2014-10-01`
- Waktu Unix dalam hitungan detik, misalnya, `1412195400`. Ini kadang-kadang disebut sebagai [waktu Unix Epoch](#) dan mewakili jumlah detik sejak tengah malam, 1 Januari 1970 UTC.

Secara default, AWS CLI versi 2 menerjemahkan semua DateTime nilai respons ke format ISO 8601.

Anda dapat mengatur format stempel waktu dengan menggunakan pengaturan [cli\\_timestamp\\_format](#) file.

## Daftar

Satu atau lebih string dipisahkan oleh spasi. Jika salah satu item string berisi spasi, Anda harus meletakkan tanda kutip di sekitar item itu. Amati [aturan kutipan](#) terminal Anda untuk mencegah hasil yang tidak terduga.

```
$ aws ec2 describe-spot-price-history --instance-types m1.xlarge m1.medium
```

## Boolean

Bendera biner yang mengaktifkan atau menonaktifkan opsi. Misalnya, `ec2 describe-spot-price-history` memiliki `--dry-run` parameter Boolean yang, ketika ditentukan, memvalidasi kueri dengan layanan tanpa benar-benar menjalankan kueri.

```
$ aws ec2 describe-spot-price-history --dry-run
```

Output menunjukkan apakah perintah itu terbentuk dengan baik. Perintah ini juga menyertakan `--no-dry-run` versi parameter yang dapat Anda gunakan untuk secara eksplisit menunjukkan bahwa perintah harus dijalankan secara normal. Menyertakannya tidak perlu karena ini adalah perilaku default.

## Bulat

Nomor bulat yang tidak ditandatangani.

```
$ aws ec2 describe-spot-price-history --max-items 5
```

## Biner/gumpalan (objek besar biner) dan gumpalan streaming

Dalam AWS CLI, Anda dapat melewati nilai biner sebagai string langsung pada baris perintah. Ada dua jenis gumpalan:

- [Gumpalan](#)
- [Gumpalan streaming](#)

### blob

Untuk meneruskan nilai ke parameter dengan tipe blob, Anda harus menentukan jalur ke file lokal yang berisi data biner menggunakan `fileb://` awalan. File yang direferensikan menggunakan `fileb://` awalan selalu diperlakukan sebagai biner mentah yang tidak dikodekan. Jalur yang ditentukan ditafsirkan sebagai relatif terhadap direktori kerja saat ini. Misalnya, `--plaintext` parameter untuk `aws kms encrypt` adalah gumpalan.

```
$ aws kms encrypt \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --plaintext fileb://ExamplePlaintextFile \  
  --output text \  
  --query CiphertextBlob | base64 \  
  --decode > ExampleEncryptedFile
```

### Note

Untuk kompatibilitas mundur, Anda dapat menggunakan `file://` awalan. Ada dua format yang digunakan berdasarkan pengaturan file [cli\\_binary\\_format](#) atau opsi baris [--cli-binary-format](#) perintah:

- Default untuk AWS CLI versi 2. Jika nilai pengaturan adalah `base64`, file yang direferensikan menggunakan `file://` awalan diperlakukan sebagai teks yang dikodekan base64.
- Default untuk AWS CLI versi 1. Jika nilai pengaturan adalah `raw-in-base64-out`, file yang direferensikan menggunakan `file://` awalan dibaca sebagai teks dan kemudian AWS CLI upaya untuk menyandikannya ke biner.

Untuk informasi selengkapnya, lihat opsi pengaturan file [cli\\_binary\\_format](#) atau baris `--cli-binary-format` perintah.

## Gumpalan streaming

Gumpalan streaming seperti `aws cloudsearchdomain upload-documents` tidak menggunakan awalan. Sebagai gantinya, parameter blob streaming diformat menggunakan jalur file langsung. Contoh berikut menggunakan path file langsung `document-batch.json` untuk `aws cloudsearchdomain upload-documents` perintah:

```
$ aws cloudsearchdomain upload-documents \  
  --endpoint-url https://doc-my-domain.us-west-1.cloudsearch.amazonaws.com \  
  --content-type application/json \  
  --documents document-batch.json
```

## Map

[Satu set pasangan kunci-nilai yang ditentukan dalam JSON atau dengan menggunakan sintaks singkatan CLI](#). Contoh JSON berikut membaca item dari tabel Amazon DynamoDB bernama `my-table` dengan parameter peta, `--key` Parameter menentukan kunci utama bernama `id` dengan nilai angka 1 dalam struktur JSON bersarang.

Untuk penggunaan JSON yang lebih maju di baris perintah, pertimbangkan untuk menggunakan prosesor JSON baris perintah, seperti `jq`, untuk membuat string JSON. Untuk informasi lebih lanjut tentang `jq`, lihat [repositori jq](#) di GitHub

```
$ aws dynamodb get-item --table-name my-table --key '{"id": {"N": "1"}}'  
  
{  
  "Item": {  
    "name": {  
      "S": "John"  
    },  
    "id": {  
      "N": "1"  
    }  
  }  
}
```

```
}
```

## Dokumen

### Note

[Sintaks singkatan](#) tidak kompatibel dengan jenis dokumen.

Jenis dokumen digunakan untuk mengirim data tanpa perlu menanamkan JSON di dalam string. Jenis dokumen memungkinkan layanan untuk menyediakan skema arbitrer bagi Anda untuk menggunakan tipe data yang lebih fleksibel.

Ini memungkinkan untuk mengirim data JSON tanpa perlu melarikan diri dari nilai. Misalnya, alih-alih menggunakan input JSON yang lolos berikut:

```
{"document": "{\"key\":true}"}
```

Anda dapat menggunakan jenis dokumen berikut:

```
{"document": {"key": true}}
```

Nilai yang valid untuk jenis dokumen

Karena sifat fleksibel dari jenis dokumen, ada beberapa jenis nilai yang valid. Nilai yang valid mencakup hal berikut:

### String

```
--option "value"
```

### Number

```
--option 123  
--option 123.456
```

### Boolean

```
--option true
```

## Nol

```
--option null
```

## Susunan

```
--option '["value1", "value2", "value3"]'  
--option '["value", 1, true, null, ["key1", 2.34], {"key2": "value2"}]'
```

## Objek

```
--option '{"key": "value"}'  
--option '{"key1": "value1", "key2": 123, "key3": true, "key4": null, "key5":  
["value3", "value4"], "key6": {"value5": "value6"}'}
```

## Tanda kutip dengan string di AWS CLI

Ada terutama dua cara tanda kutip tunggal dan ganda digunakan dalam AWS CLI.

- [Menggunakan tanda kutip di sekitar string yang berisi spasi putih](#)
- [Menggunakan tanda kutip di dalam string](#)

### Menggunakan tanda kutip di sekitar string yang berisi spasi putih

Nama parameter dan nilainya dipisahkan oleh spasi pada baris perintah. Jika nilai string berisi spasi tertanam, maka Anda harus mengelilingi seluruh string dengan tanda kutip untuk AWS CLI mencegah salah menafsirkan ruang sebagai pembagi antara nilai dan nama parameter berikutnya. Jenis tanda kutip yang Anda gunakan tergantung pada sistem operasi yang Anda gunakan. AWS CLI

#### Linux and macOS

Gunakan tanda kutip tunggal ' '

```
$ aws ec2 create-key-pair --key-name 'my key pair'
```

Untuk informasi selengkapnya tentang penggunaan tanda kutip, lihat dokumentasi pengguna untuk shell pilihan Anda.

## PowerShell

### Kutipan tunggal (disarankan)

Tanda kutip tunggal ' ' disebut *verbatim string*. String diteruskan ke perintah persis seperti yang Anda ketikkan, yang berarti PowerShell variabel tidak akan melewatinya.

```
PS C:\> aws ec2 create-key-pair --key-name 'my key pair'
```

### Kutipan ganda

Tanda kutip ganda " " disebut *expandable string*. Variabel dapat dilewatkan dalam string yang dapat diperluas.

```
PS C:\> aws ec2 create-key-pair --key-name "my key pair"
```

Untuk informasi selengkapnya tentang menggunakan tanda kutip, lihat [Tentang Aturan Mengutip](#) di Microsoft PowerShell Docs.

## Windows command prompt

Gunakan tanda " " kutip ganda.

```
C:\> aws ec2 create-key-pair --key-name "my key pair"
```

Secara opsional, Anda dapat memisahkan nama parameter dari nilai dengan tanda sama dengan = bukan spasi. Ini biasanya diperlukan hanya jika nilai parameter dimulai dengan tanda hubung.

```
$ aws ec2 delete-key-pair --key-name=-mykey
```

## Menggunakan tanda kutip di dalam string

String mungkin berisi tanda kutip, dan shell Anda mungkin memerlukan kutipan keluar agar berfungsi dengan baik. Salah satu jenis nilai parameter umum adalah string JSON. Ini kompleks karena mencakup spasi dan tanda kutip ganda " " di sekitar setiap nama elemen dan nilai dalam struktur JSON. Cara Anda memasukkan parameter yang berformat JSON pada baris perintah berbeda-beda tergantung dari sistem operasi Anda.

Untuk penggunaan JSON yang lebih maju di baris perintah, pertimbangkan untuk menggunakan prosesor JSON baris perintah, seperti `jq`, untuk membuat string JSON. Untuk informasi lebih lanjut tentang `jq`, lihat [repositori jq](#) di GitHub

## Linux and macOS

Untuk Linux dan macOS untuk menafsirkan string secara harfiah menggunakan tanda kutip tunggal `' '` untuk melampirkan struktur data JSON, seperti pada contoh berikut. Anda tidak perlu melepaskan tanda kutip ganda yang disematkan dalam string JSON, karena mereka diperlakukan secara harfiah. Karena JSON terlampir dalam tanda kutip tunggal, setiap tanda kutip tunggal dalam string harus diloloskan, ini biasanya dilakukan dengan menggunakan garis miring terbalik sebelum kutipan tunggal. `\'`

```
$ aws ec2 run-instances \
  --image-id ami-12345678 \
  --block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":"standard"}]'
```

Untuk informasi selengkapnya tentang penggunaan tanda kutip, lihat dokumentasi pengguna untuk shell pilihan Anda.

## PowerShell

Gunakan tanda kutip tunggal `' '` atau tanda kutip ganda `" "`

Kutipan tunggal (disarankan)

Tanda kutip tunggal `' '` disebut *verbatim string*. String diteruskan ke perintah persis seperti yang Anda ketikkan, yang berarti PowerShell variabel tidak akan melewatinya.

Karena struktur data JSON menyertakan tanda kutip ganda, kami menyarankan tanda kutip tunggal `' '` untuk melampirkannya. Jika Anda menggunakan tanda kutip tunggal, Anda tidak perlu melepaskan tanda kutip ganda yang disematkan dalam string JSON. Namun, Anda harus keluar dari setiap tanda kutip tunggal dengan backtick ``` dalam struktur JSON.

```
PS C:\> aws ec2 run-instances `
  --image-id ami-12345678 `
  --block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":"standard"}]'
```

## Kutipan ganda

Tanda kutip ganda " " disebut `expandable string`. Variabel dapat dilewatkan dalam string yang dapat diperluas.

Jika Anda menggunakan tanda kutip ganda, Anda tidak perlu melepaskan tanda kutip tunggal yang disematkan dalam string JSON. Namun, Anda harus keluar dari setiap tanda kutip ganda dengan backtick ` dalam struktur JSON, seperti contoh berikut.

```
PS C:\> aws ec2 run-instances `
  --image-id ami-12345678 `
  --block-device-mappings "[{"DeviceName `": `"/dev/sdb `", "Ebs `":
{"VolumeSize `":20, "DeleteOnTermination `":false, "VolumeType `": "standard `"}]"`
```

Untuk informasi selengkapnya tentang menggunakan tanda kutip, lihat [Tentang Aturan Mengutip](#) di Microsoft PowerShell Docs.

#### Warning

Sebelum PowerShell mengirim perintah ke AWS CLI, itu menentukan apakah perintah Anda ditafsirkan menggunakan aturan tipikal PowerShell atau `CommandLineToArgvW` kutipan. Saat PowerShell proses menggunakan `CommandLineToArgvW`, Anda harus melarikan diri dari karakter dengan garis miring terbalik \.

Untuk informasi lebih lanjut tentang `CommandLineToArgvW` in PowerShell, lihat [Ada apa dengan perlakuan aneh tanda kutip dan garis miring terbalik oleh CommandLineToArgv W](#) di Microsoft DevBlogs, [Semua orang mengutip argumen baris perintah dengan cara yang salah di](#) Blog Microsoft Docs, dan fungsi [CommandLineToArgvW](#) di Microsoft Docs.

#### Kutipan tunggal

Tanda kutip tunggal ' ' disebut `verbatim string`. String diteruskan ke perintah persis seperti yang Anda ketikkan, yang berarti PowerShell variabel tidak akan melewatinya. Melarikan diri dari karakter dengan garis miring terbalik \.

```
PS C:\> aws ec2 run-instances `
  --image-id ami-12345678 `
  --block-device-mappings '[{"DeviceName `": `"/dev/sdb `", "Ebs `":
{"VolumeSize `":20, "DeleteOnTermination `":false, "VolumeType `": "standard `"}]'`
```

#### Kutipan ganda

Tanda kutip ganda " " disebut `expandable string`. Variabel dapat dilewatkan dalam `expandable string`. Untuk string kutip ganda Anda harus melarikan diri dua kali menggunakan ` \ untuk setiap kutipan alih-alih hanya menggunakan backtick. Backtick



lolos dari garis miring terbalik, dan kemudian garis miring terbalik digunakan sebagai karakter pelarian untuk proses tersebut. `CommandLineToArgvW`

```
PS C:\> aws ec2 run-instances `
  --image-id ami-12345678 `
  --block-device-mappings "[{"DeviceName `": `"/dev/sdb `", `": `":
{"VolumeSize `":20, `": `":false, `": `":
`"standard `"}]"
```

Gumpalan (disarankan)

Untuk melewati aturan PowerShell kutipan untuk input data JSON, gunakan Blobs untuk meneruskan data JSON Anda langsung ke file. AWS CLI Untuk informasi lebih lanjut tentang Blobs, lihat [blob](#).

## Windows command prompt

Prompt perintah Windows memerlukan tanda kutip ganda " " untuk melampirkan struktur data JSON. Juga, untuk mencegah prosesor perintah salah menafsirkan tanda kutip ganda yang disematkan di JSON, Anda juga harus melarikan diri (mendahului dengan \ karakter garis miring terbalik) setiap tanda kutip ganda " dalam struktur data JSON itu sendiri, seperti pada contoh berikut.

```
C:\> aws ec2 run-instances ^
  --image-id ami-12345678 ^
  --block-device-mappings [{"DeviceName`":`"/dev/sdb`,`":`":
{"VolumeSize`":20,`":`":false,`":`":`"standard`"}]"
```

Hanya tanda kutip ganda terluar yang tidak lolos.

## Muat AWS CLI parameter dari file

Beberapa parameter mengharapkan nama file sebagai argumen, dari mana AWS CLI memuat data. Parameter lain memungkinkan Anda untuk menentukan nilai parameter sebagai teks yang diketik pada baris perintah atau membaca dari file. Apakah file diperlukan atau opsional, Anda harus menyandikan file dengan benar sehingga AWS CLI dapat memahaminya. Pengkodean file harus sesuai dengan lokal default sistem pembacaan. Anda dapat menentukan ini dengan menggunakan metode `Pythonlocale.getpreferredencoding()`.

**Note**

Secara default, Windows PowerShell mengeluarkan teks sebagai UTF-16, yang bertentangan dengan pengkodean UTF-8 yang digunakan oleh file JSON dan banyak sistem Linux. Kami menyarankan Anda menggunakan `-Encoding ascii` dengan PowerShell `Out-File` perintah Anda untuk memastikan AWS CLI dapat membaca file yang dihasilkan.

## Topik

- [Cara memuat parameter dari file](#)
- [File biner](#)

## Cara memuat parameter dari file

Terkadang lebih mudah untuk memuat nilai parameter dari file alih-alih mencoba mengetikkan semuanya sebagai nilai parameter baris perintah, seperti ketika parameternya adalah string JSON yang kompleks. Untuk menentukan file yang berisi nilai, tentukan URL file dalam format berikut.

```
file://complete/path/to/file
```

- Dua karakter garis miring '/' pertama adalah bagian dari spesifikasi. Jika jalur yang diperlukan dimulai dengan '/', hasilnya adalah tiga karakter garis miring: `file:///folder/file`.
- URL menyediakan path ke file yang berisi konten parameter yang sebenarnya.
- Saat menggunakan file dengan spasi atau karakter khusus, ikuti [aturan mengutip dan melarikan diri untuk terminal](#) Anda.

Jalur file dalam contoh berikut ditafsirkan relatif terhadap direktori kerja saat ini.

## Linux or macOS

```
// Read from a file in the current directory
$ aws ec2 describe-instances --filters file://filter.json

// Read from a file in /tmp
$ aws ec2 describe-instances --filters file:///tmp/filter.json

// Read from a file with a filename with whitespaces
```

```
$ aws ec2 describe-instances --filters 'file://filter_content.json'
```

## Windows command prompt

```
// Read from a file in C:\temp
C:\> aws ec2 describe-instances --filters file://C:\temp\filter.json

// Read from a file with a filename with whitespaces
C:\> aws ec2 describe-instances --filters "file://C:\temp\filter_content.json"
```

Opsi `file://` awalan mendukung ekspansi bergaya Unix, termasuk `~/`, `"`, `"`, dan `"/`. `./` `../`  
 Pada Windows, ekspresi `~/` meluas ke direktori pengguna Anda, disimpan dalam variabel `%USERPROFILE%` lingkungan. Misalnya, pada Windows 10 Anda biasanya memiliki direktori pengguna di bawah `C:\Users\UserName\`.

Anda masih harus melarikan diri dari dokumen JSON yang disematkan sebagai nilai dokumen JSON lain.

```
$ aws sqs create-queue --queue-name my-queue --attributes file://attributes.json
```

## attributes.json

```
{
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-west-2:0123456789012:deadletter\", \"maxReceiveCount\":\"5\"}"
}
```

## File biner

Untuk perintah yang mengambil data biner sebagai parameter, tentukan bahwa data tersebut adalah konten biner dengan menggunakan `fileb://` awalan. Perintah yang menerima data biner meliputi:

- **aws ec2 run-instances**: `--user-dataparameter`.
- **aws s3api put-object**: `--sse-customer-keyparameter`.
- **aws kms decrypt**: `--ciphertext-blobparameter`.

Contoh berikut menghasilkan kunci AES 256-bit biner menggunakan alat baris perintah Linux, dan kemudian menyediakannya ke Amazon S3 untuk mengenkripsi sisi server file yang diunggah.

```
$ dd if=/dev/urandom bs=1 count=32 > sse.key
32+0 records in
32+0 records out
32 bytes (32 B) copied, 0.000164441 s, 195 kB/s
$ aws s3api put-object \
  --bucket my-bucket \
  --key test.txt \
  --body test.txt \
  --sse-customer-key fileb://sse.key \
  --sse-customer-algorithm AES256
{
  "SSECustomerKeyMD5": "iVg8oWa8sy714+FjtesrJg==",
  "SSECustomerAlgorithm": "AES256",
  "ETag": "\"a6118e84b76cf98bf04bbe14b6045c6c\""
}
```

Untuk contoh lain yang mereferensikan file yang berisi parameter berformat JSON, lihat. [Lampirkan kebijakan terkelola IAM ke pengguna](#)

## AWS CLIKerangka dan file masukan

Sebagian besar AWS CLI perintah menerima semua input parameter dari file. Template ini dapat dibuat menggunakan `generate-cli-skeleton` opsi.

### Topik

- [Tentang AWS CLI kerangka dan file input](#)
- [Menghasilkan kerangka perintah](#)

## Tentang AWS CLI kerangka dan file input

Sebagian besar perintah AWS Command Line Interface (AWS CLI) mendukung kemampuan untuk menerima semua input parameter dari file menggunakan `--cli-input-yaml` parameter `--cli-input-json` dan `s`.

Perintah yang sama membantu menyediakan `--generate-cli-skeleton` parameter untuk menghasilkan file dalam format JSON atau YAMAL dengan semua parameter yang dapat Anda edit dan isi. Kemudian Anda dapat menjalankan perintah dengan `--cli-input-yaml` parameter yang relevan `--cli-input-json` dan arahkan ke file yang diisi.

### ⚠ Important

Beberapa AWS CLI perintah tidak memetakan langsung ke operasi AWS API individual, seperti [perintah](#). Perintah semacam itu tidak mendukung `--cli-input-yaml` parameter `--generate-cli-skeleton` atau `--cli-input-json` dan yang dijelaskan dalam topik ini. Jika Anda tidak tahu apakah perintah tertentu mendukung parameter ini, jalankan perintah berikut, ganti nama *layanan* dan *perintah* dengan yang Anda minati.

```
$ aws service command help
```

Outputnya mencakup Synopsis bagian yang menunjukkan parameter yang didukung oleh perintah yang ditentukan.

```
$ aws iam list-users help
...
SYNOPSIS
    list-users
    ...
    [--cli-input-json | --cli-input-yaml]
    ...
    [--generate-cli-skeleton <value>]
...
```

`--generate-cli-skeleton` Parameter menyebabkan perintah tidak berjalan, melainkan untuk menghasilkan dan menampilkan template parameter yang dapat Anda sesuaikan dan gunakan sebagai input pada perintah selanjutnya. Template yang dihasilkan mencakup semua parameter yang didukung perintah.

`--generate-cli-skeleton` Parameter menerima salah satu nilai berikut:

- `input`— Template yang dihasilkan mencakup semua parameter input yang diformat sebagai JSON. Ini adalah nilai default.
- `yaml-input`— Template yang dihasilkan mencakup semua parameter input yang diformat sebagai YAMAL.
- `output`— Template yang dihasilkan mencakup semua parameter output yang diformat sebagai JSON. Saat ini Anda tidak dapat meminta parameter keluaran sebagai YAMAL.

Karena pada dasarnya AWS CLI adalah “pembungkus” di sekitar API layanan, file kerangka mengharapkan Anda untuk mereferensikan semua parameter dengan nama parameter API yang mendasarinya. Ini mungkin berbeda dari nama AWS CLI parameter. Misalnya, AWS CLI parameter bernama `user-name` mungkin memetakan ke parameter API AWS layanan bernama `UserName` (perhatikan kapitalisasi yang diubah dan tanda hubung yang hilang). Kami menyarankan Anda menggunakan `--generate-cli-skeleton` opsi untuk menghasilkan template dengan nama parameter “benar” untuk menghindari kesalahan. Anda juga dapat mereferensikan Panduan Referensi API agar layanan dapat melihat nama parameter yang diharapkan. Anda dapat menghapus parameter apa pun dari templat yang tidak diperlukan dan yang tidak ingin Anda berikan nilainya.

Misalnya, jika Anda menjalankan perintah berikut, itu akan menghasilkan template parameter untuk perintah Amazon Elastic Compute Cloud (Amazon EC2). `run-instances`

## JSON

Contoh berikut menunjukkan bagaimana untuk menghasilkan template diformat dalam JSON dengan menggunakan nilai default (`input`) untuk parameter. `--generate-cli-skeleton`

```
$ aws ec2 run-instances --generate-cli-skeleton
```

```
{
  "DryRun": true,
  "ImageId": "",
  "MinCount": 0,
  "MaxCount": 0,
  "KeyName": "",
  "SecurityGroups": [
    ""
  ],
  "SecurityGroupIds": [
    ""
  ],
  "UserData": "",
  "InstanceType": "",
  "Placement": {
    "AvailabilityZone": "",
    "GroupName": "",
    "Tenancy": ""
  },
  "KernelId": "",
  "RamdiskId": "",
```

```
"BlockDeviceMappings": [
  {
    "VirtualName": "",
    "DeviceName": "",
    "Ebs": {
      "SnapshotId": "",
      "VolumeSize": 0,
      "DeleteOnTermination": true,
      "VolumeType": "",
      "Iops": 0,
      "Encrypted": true
    },
    "NoDevice": ""
  }
],
"Monitoring": {
  "Enabled": true
},
"SubnetId": "",
"DisableApiTermination": true,
"InstanceInitiatedShutdownBehavior": "",
"PrivateIpAddress": "",
"ClientToken": "",
"AdditionalInfo": "",
"NetworkInterfaces": [
  {
    "NetworkInterfaceId": "",
    "DeviceIndex": 0,
    "SubnetId": "",
    "Description": "",
    "PrivateIpAddress": "",
    "Groups": [
      ""
    ],
    "DeleteOnTermination": true,
    "PrivateIpAddresses": [
      {
        "PrivateIpAddress": "",
        "Primary": true
      }
    ],
    "SecondaryPrivateIpAddressCount": 0,
    "AssociatePublicIpAddress": true
  }
]
```

```

    ],
    "IamInstanceProfile": {
        "Arn": "",
        "Name": ""
    },
    "EbsOptimized": true
}

```

## YAML

Contoh berikut menunjukkan cara menghasilkan template yang diformat dalam YAMAL dengan menggunakan nilai `yaml-input` untuk parameter. `--generate-cli-skeleton`

```
$ aws ec2 run-instances --generate-cli-skeleton yaml-input
```

```

BlockDeviceMappings: # The block device mapping entries.
- DeviceName: '' # The device name (for example, /dev/sdh or xvdh).
  VirtualName: '' # The virtual device name (ephemeralN).
  Ebs: # Parameters used to automatically set up Amazon EBS volumes when the
instance is launched.
    DeleteOnTermination: true # Indicates whether the EBS volume is deleted on
instance termination.
    Iops: 0 # The number of I/O operations per second (IOPS) that the volume
supports.
    SnapshotId: '' # The ID of the snapshot.
    VolumeSize: 0 # The size of the volume, in GiB.
    VolumeType: st1 # The volume type. Valid values are: standard, io1, gp2, sc1,
st1.
    Encrypted: true # Indicates whether the encryption state of an EBS volume is
changed while being restored from a backing snapshot.
    KmsKeyId: '' # Identifier (key ID, key alias, ID ARN, or alias ARN) for a
customer managed KMS key under which the EBS volume is encrypted.
    NoDevice: '' # Suppresses the specified device included in the block device
mapping of the AMI.
ImageId: '' # The ID of the AMI.
InstanceType: c4.4xlarge # The instance type. Valid values are: t1.micro, t2.nano,
t2.micro, t2.small, t2.medium, t2.large, t2.xlarge, t2.2xlarge, t3.nano, t3.micro,
t3.small, t3.medium, t3.large, t3.xlarge, t3.2xlarge, t3a.nano, t3a.micro,
t3a.small, t3a.medium, t3a.large, t3a.xlarge, t3a.2xlarge, m1.small, m1.medium,
m1.large, m1.xlarge, m3.medium, m3.large, m3.xlarge, m3.2xlarge, m4.large,
m4.xlarge, m4.2xlarge, m4.4xlarge, m4.10xlarge, m4.16xlarge, m2.xlarge, m2.2xlarge,
m2.4xlarge, cr1.8xlarge, r3.large, r3.xlarge, r3.2xlarge, r3.4xlarge, r3.8xlarge,
r4.large, r4.xlarge, r4.2xlarge, r4.4xlarge, r4.8xlarge, r4.16xlarge, r5.large,

```



r5.xlarge, r5.2xlarge, r5.4xlarge, r5.8xlarge, r5.12xlarge, r5.16xlarge, r5.24xlarge, r5.metal, r5a.large, r5a.xlarge, r5a.2xlarge, r5a.4xlarge, r5a.8xlarge, r5a.12xlarge, r5a.16xlarge, r5a.24xlarge, r5d.large, r5d.xlarge, r5d.2xlarge, r5d.4xlarge, r5d.8xlarge, r5d.12xlarge, r5d.16xlarge, r5d.24xlarge, r5d.metal, r5ad.large, r5ad.xlarge, r5ad.2xlarge, r5ad.4xlarge, r5ad.8xlarge, r5ad.12xlarge, r5ad.16xlarge, r5ad.24xlarge, x1.16xlarge, x1.32xlarge, x1e.xlarge, x1e.2xlarge, x1e.4xlarge, x1e.8xlarge, x1e.16xlarge, x1e.32xlarge, i2.xlarge, i2.2xlarge, i2.4xlarge, i2.8xlarge, i3.large, i3.xlarge, i3.2xlarge, i3.4xlarge, i3.8xlarge, i3.16xlarge, i3.metal, i3en.large, i3en.xlarge, i3en.2xlarge, i3en.3xlarge, i3en.6xlarge, i3en.12xlarge, i3en.24xlarge, i3en.metal, hi1.4xlarge, hs1.8xlarge, c1.medium, c1.xlarge, c3.large, c3.xlarge, c3.2xlarge, c3.4xlarge, c3.8xlarge, c4.large, c4.xlarge, c4.2xlarge, c4.4xlarge, c4.8xlarge, c5.large, c5.xlarge, c5.2xlarge, c5.4xlarge, c5.9xlarge, c5.12xlarge, c5.18xlarge, c5.24xlarge, c5.metal, c5d.large, c5d.xlarge, c5d.2xlarge, c5d.4xlarge, c5d.9xlarge, c5d.18xlarge, c5n.large, c5n.xlarge, c5n.2xlarge, c5n.4xlarge, c5n.9xlarge, c5n.18xlarge, cc1.4xlarge, cc2.8xlarge, g2.2xlarge, g2.8xlarge, g3.4xlarge, g3.8xlarge, g3.16xlarge, g3s.xlarge, g4dn.xlarge, g4dn.2xlarge, g4dn.4xlarge, g4dn.8xlarge, g4dn.12xlarge, g4dn.16xlarge, cg1.4xlarge, p2.xlarge, p2.8xlarge, p2.16xlarge, p3.2xlarge, p3.8xlarge, p3.16xlarge, p3dn.24xlarge, d2.xlarge, d2.2xlarge, d2.4xlarge, d2.8xlarge, f1.2xlarge, f1.4xlarge, f1.16xlarge, m5.large, m5.xlarge, m5.2xlarge, m5.4xlarge, m5.8xlarge, m5.12xlarge, m5.16xlarge, m5.24xlarge, m5.metal, m5a.large, m5a.xlarge, m5a.2xlarge, m5a.4xlarge, m5a.8xlarge, m5a.12xlarge, m5a.16xlarge, m5a.24xlarge, m5d.large, m5d.xlarge, m5d.2xlarge, m5d.4xlarge, m5d.8xlarge, m5d.12xlarge, m5d.16xlarge, m5d.24xlarge, m5d.metal, m5ad.large, m5ad.xlarge, m5ad.2xlarge, m5ad.4xlarge, m5ad.8xlarge, m5ad.12xlarge, m5ad.16xlarge, m5ad.24xlarge, h1.2xlarge, h1.4xlarge, h1.8xlarge, h1.16xlarge, z1d.large, z1d.xlarge, z1d.2xlarge, z1d.3xlarge, z1d.6xlarge, z1d.12xlarge, z1d.metal, u-6tb1.metal, u-9tb1.metal, u-12tb1.metal, u-18tb1.metal, u-24tb1.metal, a1.medium, a1.large, a1.xlarge, a1.2xlarge, a1.4xlarge, a1.metal, m5dn.large, m5dn.xlarge, m5dn.2xlarge, m5dn.4xlarge, m5dn.8xlarge, m5dn.12xlarge, m5dn.16xlarge, m5dn.24xlarge, m5n.large, m5n.xlarge, m5n.2xlarge, m5n.4xlarge, m5n.8xlarge, m5n.12xlarge, m5n.16xlarge, m5n.24xlarge, r5dn.large, r5dn.xlarge, r5dn.2xlarge, r5dn.4xlarge, r5dn.8xlarge, r5dn.12xlarge, r5dn.16xlarge, r5dn.24xlarge, r5n.large, r5n.xlarge, r5n.2xlarge, r5n.4xlarge, r5n.8xlarge, r5n.12xlarge, r5n.16xlarge, r5n.24xlarge.

Ipv6AddressCount: 0 # [EC2-VPC] The number of IPv6 addresses to associate with the primary network interface.

Ipv6Addresses: # [EC2-VPC] The IPv6 addresses from the range of the subnet to associate with the primary network interface.

- Ipv6Address: '' # The IPv6 address.

KernelId: '' # The ID of the kernel.

KeyName: '' # The name of the key pair.

MaxCount: 0 # [REQUIRED] The maximum number of instances to launch.

MinCount: 0 # [REQUIRED] The minimum number of instances to launch.

```
Monitoring: # Specifies whether detailed monitoring is enabled for the instance.
  Enabled: true # [REQUIRED] Indicates whether detailed monitoring is enabled.
Placement: # The placement for the instance.
  AvailabilityZone: '' # The Availability Zone of the instance.
  Affinity: '' # The affinity setting for the instance on the Dedicated Host.
  GroupName: '' # The name of the placement group the instance is in.
  PartitionNumber: 0 # The number of the partition the instance is in.
  HostId: '' # The ID of the Dedicated Host on which the instance resides.
  Tenancy: dedicated # The tenancy of the instance (if the instance is running in a
VPC). Valid values are: default, dedicated, host.
  SpreadDomain: '' # Reserved for future use.
RamdiskId: '' # The ID of the RAM disk to select.
SecurityGroupIds: # The IDs of the security groups.
- ''
SecurityGroups: # [default VPC] The names of the security groups.
- ''
SubnetId: '' # [EC2-VPC] The ID of the subnet to launch the instance into.
UserData: '' # The user data to make available to the instance.
AdditionalInfo: '' # Reserved.
ClientToken: '' # Unique, case-sensitive identifier you provide to ensure the
  idempotency of the request.
DisableApiTermination: true # If you set this parameter to true, you can't terminate
  the instance using the Amazon EC2 console, CLI, or API; otherwise, you can.
DryRun: true # Checks whether you have the required permissions for the action,
  without actually making the request, and provides an error response.
EbsOptimized: true # Indicates whether the instance is optimized for Amazon EBS I/O.
IamInstanceProfile: # The IAM instance profile.
  Arn: '' # The Amazon Resource Name (ARN) of the instance profile.
  Name: '' # The name of the instance profile.
InstanceInitiatedShutdownBehavior: stop # Indicates whether an instance stops or
  terminates when you initiate shutdown from the instance (using the operating system
  command for system shutdown). Valid values are: stop, terminate.
NetworkInterfaces: # The network interfaces to associate with the instance.
- AssociatePublicIpAddress: true # Indicates whether to assign a public IPv4
  address to an instance you launch in a VPC.
  DeleteOnTermination: true # If set to true, the interface is deleted when the
  instance is terminated.
  Description: '' # The description of the network interface.
  DeviceIndex: 0 # The position of the network interface in the attachment order.
  Groups: # The IDs of the security groups for the network interface.
  - ''
  Ipv6AddressCount: 0 # A number of IPv6 addresses to assign to the network
  interface.
  Ipv6Addresses: # One or more IPv6 addresses to assign to the network interface.
```

```
- Ipv6Address: '' # The IPv6 address.
NetworkInterfaceId: '' # The ID of the network interface.
PrivateIpAddress: '' # The private IPv4 address of the network interface.
PrivateIpAddresses: # One or more private IPv4 addresses to assign to the network
interface.
- Primary: true # Indicates whether the private IPv4 address is the primary
private IPv4 address.
  PrivateIpAddress: '' # The private IPv4 addresses.
  SecondaryPrivateIpAddressCount: 0 # The number of secondary private IPv4
addresses.
  SubnetId: '' # The ID of the subnet associated with the network interface.
  InterfaceType: '' # The type of network interface.
PrivateIpAddress: '' # [EC2-VPC] The primary IPv4 address.
ElasticGpuSpecification: # An elastic GPU to associate with the instance.
- Type: '' # [REQUIRED] The type of Elastic Graphics accelerator.
ElasticInferenceAccelerators: # An elastic inference accelerator to associate with
the instance.
- Type: '' # [REQUIRED] The type of elastic inference accelerator.
TagSpecifications: # The tags to apply to the resources during launch.
- ResourceType: network-interface # The type of resource to tag. Valid values
are: client-vpn-endpoint, customer-gateway, dedicated-host, dhcp-options, elastic-
ip, fleet, fpga-image, host-reservation, image, instance, internet-gateway,
launch-template, natgateway, network-acl, network-interface, reserved-instances,
route-table, security-group, snapshot, spot-instances-request, subnet, traffic-
mirror-filter, traffic-mirror-session, traffic-mirror-target, transit-gateway,
transit-gateway-attachment, transit-gateway-route-table, volume, vpc, vpc-peering-
connection, vpn-connection, vpn-gateway.
  Tags: # The tags to apply to the resource.
  - Key: '' # The key of the tag.
    Value: '' # The value of the tag.
LaunchTemplate: # The launch template to use to launch the instances.
  LaunchTemplateId: '' # The ID of the launch template.
  LaunchTemplateName: '' # The name of the launch template.
  Version: '' # The version number of the launch template.
InstanceMarketOptions: # The market (purchasing) option for the instances.
  MarketType: spot # The market type. Valid values are: spot.
  SpotOptions: # The options for Spot Instances.
  MaxPrice: '' # The maximum hourly price you're willing to pay for the Spot
Instances.
  SpotInstanceType: one-time # The Spot Instance request type. Valid values are:
one-time, persistent.
  BlockDurationMinutes: 0 # The required duration for the Spot Instances (also
known as Spot blocks), in minutes.
  ValidUntil: 1970-01-01 00:00:00 # The end date of the request.
```

```

InstanceInterruptionBehavior: terminate # The behavior when a Spot Instance is
interrupted. Valid values are: hibernate, stop, terminate.
CreditSpecification: # The credit option for CPU usage of the T2 or T3 instance.
  CpuCredits: '' # [REQUIRED] The credit option for CPU usage of a T2 or T3
instance.
CpuOptions: # The CPU options for the instance.
  CoreCount: 0 # The number of CPU cores for the instance.
  ThreadsPerCore: 0 # The number of threads per CPU core.
CapacityReservationSpecification: # Information about the Capacity Reservation
targeting option.
  CapacityReservationPreference: none # Indicates the instance's Capacity
Reservation preferences. Valid values are: open, none.
  CapacityReservationTarget: # Information about the target Capacity Reservation.
  CapacityReservationId: '' # The ID of the Capacity Reservation.
HibernationOptions: # Indicates whether an instance is enabled for hibernation.
  Configured: true # If you set this parameter to true, your instance is enabled
for hibernation.
LicenseSpecifications: # The license configurations.
- LicenseConfigurationArn: '' # The Amazon Resource Name (ARN) of the license
configuration.

```

## Menghasilkan kerangka perintah

Untuk menghasilkan dan menggunakan file kerangka parameter

1. Jalankan perintah dengan `--generate-cli-skeleton` parameter untuk menghasilkan JSON atau YAMAL dan arahkan output ke file untuk menyimpannya.

JSON

```
$ aws ec2 run-instances --generate-cli-skeleton input > ec2runinst.json
```

YAML

```
$ aws ec2 run-instances --generate-cli-skeleton yaml-input > ec2runinst.yaml
```

2. Buka file kerangka parameter di editor teks Anda dan hapus salah satu parameter yang tidak Anda butuhkan. Misalnya, Anda dapat menghapus template ke bawah ini. Pastikan file tersebut masih valid JSON atau YAML setelah Anda menghapus elemen yang tidak Anda butuhkan.

## JSON

```
{
  "DryRun": true,
  "ImageId": "",
  "KeyName": "",
  "SecurityGroups": [
    ""
  ],
  "InstanceType": "",
  "Monitoring": {
    "Enabled": true
  }
}
```

## YAML

```
DryRun: true
ImageId: ''
KeyName: ''
SecurityGroups:
- ''
InstanceType:
Monitoring:
  Enabled: true
```

Dalam contoh ini, kita membiarkan `DryRun` parameter disetel `true` untuk menggunakan fitur dry run Amazon EC2. Fitur ini memungkinkan Anda menguji perintah dengan aman tanpa benar-benar membuat atau memodifikasi sumber daya apa pun.

3. Isi nilai yang tersisa dengan nilai yang sesuai untuk skenario Anda. Dalam contoh ini, kami menyediakan jenis instans, nama kunci, grup keamanan, dan pengenalan Amazon Machine Image (AMI) untuk digunakan. Contoh ini mengasumsikan AWS Region default. AMI `ami-dfc39aef` adalah gambar Linux Amazon 64-bit yang dihosting di `us-west-2` Wilayah. Jika Anda menggunakan Wilayah yang berbeda, Anda harus [menemukan ID AMI yang benar untuk digunakan](#).

## JSON

```
{
```

```
"DryRun": true,
"ImageId": "ami-dfc39aef",
"KeyName": "mykey",
"SecurityGroups": [
  "my-sg"
],
"InstanceType": "t2.micro",
"Monitoring": {
  "Enabled": true
}
}
```

## YAML

```
DryRun: true
ImageId: 'ami-dfc39aef'
KeyName: 'mykey'
SecurityGroups:
- 'my-sg'
InstanceType: 't2.micro'
Monitoring:
  Enabled: true
```

4. Jalankan perintah dengan parameter lengkap dengan meneruskan file template yang sudah selesai ke `cli-input-yaml` parameter `--cli-input-json` or `--` dengan menggunakan `file://` awalan. AWS CLIMenafsirkan jalur menjadi relatif terhadap direktori kerja Anda saat ini, jadi dalam contoh berikut yang hanya menampilkan nama file tanpa jalur, ia mencari file langsung di direktori kerja saat ini.

## JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json
```

```
A client error (DryRunOperation) occurred when calling the RunInstances
operation: Request would have succeeded, but DryRun flag is set.
```

## YAML

```
$ aws ec2 run-instances --cli-input-yaml file://ec2runinst.yaml
```

```
A client error (DryRunOperation) occurred when calling the RunInstances operation: Request would have succeeded, but DryRun flag is set.
```

Kesalahan dry run menunjukkan bahwa JSON atau YAMAL terbentuk dengan benar dan nilai parameternya valid. Jika masalah lain dilaporkan dalam output, perbaiki dan ulangi langkah sebelumnya hingga pesan Request would have succeeded "" ditampilkan.

5. Sekarang Anda dapat mengatur DryRun parameter false untuk menonaktifkan dry run.

## JSON

```
{
  "DryRun": false,
  "ImageId": "ami-dfc39aef",
  "KeyName": "mykey",
  "SecurityGroups": [
    "my-sg"
  ],
  "InstanceType": "t2.micro",
  "Monitoring": {
    "Enabled": true
  }
}
```

## YAML

```
DryRun: false
ImageId: 'ami-dfc39aef'
KeyName: 'mykey'
SecurityGroups:
- 'my-sg'
InstanceType: 't2.micro'
Monitoring:
  Enabled: true
```

6. Jalankan perintah, dan run-instances benar-benar meluncurkan instans Amazon EC2 dan menampilkan detail yang dihasilkan oleh peluncuran yang berhasil. Format output dikontrol oleh --output parameter, terpisah dari format template parameter input Anda.

## JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json --output json
```

```
{
  "OwnerId": "123456789012",
  "ReservationId": "r-d94a2b1",
  "Groups": [],
  "Instances": [
    ...
  ]
}
```

## YAML

```
$ aws ec2 run-instances --cli-input-yaml file://ec2runinst.yaml --output yaml
```

```
OwnerId: '123456789012'
ReservationId: 'r-d94a2b1',
Groups":
- ''
Instances:
...

```

## Gunakan sintaks singkatan dengan AWS CLI

The AWS Command Line Interface (AWS CLI) dapat menerima banyak parameter opsinya dalam format JSON. Namun, bisa membosankan untuk memasukkan daftar atau struktur JSON besar pada baris perintah. Untuk membuat ini lebih mudah, AWS CLI juga mendukung sintaks singkatan yang memungkinkan representasi yang lebih sederhana dari parameter opsi Anda daripada menggunakan format JSON lengkap.

### Topik

- [Parameter struktur](#)
- [Menggunakan sintaks singkatan dengan AWS Command Line Interface](#)



## Parameter struktur

Sintaks singkatan dalam AWS CLI memudahkan pengguna untuk memasukkan parameter yang datar (struktur non-bersarang). Formatnya adalah daftar pasangan kunci-nilai yang dipisahkan koma. Pastikan untuk menggunakan aturan [kutipan](#) dan escaping yang sesuai untuk terminal Anda karena sintaks singkatan adalah string.

### Linux or macOS

```
--option key1=value1,key2=value2,key3=value3
```

### PowerShell

```
--option "key1=value1,key2=value2,key3=value3"
```

Keduanya setara dengan contoh berikut, diformat dalam JSON.

```
--option '{"key1":"value1","key2":"value2","key3":"value3"}'
```

Tidak boleh ada spasi putih di antara setiap pasangan nilai kunci yang dipisahkan koma. Berikut adalah contoh perintah Amazon `update-table` DynamoDB dengan opsi yang ditentukan `--provisioned-throughput` dalam singkatan.

```
$ aws dynamodb update-table \  
  --provisioned-throughput ReadCapacityUnits=15,WriteCapacityUnits=10 \  
  --table-name MyDDBTable
```

Ini setara dengan contoh berikut diformat dalam JSON.

```
$ aws dynamodb update-table \  
  --provisioned-throughput '{"ReadCapacityUnits":15,"WriteCapacityUnits":10}' \  
  --table-name MyDDBTable
```

## Menggunakan sintaks singkatan dengan AWS Command Line Interface

Anda dapat menentukan parameter Input dalam formulir daftar dengan dua cara: JSON atau singkatan. Sintaks AWS CLI singkatan dirancang untuk membuatnya lebih mudah untuk meneruskan daftar dengan angka, string, atau struktur non-bersarang.

Format dasar ditampilkan di sini, di mana nilai-nilai dalam daftar dipisahkan oleh satu spasi.

```
--option value1 value2 value3
```

Ini setara dengan contoh berikut, diformat dalam JSON.

```
--option '[value1,value2,value3]'
```

Seperti disebutkan sebelumnya, Anda dapat menentukan daftar angka, daftar string, atau daftar struktur non-bersarang di singkatan. Berikut ini adalah contoh `stop-instances` perintah untuk Amazon Elastic Compute Cloud (Amazon EC2), di mana parameter input (daftar string) untuk `--instance-ids` opsi ditentukan dalam singkatan.

```
$ aws ec2 stop-instances \  
  --instance-ids i-1486157a i-1286157c i-ec3a7e87
```

Ini setara dengan contoh berikut diformat dalam JSON.

```
$ aws ec2 stop-instances \  
  --instance-ids '["i-1486157a","i-1286157c","i-ec3a7e87"]'
```

Contoh berikut menunjukkan `create-tags` perintah Amazon EC2, yang mengambil daftar struktur non-bersarang untuk opsi. `--tags` `--resources` Pilihan menentukan ID dari contoh untuk tag.

```
$ aws ec2 create-tags \  
  --resources i-1286157c \  
  --tags Key=My1stTag,Value=Value1 Key=My2ndTag,Value=Value2  
Key=My3rdTag,Value=Value3
```

Ini setara dengan contoh berikut, diformat dalam JSON. Parameter JSON ditulis melalui beberapa baris untuk keterbacaan.

```
$ aws ec2 create-tags \  
  --resources i-1286157c \  
  --tags '[  
    {"Key": "My1stTag", "Value": "Value1"},  
    {"Key": "My2ndTag", "Value": "Value2"},  
    {"Key": "My3rdTag", "Value": "Value3"}  
]'
```

# AWS CLIMinta Anda meminta perintah

Anda dapat meminta perintah, parameter, dan sumber daya AWS CLI versi 2 saat menjalankan aws perintah.

Topik

- [Cara kerjanya](#)
- [Fitur prompt otomatis](#)
- [Mode prompt otomatis](#)
- [Konfigurasi prompt otomatis](#)

## Cara kerjanya

Jika diaktifkan, prompt otomatis memungkinkan Anda menggunakan tombol ENTER untuk menyelesaikan perintah yang dimasukkan sebagian. Setelah menekan tombol ENTER, perintah, parameter, dan sumber daya disarankan berdasarkan apa yang terus Anda ketik. Saran mencantumkan nama perintah, parameter, atau sumber daya di sebelah kiri dan deskripsi di sebelah kanan. Untuk memilih dan menggunakan saran, gunakan tombol panah untuk menyorot baris, lalu tekan tombol SPASI. Setelah selesai memasukkan perintah Anda, tekan ENTER untuk menggunakan perintah. Contoh berikut menunjukkan seperti apa daftar yang disarankan dari prompt otomatis.

```
$ aws
> aws a
    accessanalyzer      Access Analyzer
    acm                  AWS Certificate Manager
    acm-pca              AWS Certificate Manager Private Certificate
Authority
    alexaforbusiness    Alexa For Business
    amplify              AWS Amplify
```

## Fitur prompt otomatis

Prompt otomatis berisi fitur-fitur berguna berikut:

### Panel dokumentasi

Menyediakan dokumentasi bantuan untuk perintah saat ini. Untuk membuka dokumentasi, tekan tombol F3.

## Penyelesaian perintah

Menyarankan aws perintah untuk digunakan. Untuk melihat daftar, masukkan sebagian perintah. Contoh berikut adalah mencari layanan yang dimulai dengan huruf itua.

```
$ aws
> aws a
    accessanalyzer           Access Analyzer
    acm                      AWS Certificate Manager
    acm-pca                  AWS Certificate Manager Private Certificate
Authority
    alexaforbusiness        Alexa For Business
    amplify                 AWS Amplify
```

## Penyelesaian parameter

Setelah perintah diketik, prompt otomatis mulai menyarankan parameter. Deskripsi untuk parameter termasuk jenis nilai, dan deskripsi tentang apa parameternya. Parameter yang diperlukan tercantum terlebih dahulu, dan diberi label sesuai kebutuhan. Contoh berikut menunjukkan daftar auto-prompt parameter untuk `aws dynamodb describe-table`.

```
$ aws dynamodb describe-table
> aws dynamodb describe-table
    --table-name (required) [string] The name of the
table to describe.
    --cli-input-json [string] Reads arguments
from the JSON string provided. The JSON string follows the format provide...
    --cli-input-yaml [string] Reads arguments
from the YAML string provided. The YAML string follows the format provide...
    --generate-cli-skeleton [string] Prints a JSON
skeleton to standard output without sending an API request. If provided wit...
```

## Penyelesaian sumber daya

Prompt otomatis membuat panggilan AWS API menggunakan properti sumber daya AWS yang tersedia untuk menyarankan nilai sumber daya. Ini memungkinkan prompt otomatis untuk menyarankan kemungkinan sumber daya yang Anda miliki saat memasukkan parameter. Dalam contoh berikut, prompt otomatis mencantumkan nama tabel Anda saat mengisi `--table-name` parameter untuk `aws dynamodb describe-table` perintah.

```
$ aws dynamodb describe-table
```

```
> aws dynamodb describe-table --table-name
                                     Table1
                                     Table2
                                     Table3
```

## Penyelesaian singkatan

Untuk parameter yang menggunakan sintaks singkatan, prompt otomatis menyarankan nilai untuk digunakan. Dalam contoh berikut, prompt otomatis mencantumkan nilai sintaks singkatan untuk `--placement` parameter dalam perintah. `aws ec2 run-instances`

```
$ aws ec2 run-instances
> aws ec2 run-instances --placement
AvailabilityZone= [string] The Availability Zone of the instance. If not
specified, an Availability Zone wil...
Affinity= [string] The affinity setting for the instance on the
Dedicated Host. This parameter is no...
GroupName= [string] The name of the placement group the instance is in.
PartitionNumber= [integer] The number of the partition the instance is in.
Valid only if the placement grou...
```

## Penyelesaian berkas

Saat mengisi parameter dalam aws perintah, pelengkapan otomatis menyarankan nama file lokal setelah menggunakan awalan atau. `file:// fileb://` Dalam contoh berikut, prompt otomatis menyarankan file lokal setelah masuk `--item file://` untuk `aws ec2 run-instances` perintah.

```
$ aws ec2 run-instances
> aws ec2 run-instances --item file://
                               item1.txt
                               file1.json
                               file2.json
```

## Penyelesaian wilayah

Saat menggunakan parameter global `--region`, prompt otomatis mencantumkan kemungkinan Wilayah untuk dipilih. Dalam contoh berikut, prompt otomatis menyarankan Wilayah dalam urutan abjad setelah masuk `--region` untuk perintah. `aws dynamodb list-tables`

```
$ aws dynamodb list-tables
```

```
> aws dynamodb list-tables --region
                                af-south-1
                                ap-east-1
                                ap-northeast-1
                                ap-northeast-2
```

## Penyelesaian profil

Saat menggunakan parameter global `--profile`, prompt otomatis mencantumkan profil Anda. Dalam contoh berikut, prompt otomatis menyarankan profil Anda setelah masuk `--profile` untuk `aws dynamodb list-tables` perintah.

```
$ aws dynamodb list-tables
> aws dynamodb list-tables --profile
                                profile1
                                profile2
                                profile3
```

## Pencarian fuzzy

Lengkapi perintah dan nilai yang berisi serangkaian karakter tertentu. Dalam contoh berikut, prompt otomatis menyarankan Wilayah yang berisi eu setelah masuk `--region eu` untuk `aws dynamodb list-tables` perintah.

```
$ aws dynamodb list-tables
> aws dynamodb list-tables --region west
                                eu-west-1
                                eu-west-2
                                eu-west-3
                                us-west-1
```

## Riwayat

Untuk melihat dan menjalankan perintah yang sebelumnya digunakan dalam mode prompt otomatis, tekan CTRL+R. Riwayat mencantumkan perintah sebelumnya yang dapat Anda pilih dengan menggunakan tombol panah. Dalam contoh berikut, riwayat mode auto-prompt ditampilkan.

```
$ aws
> aws
                                dynamodb list-tables
```

```
s3 ls
```

## Mode prompt otomatis

Auto-prompt untuk AWS CLI versi 2 memiliki 2 mode yang dapat dikonfigurasi:

- Mode penuh: Menggunakan prompt otomatis setiap kali Anda mencoba menjalankan aws perintah, apakah Anda memanggilnya secara manual menggunakan `--cli-auto-prompt` parameter atau mengaktifkannya secara permanen. Ini termasuk menekan ENTER setelah perintah lengkap atau perintah tidak lengkap.
- Mode paral: Menggunakan prompt otomatis jika perintah tidak lengkap atau tidak dapat dijalankan karena kesalahan validasi sisi klien. Mode ini sangat berguna jika Anda memiliki skrip, runbook, atau Anda hanya ingin diminta secara otomatis untuk perintah yang tidak Anda kenal daripada diminta pada setiap perintah.

## Konfigurasi prompt otomatis

Untuk mengkonfigurasi prompt otomatis Anda dapat menggunakan metode berikut dalam urutan prioritas:

- Opsi baris perintah mengaktifkan atau menonaktifkan prompt otomatis untuk satu perintah. Gunakan `--cli-auto-prompt` untuk memanggil auto-prompt dan `--no-cli-auto-prompt` untuk menonaktifkan auto-prompt.
- Variabel lingkungan menggunakan `aws_cli_auto_prompt` variabel.
- File konfigurasi bersama menggunakan `cli_auto_prompt` pengaturan.

## Kontrol output perintah dari AWS CLI

Bagian ini menjelaskan berbagai cara untuk mengontrol output dari AWS Command Line Interface (AWS CLI). Menyesuaikan AWS CLI output di terminal Anda dapat meningkatkan keterbacaan, merampingkan otomatisasi skrip, dan memberikan navigasi yang lebih mudah melalui kumpulan data yang lebih besar.

AWS CLI Mendukung beberapa [format output](#), termasuk [json](#), [text](#), [yaml](#), dan [table](#). Beberapa layanan memiliki [pagination](#) sisi server untuk data mereka dan AWS CLI menyediakan fitur sisi kliennya sendiri untuk opsi pagination tambahan.

Terakhir, AWS CLI memiliki [penyaringan sisi server dan sisi klien yang dapat Anda gunakan secara individual atau bersama-sama untuk memfilter](#) output Anda. AWS CLI

## Topik

- [Output sensitif](#)
- [Opsi keluaran sisi server vs sisi klien](#)
- [Mengatur format AWS CLI output](#)
- [Gunakan AWS CLI opsi pagination](#)
- [Filter AWS CLI keluaran](#)

## Output sensitif

Beberapa operasi AWS CLI mungkin mengembalikan informasi yang dapat dianggap sensitif, termasuk informasi dari variabel lingkungan. Paparan informasi ini mungkin mewakili risiko keamanan dalam skenario tertentu; misalnya, informasi dapat dimasukkan dalam log integrasi berkelanjutan dan penyebaran berkelanjutan (CI/CD). Oleh karena itu penting bahwa Anda meninjau ketika Anda memasukkan output tersebut sebagai bagian dari log Anda, dan menekan output ketika tidak diperlukan.

Untuk informasi tambahan tentang melindungi data sensitif, lihat [the section called “Perlindungan Data”](#).

Pertimbangkan praktik terbaik berikut:

- Pertimbangkan untuk mengambil rahasia Anda secara terprogram dari toko rahasia, seperti AWS Secrets Manager
- Tinjau konten log build Anda untuk memastikan bahwa log tersebut tidak berisi informasi sensitif. Pertimbangkan pendekatan seperti perpipaan ke `/dev/null` atau menangkap output sebagai bash atau PowerShell variabel untuk menekan output perintah.

Berikut ini adalah contoh bash untuk mengarahkan output, tetapi bukan kesalahan, ke: `/dev/null`

```
$ aws s3 ls > /dev/null
```

Untuk spesifik tentang menekan output untuk terminal Anda, lihat dokumentasi pengguna terminal yang Anda gunakan.



- Pertimbangkan akses log Anda dan cakupan akses yang tepat untuk kasus penggunaan Anda.

## Opsi keluaran sisi server vs sisi klien

Ini AWS CLI memiliki penyaringan sisi [server dan sisi klien yang dapat Anda gunakan secara individual atau bersama-sama untuk memfilter](#) output Anda. AWS CLI Pemfilteran sisi server diproses terlebih dahulu dan mengembalikan output Anda untuk pemfilteran sisi klien. Pemfilteran sisi server didukung oleh API layanan. Pemfilteran sisi klien didukung oleh AWS CLI klien menggunakan parameter. `--query`

Opsi keluaran sisi server adalah fitur yang didukung langsung oleh API. Layanan AWS Setiap data yang disaring atau paged out tidak dikirim ke klien, yang dapat mempercepat waktu respons HTTP dan meningkatkan bandwidth untuk kumpulan data yang lebih besar.

Opsi keluaran sisi klien adalah fitur yang dibuat oleh. AWS CLI Semua data dikirim ke klien, kemudian AWS CLI filter atau halaman konten ditampilkan. Operasi sisi klien tidak menghemat kecepatan atau bandwidth untuk kumpulan data yang lebih besar.

Ketika opsi sisi server dan sisi klien digunakan bersama-sama, operasi sisi server diselesaikan terlebih dahulu dan kemudian dikirim ke klien untuk operasi sisi klien. Ini menggunakan potensi penghematan kecepatan dan bandwidth dari opsi sisi server, sambil menggunakan AWS CLI fitur tambahan untuk mendapatkan output yang Anda inginkan.

## Mengatur format AWS CLI output

Topik ini menjelaskan format keluaran yang berbeda untuk AWS Command Line Interface (AWS CLI). AWS CLIdukung format output berikut:

- **[json](#)**— Output diformat sebagai string [JSON](#).
- **[yaml](#)**— Output diformat sebagai string [YAML](#).
- **[yaml-stream](#)**— Output dialirkan dan diformat sebagai string [YAML](#). Streaming memungkinkan penanganan tipe data besar yang lebih cepat.
- **[text](#)**- Output diformat sebagai beberapa baris nilai string yang dipisahkan tab. Ini dapat berguna untuk meneruskan output ke prosesor teks, seperti `grep`, `sed`, atau `awk`.
- **[table](#)**— Output diformat sebagai tabel menggunakan karakter `+`/`|`- untuk membentuk batas sel. Ini biasanya menyajikan informasi dalam format “ramah manusia” yang jauh lebih mudah dibaca daripada yang lain, tetapi tidak berguna secara terprogram.

## Cara memilih format output

Seperti yang dijelaskan dalam topik [konfigurasi](#), Anda dapat menentukan format output dalam tiga cara:

- Menggunakan **output** opsi dalam profil bernama dalam **config** file - Contoh berikut menetapkan format output default ke `text`.

```
[default]
output=text
```

- Menggunakan variabel **AWS\_DEFAULT\_OUTPUT** lingkungan - Output berikut menetapkan format `table` untuk perintah dalam sesi baris perintah ini sampai variabel diubah atau sesi berakhir. Menggunakan variabel lingkungan ini mengesampingkan nilai apa pun yang ditetapkan dalam file. `config`

```
$ export AWS_DEFAULT_OUTPUT="table"
```

- Menggunakan **--output** opsi pada baris perintah - Contoh berikut menetapkan output dari hanya satu perintah ini `json`. Menggunakan opsi ini pada perintah mengesampingkan variabel lingkungan yang saat ini ditetapkan atau nilai dalam file. `config`

```
$ aws swf list-domains --registration-status REGISTERED --output json
```

### Important

Jenis output yang Anda tentukan mengubah cara `--query` opsi beroperasi:

- Jika Anda menentukan `--output text`, output diberi paginasi sebelum `--query` filter diterapkan, dan AWS CLI menjalankan kueri sekali pada setiap halaman output. Karena ini, kueri menyertakan elemen pencocokan pertama pada setiap halaman yang dapat menghasilkan output tambahan yang tidak terduga. Untuk memfilter output tambahan, Anda dapat menggunakan alat baris perintah lainnya seperti `head` atau `tail`.
- Jika Anda menentukan `--output json`, `--output yaml`, atau `--output yaml-stream` output sepenuhnya diproses sebagai struktur asli tunggal sebelum `--query` filter diterapkan. AWS CLI menjalankan kueri hanya sekali terhadap seluruh struktur, menghasilkan hasil yang difilter yang kemudian dikeluarkan.

## Format keluaran JSON

[JSON](#) adalah format output default dari file. AWS CLI Sebagian besar bahasa pemrograman dapat dengan mudah memecahkan kode string JSON menggunakan fungsi bawaan atau dengan perpustakaan yang tersedia untuk umum. Anda dapat menggabungkan output JSON dengan [opsi --query](#) dengan cara yang ampuh untuk memfilter dan memformat output berformat AWS CLI JSON.

Untuk pemfilteran lebih lanjut yang mungkin tidak dapat Anda lakukan `--query`, Anda dapat mempertimbangkan `jq`, prosesor JSON baris perintah. Anda dapat mengunduhnya dan menemukan tutorial resminya di <http://stedolan.github.io/jq/>.

Berikut ini adalah contoh dari output JSON.

```
$ aws iam list-users --output json
```

```
{
  "Users": [
    {
      "Path": "/",
      "UserName": "Admin",
      "UserId": "AIDA111111111111EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/Admin",
      "CreateDate": "2014-10-16T16:03:09+00:00",
      "PasswordLastUsed": "2016-06-03T18:37:29+00:00"
    },
    {
      "Path": "/backup/",
      "UserName": "backup-user",
      "UserId": "AIDA222222222222EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/backup/backup-user",
      "CreateDate": "2019-09-17T19:30:40+00:00"
    },
    {
      "Path": "/",
      "UserName": "cli-user",
      "UserId": "AIDA333333333333EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/cli-user",
      "CreateDate": "2019-09-17T19:11:39+00:00"
    }
  ]
}
```

## Format keluaran YAMG

[YAMAL](#) adalah pilihan yang baik untuk menangani output secara terprogram dengan layanan dan alat yang memancarkan atau menggunakan string berformat YAMAL, seperti dengan dukungannya untuk template yang diformat YAMAL. [AWS CloudFormation](#)

Untuk pemfilteran lebih lanjut yang mungkin tidak dapat Anda lakukan `--query`, Anda dapat mempertimbangkanyq, prosesor YAMAL baris perintah. Anda dapat mengunduh yq di [repositori yq](#) di GitHub

Berikut ini adalah contoh output YAMG.

```
$ aws iam list-users --output yaml
```

Users:

```
- Arn: arn:aws:iam::123456789012:user/Admin
  CreateDate: '2014-10-16T16:03:09+00:00'
  PasswordLastUsed: '2016-06-03T18:37:29+00:00'
  Path: /
  UserId: AIDA111111111111EXAMPLE
  UserName: Admin
- Arn: arn:aws:iam::123456789012:user/backup/backup-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /backup/
  UserId: AIDA222222222222EXAMPLE
  UserName: arq-45EFD6D1-CE56-459B-B39F-F9C1F78FBE19
- Arn: arn:aws:iam::123456789012:user/cli-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /
  UserId: AIDA333333333333EXAMPLE
  UserName: cli-user
```

## Format keluaran aliran YAMG

`yaml-streamFormat` ini memanfaatkan format [YAMAL](#) sambil memberikan tampilan kumpulan data besar yang lebih responsif/lebih cepat dengan mengalirkan data kepada Anda. Anda dapat mulai melihat dan menggunakan data YAMG sebelum seluruh unduhan kueri.

Untuk pemfilteran lebih lanjut yang mungkin tidak dapat Anda lakukan `--query`, Anda dapat mempertimbangkanyq, prosesor YAMAL baris perintah. Anda dapat mengunduh yq di [repositori yq](#) di GitHub

Berikut ini adalah contoh `yaml-stream` output.

```
$ aws iam list-users --output yaml-stream
```

```
- IsTruncated: false
Users:
- Arn: arn:aws:iam::123456789012:user/Admin
  CreateDate: '2014-10-16T16:03:09+00:00'
  PasswordLastUsed: '2016-06-03T18:37:29+00:00'
  Path: /
  UserId: AIDA111111111111EXAMPLE
  UserName: Admin
- Arn: arn:aws:iam::123456789012:user/backup/backup-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /backup/
  UserId: AIDA222222222222EXAMPLE
  UserName: arq-45EFD6D1-CE56-459B-B39F-F9C1F78FBE19
- Arn: arn:aws:iam::123456789012:user/cli-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /
  UserId: AIDA333333333333EXAMPLE
  UserName: cli-user
```

Berikut ini adalah contoh `yaml-stream` output dalam hubungannya dengan menggunakan `--page-size` parameter untuk paginasi konten YAML yang dialirkan.

```
$ aws iam list-users --output yaml-stream --page-size 2
```

```
- IsTruncated: true
Marker: ab1234cdef5ghi67jk8lmo9p/
q012rs3t445uv6789w0x1y2z/345a6b78c9d00/1efgh234ij56klmno78pqrstu90vwxyx
Users:
- Arn: arn:aws:iam::123456789012:user/Admin
  CreateDate: '2014-10-16T16:03:09+00:00'
  PasswordLastUsed: '2016-06-03T18:37:29+00:00'
  Path: /
  UserId: AIDA111111111111EXAMPLE
  UserName: Admin
- Arn: arn:aws:iam::123456789012:user/backup/backup-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /backup/
  UserId: AIDA222222222222EXAMPLE
```

```

  UserName: arq-45EFD6D1-CE56-459B-B39F-F9C1F78FBE19
- IsTruncated: false
Users:
- Arn: arn:aws:iam::123456789012:user/cli-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /
  UserId: AIDA333333333333EXAMPLE
  UserName: cli-user

```

## Format keluaran teks

`textFormat` mengatur AWS CLI output ke dalam baris yang dibatasi tab. Ini bekerja dengan baik dengan alat teks Unix tradisional seperti `grep`, `sed`, dan `awk`, dan pemrosesan teks yang dilakukan oleh PowerShell.

Format `text` output mengikuti struktur dasar yang ditunjukkan di bawah ini. Kolom diurutkan menurut abjad dengan nama kunci yang sesuai dari objek JSON yang mendasarinya.

```

IDENTIFIER  sorted-column1 sorted-column2
IDENTIFIER2 sorted-column1 sorted-column2

```

Berikut ini adalah contoh `text` output. Setiap bidang adalah tab yang terpisah dari yang lain, dengan tab tambahan di mana ada bidang kosong.

```
$ aws iam list-users --output text
```

```

USERS  arn:aws:iam::123456789012:user/Admin                2014-10-16T16:03:09+00:00
2016-06-03T18:37:29+00:00 / AIDA111111111111EXAMPLE Admin
USERS  arn:aws:iam::123456789012:user/backup/backup-user      2019-09-17T19:30:40+00:00
/backup/ AIDA222222222222EXAMPLE backup-user
USERS  arn:aws:iam::123456789012:user/cli-user                2019-09-17T19:11:39+00:00
/ AIDA333333333333EXAMPLE cli-user

```

Kolom keempat adalah `PasswordLastUsed` bidang, dan kosong untuk dua entri terakhir karena pengguna tersebut tidak pernah masuk ke AWS Management Console.

### Important

Kami sangat menyarankan bahwa jika Anda menentukan `text` output, Anda juga selalu menggunakan `--query` opsi untuk memastikan perilaku yang konsisten.

Ini karena format teks menurut abjad mengurutkan kolom keluaran dengan nama kunci objek JSON yang mendasari yang dikembalikan oleh AWS layanan, dan sumber daya serupa mungkin tidak memiliki nama kunci yang sama. Misalnya, representasi JSON dari instans Amazon EC2 berbasis Linux mungkin memiliki elemen yang tidak ada dalam representasi JSON dari instance berbasis Windows, atau sebaliknya. Selain itu, sumber daya mungkin memiliki elemen nilai kunci yang ditambahkan atau dihapus di pembaruan masa depan, mengubah urutan kolom. Di sinilah `--query` menambah fungsionalitas text output untuk memberi Anda kontrol penuh atas format output.

Dalam contoh berikut, perintah menentukan elemen mana yang akan ditampilkan dan mendefinisikan urutan kolom dengan notasi daftar. `[key1, key2, ...]` Ini memberi Anda keyakinan penuh bahwa nilai kunci yang benar selalu ditampilkan di kolom yang diharapkan. Akhirnya, perhatikan bagaimana AWS CLI output `None` sebagai nilai untuk kunci yang tidak ada.

```
$ aws iam list-users --output text --query 'Users[*].
[UserName,Arn,CreateDate,PasswordLastUsed,UserId]'
```

```
Admin          arn:aws:iam::123456789012:user/Admin
2014-10-16T16:03:09+00:00  2016-06-03T18:37:29+00:00  AIDA111111111111EXAMPLE
backup-user    arn:aws:iam::123456789012:user/backup-user
2019-09-17T19:30:40+00:00  None                        AIDA222222222222EXAMPLE
cli-user       arn:aws:iam::123456789012:user/cli-backup
2019-09-17T19:11:39+00:00  None                        AIDA333333333333EXAMPLE
```

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan `grep` dan `awk` dengan text output dari `aws ec2 describe-instances` perintah. Perintah pertama menampilkan Availability Zone, status saat ini, dan ID instance dari setiap instance dalam text output. Perintah kedua memproses output yang hanya menampilkan ID instance dari semua instance yang berjalan di `us-west-2a` Availability Zone.

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text
```

```
us-west-2a    running i-4b41a37c
us-west-2a    stopped i-a071c394
us-west-2b    stopped i-97a217a0
us-west-2a    running i-3045b007
```

```
us-west-2a      running i-6fc67758
```

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text | grep us-west-2a |
grep running | awk '{print $3}'
```

```
i-4b41a37c
i-3045b007
i-6fc67758
```

Contoh berikut melangkah lebih jauh dan menunjukkan tidak hanya cara memfilter output, tetapi bagaimana menggunakan output itu untuk mengotomatiskan jenis instance yang berubah untuk setiap instance yang dihentikan.

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].[State.Name,
InstanceId]' --output text |
> grep stopped |
> awk '{print $2}' |
> while read line;
> do aws ec2 modify-instance-attribute --instance-id $line --instance-type '{"Value":
"m1.medium"}';
> done
```

textOutputnya juga bisa berguna di PowerShell. Karena kolom dalam text output dibatasi tab, Anda dapat dengan mudah membagi output menjadi array dengan menggunakan PowerShell pembatas. Perintah berikut menampilkan nilai kolom ketiga (InstanceId) jika kolom pertama (AvailabilityZone) cocok dengan string us-west-2a.

```
PS C:\>aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text |
%{if ($_.split("`t")[0] -match "us-west-2a") { $_.split("`t")[2]; } }
```

```
-4b41a37c
i-a071c394
i-3045b007
i-6fc67758
```

Perhatikan bahwa meskipun contoh sebelumnya menunjukkan cara menggunakan --query parameter untuk mengurai objek JSON yang mendasarinya dan mengeluarkan kolom yang



diinginkan, PowerShell memiliki kemampuannya sendiri untuk menangani JSON, jika kompatibilitas lintas platform tidak menjadi perhatian. Alih-alih menangani output sebagai teks, seperti yang dibutuhkan kebanyakan shell perintah, PowerShell memungkinkan Anda menggunakan `ConvertFrom-JSON` cmdlet untuk menghasilkan objek yang terstruktur secara hierarkis. Anda kemudian dapat langsung mengakses anggota yang Anda inginkan dari objek itu.

```
(aws ec2 describe-instances --output json | ConvertFrom-Json).Reservations.Instances.InstanceId
```

### Tip

Jika Anda mengeluarkan teks, dan memfilter output ke satu bidang menggunakan `--query` parameter, outputnya adalah satu baris dari nilai yang dipisahkan tab. Untuk mendapatkan setiap nilai ke baris terpisah, Anda dapat menempatkan bidang output dalam tanda kurung, seperti yang ditunjukkan pada contoh berikut.

Tab dipisahkan, output satu baris:

```
$ aws iam list-groups-for-user --user-name susan --output text --query "Groups[].GroupName"
```

```
HRDepartment    Developers    SpreadsheetUsers    LocalAdmins
```

Setiap nilai pada barisnya sendiri dengan memasukkan `[GroupName]` tanda kurung:

```
$ aws iam list-groups-for-user --user-name susan --output text --query "Groups[].[GroupName]"
```

```
HRDepartment
Developers
SpreadsheetUsers
LocalAdmins
```

## Format keluaran tabel

`tableFormat` ini menghasilkan representasi yang dapat dibaca manusia dari AWS CLI output kompleks dalam bentuk tabel.

```
$ aws iam list-users --output table
```

```
-----
|
| ListUsers |
+-----+
|
| Users |
+-----+
|+-----+-----+-----+-----+|
+-----+-----+-----+-----+|
| | Arn | CreateDate |
| PasswordLastUsed | Path | UserId | Username |
+-----+-----+-----+-----+|
+-----+-----+-----+-----+|
| | arn:aws:iam::123456789012:user/Admin | 2014-10-16T16:03:09+00:00 | |
| 2016-06-03T18:37:29+00:00 | / | AIDA111111111111EXAMPLE | Admin |
| | arn:aws:iam::123456789012:user/backup/backup-user | 2019-09-17T19:30:40+00:00 |
| | /backup/ | AIDA222222222222EXAMPLE | backup-user |
| | arn:aws:iam::123456789012:user/cli-user | 2019-09-17T19:11:39+00:00 |
| | / | AIDA333333333333EXAMPLE | cli-user |
+-----+-----+-----+-----+|
+

```

Anda dapat menggabungkan `--query` opsi dengan `table` format untuk menampilkan satu set elemen yang telah dipilih sebelumnya dari output mentah. Perhatikan perbedaan output antara notasi kamus dan daftar: pada contoh pertama, nama kolom diurutkan menurut abjad, dan pada contoh kedua, kolom yang tidak disebutkan namanya diurutkan seperti yang ditentukan oleh pengguna. Untuk informasi selengkapnya tentang `--query` opsi, lihat [Filter AWS CLI keluaran](#).

```
$ aws ec2 describe-volumes --query 'Volumes[*].
{ID:VolumeId,InstanceId:Attachments[0].InstanceId,AZ:AvailabilityZone,Size:Size}' --
output table
```

```
-----
| DescribeVolumes |
+-----+-----+-----+-----+
| AZ | ID | InstanceId | Size |
+-----+-----+-----+-----+
| us-west-2a | vol-e11a5288 | i-a071c394 | 30 |
| us-west-2a | vol-2e410a47 | i-4b41a37c | 8 |

```

```
+-----+-----+-----+-----+
```

```
$ aws ec2 describe-volumes --query 'Volumes[*].
[VolumeId,Attachments[0].InstanceId,AvailabilityZone,Size]' --output table
```

```
-----
|                               DescribeVolumes                               |
+-----+-----+-----+-----+
| vol-e11a5288| i-a071c394 | us-west-2a | 30 |
| vol-2e410a47| i-4b41a37c | us-west-2a | 8  |
+-----+-----+-----+-----+
```

## Gunakan AWS CLI opsi pagination

Topik ini menjelaskan berbagai cara untuk melakukan paginasi output dari AWS CLI

Terutama ada dua cara untuk mengontrol pagination dari AWS CLI

- [Menggunakan parameter pagination sisi server.](#)
- [Menggunakan program paging sisi klien keluaran default Anda.](#)

Parameter pagination sisi server diproses terlebih dahulu dan output apa pun dikirim ke pagination sisi klien.

### Pagination sisi server

Untuk perintah yang dapat mengembalikan daftar item yang besar, AWS Command Line Interface (AWS CLI) memiliki beberapa opsi untuk mengontrol jumlah item yang disertakan dalam output saat AWS CLI memanggil API layanan untuk mengisi daftar.

Pilihannya meliputi yang berikut:

- [Cara menggunakan parameter --no-paginate](#)
- [Cara menggunakan parameter --page-size](#)
- [Cara menggunakan parameter --max-items](#)
- [Cara menggunakan parameter --starting-token](#)

Secara default, AWS CLI menggunakan ukuran halaman yang ditentukan oleh layanan individu dan mengambil semua item yang tersedia. Misalnya, Amazon S3 memiliki ukuran halaman default 1000.

Jika Anda menjalankan `aws s3api list-objects` bucket Amazon S3 yang berisi 3.500 objek, AWS CLI secara otomatis membuat empat panggilan ke Amazon S3, menangani logika pagination khusus layanan untuk Anda di latar belakang dan mengembalikan semua 3.500 objek di output akhir.

### Cara menggunakan parameter `--no-paginate`

`--no-paginate` Opsi ini menonaktifkan token pagination berikut di sisi klien. Saat menggunakan perintah, secara default AWS CLI secara otomatis membuat beberapa panggilan untuk mengembalikan semua hasil yang mungkin untuk membuat pagination. Satu panggilan untuk setiap halaman. Menonaktifkan pagination memiliki AWS CLI satu-satunya panggilan sekali untuk halaman pertama hasil perintah.

Misalnya, jika Anda menjalankan `aws s3api list-objects` bucket Amazon S3 yang berisi 3.500 objek, AWS CLI satu-satunya yang membuat panggilan pertama ke Amazon S3, hanya mengembalikan 1.000 objek pertama dalam output akhir.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --no-paginate  
{  
  "Contents": [  
  ...
```

### Cara menggunakan parameter `--page-size`

Jika Anda melihat masalah saat menjalankan perintah daftar pada sejumlah besar sumber daya, ukuran halaman default mungkin terlalu tinggi. Hal ini dapat menyebabkan panggilan ke AWS layanan melebihi waktu maksimum yang diizinkan dan menghasilkan kesalahan “waktu habis”. Anda dapat menggunakan `--page-size` opsi untuk menentukan bahwa AWS CLI permintaan sejumlah kecil item dari setiap panggilan ke AWS layanan. AWS CLI Masih mengambil daftar lengkap, tetapi melakukan lebih banyak panggilan API layanan di latar belakang dan mengambil sejumlah kecil item dengan setiap panggilan. Ini memberi panggilan individu kesempatan yang lebih baik untuk berhasil tanpa batas waktu. Mengubah ukuran halaman tidak memengaruhi output; itu hanya memengaruhi jumlah panggilan API yang perlu dilakukan untuk menghasilkan output.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --page-size 100  
{
```

```
"Contents": [  
...
```

## Cara menggunakan parameter `--max-items`

Untuk memasukkan lebih sedikit item sekaligus dalam AWS CLI output, gunakan `--max-items` opsi. AWS CLIMasih menangani pagination dengan layanan seperti yang dijelaskan sebelumnya, tetapi hanya mencetak jumlah item pada waktu yang Anda tentukan.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --max-items 100  
{  
  "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ==",  
  "Contents": [  
  ...
```

## Cara menggunakan parameter `--starting-token`

Jika jumlah item output (`--max-items`) lebih sedikit dari jumlah total item yang dikembalikan oleh panggilan API yang mendasarinya, outputnya menyertakan perintah `NextToken` yang dapat Anda teruskan ke perintah berikutnya untuk mengambil set item berikutnya. Contoh berikut menunjukkan bagaimana menggunakan `NextToken` nilai yang dikembalikan oleh contoh sebelumnya, dan memungkinkan Anda untuk mengambil 100 item kedua.

### Note

Parameter `--starting-token` tidak bisa nol atau kosong. Jika perintah sebelumnya tidak mengembalikan `NextToken` nilai, tidak ada lagi item untuk dikembalikan dan Anda tidak perlu memanggil perintah lagi.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --max-items 100 \  
  --starting-token eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ==  
{  
  "Contents": [  
  ...
```

AWS Layanan yang ditentukan mungkin tidak mengembalikan item dalam urutan yang sama setiap kali Anda menelepon. Jika Anda menentukan nilai yang berbeda untuk `--page-size` dan `--max-items`, Anda bisa mendapatkan hasil yang tidak terduga dengan item yang hilang atau digandakan. Untuk mencegah hal ini, gunakan nomor yang sama untuk `--page-size` dan `--max-items` untuk menyinkronkan AWS CLI pagination dengan pagination dari layanan yang mendasarinya. Anda juga dapat mengambil seluruh daftar dan melakukan operasi paging yang diperlukan secara lokal.

## Pager sisi klien

AWS CLI versi 2 menyediakan penggunaan program pager sisi klien untuk output. Secara default, fitur ini mengembalikan semua output melalui program pager default sistem operasi Anda.

Dalam urutan prioritas, Anda dapat menentukan pager output dengan cara berikut:

- Menggunakan `cli_pager` pengaturan dalam `config` file di profil default atau bernama.
- Menggunakan variabel `AWS_PAGER` lingkungan.
- Menggunakan variabel `PAGER` lingkungan.

Dalam urutan prioritas, Anda dapat menonaktifkan semua penggunaan program paging eksternal dengan cara berikut:

- Gunakan opsi baris `--no-cli-pager` perintah untuk menonaktifkan pager untuk penggunaan perintah tunggal.
- Atur `cli_pager` pengaturan atau `AWS_PAGER` variabel ke string kosong.

Topik pager sisi klien:

- [Cara menggunakan pengaturan `cli\_pager`](#)
- [Cara menggunakan variabel lingkungan `AWS\_PAGER`](#)
- [Cara menggunakan `no-cli-pager` opsi `--`](#)
- [Cara menggunakan bendera pager](#)

Cara menggunakan pengaturan `cli_pager`

Anda dapat menyimpan pengaturan konfigurasi dan kredensial yang sering digunakan dalam file yang dikelola oleh file. AWS CLI Pengaturan di profil nama lebih diutamakan daripada pengaturan di

profil. default Untuk informasi selengkapnya tentang pengaturan konfigurasi, lihat [Konfigurasi dan pengaturan file kredensi](#).

Contoh berikut menetapkan default output pager untuk `less` program.

```
[default]
cli_pager=less
```

Contoh berikut menetapkan default untuk menonaktifkan penggunaan pager.

```
[default]
cli_pager=
```

Cara menggunakan variabel lingkungan `AWS_PAGER`

Contoh berikut menetapkan default output pager untuk `less` program. Untuk informasi lebih lanjut tentang variabel lingkungan, lihat [Variabel lingkungan untuk mengkonfigurasi AWS CLI](#).

Linux and macOS

```
$ export AWS_PAGER="less"
```

Windows

```
C:\> setx AWS_PAGER "less"
```

Contoh berikut menonaktifkan penggunaan pager.

Linux and macOS

```
$ export AWS_PAGER=""
```

Windows

```
C:\> setx AWS_PAGER ""
```

## Cara menggunakan no-cli-pager opsi --

Untuk menonaktifkan penggunaan pager pada satu perintah, gunakan `--no-cli-pager` opsi. Untuk informasi selengkapnya tentang opsi baris perintah, lihat [Opsi baris perintah](#).

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --no-cli-pager  
{  
  "Contents": [  
  ...
```

## Cara menggunakan bendera pager

Anda dapat menentukan bendera untuk digunakan secara otomatis dengan program paging Anda. Bendera tergantung pada program paging yang Anda gunakan. Contoh di bawah ini adalah untuk default khas dan. `less more`

### Linux and macOS

Jika Anda tidak menentukan sebaliknya, pager AWS CLI versi 2 menggunakan secara default adalah `less`. Jika Anda tidak memiliki variabel `LESS` lingkungan yang disetel, AWS CLI versi 2 menggunakan `FRX` bendera. Anda dapat menggabungkan bendera dengan menentukannya saat mengatur pager. AWS CLI

Contoh berikut menggunakan `S` bendera. Bendera ini kemudian digabungkan dengan `FRX` bendera default untuk membuat `FRXS` bendera akhir.

```
$ export AWS_PAGER="less -S"
```

Jika Anda tidak menginginkan `FRX` bendera apa pun, Anda dapat meniadakannya. Contoh berikut meniadakan `F` bendera untuk membuat `RX` bendera akhir.

```
$ export AWS_PAGER="less -+F"
```

Untuk informasi lebih lanjut tentang `less` bendera, lihat lebih [sedikit](#) di [manpages.org](http://manpages.org).

### Windows

Jika Anda tidak menentukan sebaliknya, pager AWS CLI versi 2 yang digunakan secara default adalah `more` tanpa tanda tambahan.



Contoh berikut menggunakan `/c` parameter.

```
C:\> setx AWS_PAGER "more /c"
```

Untuk informasi selengkapnya tentang `more` bendera, lihat [selengkapnya](#) di Microsoft Docs.

## Filter AWS CLI keluaran

The AWS Command Line Interface (AWS CLI) memiliki penyaringan sisi server dan sisi klien yang dapat Anda gunakan secara individual atau bersama-sama untuk memfilter output Anda. AWS CLI Pemfilteran sisi server diproses terlebih dahulu dan mengembalikan output Anda untuk pemfilteran sisi klien.

- Pemfilteran sisi server didukung oleh API, dan Anda biasanya menerapkannya dengan parameter. `--filter` Layanan hanya mengembalikan hasil yang cocok yang dapat mempercepat waktu respons HTTP untuk kumpulan data besar.
- Pemfilteran sisi klien didukung oleh AWS CLI klien menggunakan parameter. `--query` Parameter ini memiliki kemampuan yang mungkin tidak dimiliki oleh penyaringan sisi server.

### Topik

- [Pemfilteran sisi server](#)
- [Penyaringan sisi klien](#)
- [Menggabungkan penyaringan sisi server dan sisi klien](#)
- [Sumber daya tambahan](#)

## Pemfilteran sisi server

Pemfilteran sisi server di AWS CLI disediakan oleh API layanan. AWS AWS Layanan hanya mengembalikan catatan dalam respons HTTP yang cocok dengan filter Anda, yang dapat mempercepat waktu respons HTTP untuk kumpulan data besar. Karena pemfilteran sisi server ditentukan oleh API layanan, nama dan fungsi parameter bervariasi antar layanan. Beberapa nama parameter umum yang digunakan untuk penyaringan adalah:

- `--filter` seperti dan [ce](#).
- `--filters`

- Nama dimulai dengan katafilter, misalnya `--filter-expression` untuk perintah.

[Untuk informasi tentang apakah perintah tertentu memiliki pemfilteran sisi server dan aturan pemfilteran, lihat panduan referensi referensi versi 2 .](#)

## Penyaringan sisi klien

AWS CLI Ini menyediakan kemampuan penyaringan sisi klien berbasis JSON bawaan dengan parameter. `--query` `--queryParameter` adalah alat yang ampuh yang dapat Anda gunakan untuk menyesuaikan konten dan gaya output Anda. `--queryParameter` mengambil respons HTTP yang kembali dari server dan memfilter hasil sebelum menampilkannya. Karena seluruh respons HTTP dikirim ke klien sebelum pemfilteran, pemfilteran sisi klien bisa lebih lambat daripada penyaringan sisi server untuk kumpulan data besar.

Query menggunakan [sintaks JMESPath](#) untuk membuat ekspresi untuk memfilter output Anda. Untuk mempelajari sintaks JMESPath, lihat [Tutorial](#) di situs web JMESPath.

### Important

Jenis keluaran yang Anda tentukan mengubah cara `--query` opsi beroperasi:

- Jika Anda menentukan `--output text`, output diberi paginasi sebelum `--query` filter diterapkan, dan AWS CLI menjalankan kueri sekali pada setiap halaman output. Karena ini, kueri menyertakan elemen pencocokan pertama pada setiap halaman yang dapat menghasilkan output tambahan yang tidak terduga. Untuk memfilter output tambahan, Anda dapat menggunakan alat baris perintah lainnya seperti `head` atau `tail`.
- Jika Anda menentukan `--output json` `--output yaml`, atau `--output yaml-stream` output sepenuhnya diproses sebagai struktur asli tunggal sebelum `--query` filter diterapkan. AWS CLI Menjalankan kueri hanya sekali terhadap seluruh struktur, menghasilkan hasil yang difilter yang kemudian dikeluarkan.

## Topik penyaringan sisi klien

- [Sebelum Anda mulai](#)
- [Pengidentifikasi](#)
- [Memilih dari daftar](#)
- [Memfilter data bersarang](#)

- [Hasil perataan](#)
- [Pemfilteran untuk nilai tertentu](#)
- [Ekspresi perpipaan](#)
- [Memfilter untuk beberapa nilai pengenalan](#)
- [Menambahkan label ke nilai pengenalan](#)
- [Fungsi](#)
- [--queryContoh lanjutan](#)

Sebelum Anda mulai

Saat menggunakan ekspresi filter yang digunakan dalam contoh ini, pastikan untuk menggunakan aturan kutipan yang benar untuk shell terminal Anda. Untuk informasi selengkapnya, lihat [the section called "Kutipan dengan Strings"](#).

Output JSON berikut menunjukkan contoh dari apa `--query` parameter dapat menghasilkan. Outputnya menjelaskan tiga volume Amazon EBS yang dilampirkan ke instans Amazon EC2 yang terpisah.

Contoh Output

```
$ aws ec2 describe-volumes
{
  "Volumes": [
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-17T00:55:03.000Z",
          "InstanceId": "i-a071c394",
          "VolumeId": "vol-e11a5288",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-e11a5288",
      "State": "in-use",
      "SnapshotId": "snap-f23ec1c8",
      "CreateTime": "2013-09-17T00:55:03.000Z",
```

```
    "Size": 30
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-18T20:26:16.000Z",
        "InstanceId": "i-4b41a37c",
        "VolumeId": "vol-2e410a47",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-2e410a47",
    "State": "in-use",
    "SnapshotId": "snap-708e8348",
    "CreateTime": "2013-09-18T20:26:15.000Z",
    "Size": 8
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  }
]
}
```

## Pengidentifikasi

Identifier adalah label untuk nilai output. Saat membuat filter, Anda menggunakan pengidentifikasi untuk mempersempit hasil kueri Anda. Dalam contoh output berikut, semua pengidentifikasi seperti `Volumes`, `AvailabilityZone`, dan `AttachTime` disorot.

```
$ aws ec2 describe-volumes
{
  "Volumes": [
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-17T00:55:03.000Z",
          "InstanceId": "i-a071c394",
          "VolumeId": "vol-e11a5288",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-e11a5288",
      "State": "in-use",
      "SnapshotId": "snap-f23ec1c8",
      "CreateTime": "2013-09-17T00:55:03.000Z",
      "Size": 30
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-18T20:26:16.000Z",
          "InstanceId": "i-4b41a37c",
          "VolumeId": "vol-2e410a47",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-2e410a47",
      "State": "in-use",
```

```

    "SnapshotId": "snap-708e8348",
    "CreateTime": "2013-09-18T20:26:15.000Z",
    "Size": 8
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  }
]
}

```

Untuk informasi selengkapnya, lihat [Pengidentifikasi](#) di situs web JMESPath.

Memilih dari daftar

Daftar atau array adalah pengidentifikasi yang diikuti oleh braket persegi "[" seperti `Volumes` dan `Attachments` di [the section called "Sebelum Anda mulai"](#)

Sintaksis

```
<listName>[ ]
```

Untuk memfilter semua output dari array, Anda dapat menggunakan notasi wildcard. Ekspresi [wildcard](#) adalah ekspresi yang digunakan untuk mengembalikan elemen menggunakan \* notasi.

Contoh berikut menanyakan semua `Volumes` konten.

```
$ aws ec2 describe-volumes \
```

```
--query 'Volumes[*]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  }
]
```

Untuk melihat volume tertentu dalam array dengan indeks, Anda memanggil indeks array. Misalnya, item pertama dalam `Volumes` array memiliki indeks 0, menghasilkan `Volumes[0]` kueri. Untuk informasi selengkapnya tentang indeks array, lihat [ekspresi indeks](#) di situs web JMESPath.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[0]'
{
  "AvailabilityZone": "us-west-2a",
  "Attachments": [
    {
      "AttachTime": "2013-09-17T00:55:03.000Z",
      "InstanceId": "i-a071c394",
      "VolumeId": "vol-e11a5288",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  "VolumeType": "standard",
  "VolumeId": "vol-e11a5288",
  "State": "in-use",
  "SnapshotId": "snap-f23ec1c8",
  "CreateTime": "2013-09-17T00:55:03.000Z",
  "Size": 30
}
```

Untuk melihat rentang volume tertentu berdasarkan indeks, gunakan `slice` dengan sintaks berikut, di mana awal adalah indeks array awal, stop adalah indeks tempat filter berhenti memproses, dan langkahnya adalah interval lewati.

## Sintaksis

```
<arrayName>[<start>:<stop>:<step>]
```

Jika salah satu dari ini dihilangkan dari ekspresi irisan, mereka menggunakan nilai default berikut:

- Mulai — Indeks pertama dalam daftar, 0.
- Stop — Indeks terakhir dalam daftar.
- Langkah - Tidak ada langkah melompati, di mana nilainya adalah 1.



Untuk mengembalikan hanya dua volume pertama, Anda menggunakan nilai awal 0, nilai stop 2, dan nilai langkah 1 seperti yang ditunjukkan pada contoh berikut.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[0:2:1]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-18T20:26:16.000Z",
        "InstanceId": "i-4b41a37c",
        "VolumeId": "vol-2e410a47",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-2e410a47",
    "State": "in-use",
    "SnapshotId": "snap-708e8348",
    "CreateTime": "2013-09-18T20:26:15.000Z",
    "Size": 8
  }
]
```

```
}  
]
```

Karena contoh ini berisi nilai default, Anda dapat mempersingkat irisan dari `Volumes[0:2:1]` ke `Volumes[:2]`.

Contoh berikut menghilangkan nilai default dan mengembalikan setiap dua volume di seluruh array.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[:2]'  
[  
  {  
    "AvailabilityZone": "us-west-2a",  
    "Attachments": [  
      {  
        "AttachTime": "2013-09-17T00:55:03.000Z",  
        "InstanceId": "i-a071c394",  
        "VolumeId": "vol-e11a5288",  
        "State": "attached",  
        "DeleteOnTermination": true,  
        "Device": "/dev/sda1"  
      }  
    ],  
    "VolumeType": "standard",  
    "VolumeId": "vol-e11a5288",  
    "State": "in-use",  
    "SnapshotId": "snap-f23ec1c8",  
    "CreateTime": "2013-09-17T00:55:03.000Z",  
    "Size": 30  
  },  
  {  
    "AvailabilityZone": "us-west-2a",  
    "Attachments": [  
      {  
        "AttachTime": "2020-11-20T19:54:06.000Z",  
        "InstanceId": "i-1jd73kv8",  
        "VolumeId": "vol-a1b3c7nd",  
        "State": "attached",  
        "DeleteOnTermination": true,  
        "Device": "/dev/sda1"  
      }  
    ],  
    "VolumeType": "standard",
```

```
"VolumeId": "vol-a1b3c7nd",
"State": "in-use",
"SnapshotId": "snap-234087fb",
"CreateTime": "2020-11-20T19:54:05.000Z",
"Size": 15
}
]
```

Langkah-langkah juga dapat menggunakan angka negatif untuk memfilter dalam urutan terbalik dari array seperti yang ditunjukkan pada contoh berikut.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[::-2]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ]
  }
]
```

```

    }
  ],
  "VolumeType": "standard",
  "VolumeId": "vol-e11a5288",
  "State": "in-use",
  "SnapshotId": "snap-f23ec1c8",
  "CreateTime": "2013-09-17T00:55:03.000Z",
  "Size": 30
}
]

```

Untuk informasi selengkapnya, lihat [Slices](#) di situs web JMESPath.

### Memfilter data bersarang

Untuk mempersempit pemfilteran `Volumes[*]` untuk nilai bersarang, Anda menggunakan subexpressions dengan menambahkan periode dan kriteria filter Anda.

### Sintaksis

```
<expression>.<expression>
```

Contoh berikut menunjukkan semua Attachments informasi untuk semua volume.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments'
[
  [
    {
      "AttachTime": "2013-09-17T00:55:03.000Z",
      "InstanceId": "i-a071c394",
      "VolumeId": "vol-e11a5288",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  [
    {
      "AttachTime": "2013-09-18T20:26:16.000Z",
      "InstanceId": "i-4b41a37c",
      "VolumeId": "vol-2e410a47",
      "State": "attached",

```

```

    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
],
[
  {
    "AttachTime": "2020-11-20T19:54:06.000Z",
    "InstanceId": "i-1jd73kv8",
    "VolumeId": "vol-a1b3c7nd",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
]
]
```

Untuk memfilter lebih jauh ke dalam nilai bersarang, tambahkan ekspresi untuk setiap indentifier bersarang. Contoh berikut mencantumkan State untuk semuaVolumes.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[*].State'
[
  [
    "attached"
  ],
  [
    "attached"
  ],
  [
    "attached"
  ]
]
```

### Hasil perataan

Untuk informasi lebih lanjut, lihat [SubExpressions](#) di situs web JMESPath.

Anda dapat meratakan hasilnya `Volumes[*].Attachments[*].State` dengan menghapus notasi wildcard yang menghasilkan kueri. `Volumes[*].Attachments[].State` Meratakan sering berguna untuk meningkatkan keterbacaan hasil.

```

$ aws ec2 describe-volumes \
```

```
--query 'Volumes[*].Attachments[].State'
[
  "attached",
  "attached",
  "attached"
]
```

Untuk informasi lebih lanjut, lihat [Flatten](#) di situs web JMESPath.

### Pemfilteran untuk nilai tertentu

Untuk memfilter nilai tertentu dalam daftar, Anda menggunakan ekspresi filter seperti yang ditunjukkan dalam sintaks berikut.

### Sintaksis

```
? <expression> <comparator> <expression>]
```

Komparator ekspresi meliputi ==, !=, <, >, <=>, dan >=. Contoh filter berikut VolumeIds untuk semua Volumes dalam file AttachedState.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[?State==`attached`].VolumeId'
[
  [
    "vol-e11a5288"
  ],
  [
    "vol-2e410a47"
  ],
  [
    "vol-a1b3c7nd"
  ]
]
```

Ini kemudian dapat diratakan sehingga menghasilkan contoh berikut.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[?State==`attached`].VolumeId[]'
[
  "vol-e11a5288",
```

```
"vol-2e410a47",
"vol-a1b3c7nd"
]
```

Berikut contoh filter untuk semua VolumeIds Volumes yang memiliki ukuran kurang dari 20.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[?Size < `20`].VolumeId'
[
  "vol-2e410a47",
  "vol-a1b3c7nd"
]
```

Untuk informasi selengkapnya, lihat [Menyaring Ekspresi](#) di situs web JMESPath.

Ekspresi perpipaan

Anda dapat menyalurkan hasil filter ke daftar baru, dan kemudian memfilter hasilnya dengan ekspresi lain menggunakan sintaks berikut:

Sintaksis

```
<expression> | <expression>]
```

Contoh berikut mengambil hasil filter `Volumes[*].Attachments[].InstanceId` ekspresi dan output hasil pertama dalam array.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[].InstanceId | [0]'
"i-a071c394"
```

Contoh ini melakukan ini dengan terlebih dahulu membuat array dari ekspresi berikut.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[].InstanceId'
"i-a071c394",
"i-4b41a37c",
"i-1jd73kv8"
```

Dan kemudian mengembalikan elemen pertama dalam array itu.

```
"i-a071c394"
```

Untuk informasi selengkapnya, lihat [Ekspresi Pipa](#) di situs web JMESPath.

Memfilter untuk beberapa nilai pengenalan

Untuk memfilter beberapa pengidentifikasi, Anda menggunakan daftar multiselect dengan menggunakan sintaks berikut:

Sintaksis

```
<listName>[].[<expression>, <expression>]
```

Dalam contoh berikut, VolumeId dan VolumeType disaring dalam Volumes daftar yang menghasilkan ekspresi berikut.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[][VolumeId, VolumeType]'  
[  
  [  
    "vol-e11a5288",  
    "standard"  
  ],  
  [  
    "vol-2e410a47",  
    "standard"  
  ],  
  [  
    "vol-a1b3c7nd",  
    "standard"  
  ]  
]
```

Untuk menambahkan data bersarang ke daftar, Anda menambahkan daftar multipilih lainnya. Contoh berikut memperluas contoh sebelumnya dengan juga memfilter untuk InstanceId dan State dalam daftar bersarangAttachments. Ini menghasilkan ekspresi berikut.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[][VolumeId, VolumeType, Attachments[][InstanceId, State]]'  
[  
  [  
    [  
      [  
        "vol-e11a5288",  
        "standard",  
        [  
          [  
            "i-a071c394",  
            "running"  
          ]  
        ]  
      ]  
    ]  
  ]  
]
```



```

    "vol-e11a5288",
    "standard",
    [
      [
        "i-a071c394",
        "attached"
      ]
    ]
  ],
  [
    "vol-2e410a47",
    "standard",
    [
      [
        "i-4b41a37c",
        "attached"
      ]
    ]
  ],
  [
    "vol-a1b3c7nd",
    "standard",
    [
      [
        "i-1jd73kv8",
        "attached"
      ]
    ]
  ]
]

```

Agar lebih mudah dibaca, ratakan ekspresi seperti yang ditunjukkan pada contoh berikut.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[][VolumeId, VolumeType, Attachments[][InstanceId, State][]][]'
[
  "vol-e11a5288",
  "standard",
  [
    "i-a071c394",
    "attached"
  ],
  "vol-2e410a47",

```

```

"standard",
[
  "i-4b41a37c",
  "attached"
],
"vol-a1b3c7nd",
"standard",
[
  "i-1jd73kv8",
  "attached"
]
]

```

Untuk informasi selengkapnya, lihat [Daftar multiselect](#) di situs web JMESPath.

Menambahkan label ke nilai pengenalan

Untuk membuat output ini lebih mudah dibaca, gunakan hash multiselect dengan sintaks berikut.

Sintaksis

```
<listName>[].{<label>: <expression>, <label>: <expression>}
```

Label pengenalan Anda tidak harus sama dengan nama pengenalan. Contoh berikut menggunakan label `VolumeType` untuk `VolumeType` nilai-nilai.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[].{VolumeType: VolumeType}'
[
  {
    "VolumeType": "standard",
  },
  {
    "VolumeType": "standard",
  },
  {
    "VolumeType": "standard",
  }
]

```

Untuk mempermudah, contoh berikut menyimpan nama pengenalan untuk setiap label dan menampilkan `VolumeId`, `VolumeTypeInstanceId`, dan `State` untuk semua volume:

```
$ aws ec2 describe-volumes \
  --query 'Volumes[].{VolumeId: VolumeId, VolumeType: VolumeType, InstanceId:
  Attachments[0].InstanceId, State: Attachments[0].State}'
[
  {
    "VolumeId": "vol-e11a5288",
    "VolumeType": "standard",
    "InstanceId": "i-a071c394",
    "State": "attached"
  },
  {
    "VolumeId": "vol-2e410a47",
    "VolumeType": "standard",
    "InstanceId": "i-4b41a37c",
    "State": "attached"
  },
  {
    "VolumeId": "vol-a1b3c7nd",
    "VolumeType": "standard",
    "InstanceId": "i-1jd73kv8",
    "State": "attached"
  }
]
```

Untuk informasi selengkapnya, lihat [hash Multiselect](#) di situs web JMESPath.

## Fungsi

Sintaks JMESPath berisi banyak fungsi yang dapat Anda gunakan untuk kueri Anda. Untuk informasi tentang fungsi JMESPath, lihat Fungsi [Bawaan di situs web](#) JMESPath.

Untuk menunjukkan bagaimana Anda dapat menggabungkan fungsi ke dalam kueri Anda, contoh berikut menggunakan `sort_by` fungsi tersebut. `sort_by` Fungsi mengurutkan array menggunakan ekspresi sebagai kunci sortir menggunakan sintaks berikut:

## Sintaksis

```
sort_by(<listName>, <sort expression>)[].<expression>
```

Contoh berikut menggunakan contoh [hash multiselect](#) sebelumnya dan mengurutkan output berdasarkan `VolumeId`

```
$ aws ec2 describe-volumes \
  --query 'sort_by(Volumes, &VolumeId)[].{VolumeId: VolumeId, VolumeType: VolumeType,
  InstanceId: Attachments[0].InstanceId, State: Attachments[0].State}'
[
  {
    "VolumeId": "vol-2e410a47",
    "VolumeType": "standard",
    "InstanceId": "i-4b41a37c",
    "State": "attached"
  },
  {
    "VolumeId": "vol-a1b3c7nd",
    "VolumeType": "standard",
    "InstanceId": "i-1jd73kv8",
    "State": "attached"
  },
  {
    "VolumeId": "vol-e11a5288",
    "VolumeType": "standard",
    "InstanceId": "i-a071c394",
    "State": "attached"
  }
]
```

Untuk informasi selengkapnya, lihat [sort\\_by](#) di situs web JMESPath.

### --query Contoh lanjutan

Untuk mengekstrak informasi dari item tertentu

Contoh berikut menggunakan --query parameter untuk menemukan item tertentu dalam daftar dan kemudian mengekstrak informasi dari item tersebut. Contoh ini mencantumkan semua yang AvailabilityZones terkait dengan titik akhir layanan yang ditentukan. Ini mengekstrak item dari ServiceDetails daftar yang telah ditentukan ServiceName, lalu mengeluarkan AvailabilityZones bidang dari item yang dipilih.

```
$ aws --region us-east-1 ec2 describe-vpc-endpoint-services \
  --query 'ServiceDetails[?ServiceName==`com.amazonaws.us-
  east-1.ecs`].AvailabilityZones'
[
  [
    "us-east-1a",
```

```
    "us-east-1b",
    "us-east-1c",
    "us-east-1d",
    "us-east-1e",
    "us-east-1f"
  ]
]
```

Untuk menampilkan snapshot setelah tanggal pembuatan yang ditentukan

Contoh berikut menunjukkan cara membuat daftar semua snapshot Anda yang dibuat setelah tanggal tertentu, termasuk hanya beberapa bidang yang tersedia dalam output.

```
$ aws ec2 describe-snapshots --owner self \
  --output json \
  --query 'Snapshots[?StartTime>=`2018-02-07`].
{Id:SnapshotId,VId:VolumeId,Size:VolumeSize}'
[
  {
    "id": "snap-0effb42b7a1b2c3d4",
    "vid": "vol-0be9bb0bf12345678",
    "Size": 8
  }
]
```

Untuk menunjukkan AMI terbaru

Contoh berikut mencantumkan lima Gambar Mesin Amazon (AMI) terbaru yang Anda buat, diurutkan dari yang terbaru ke yang paling lama.

```
$ aws ec2 describe-images \
  --owners self \
  --query 'reverse(sort_by(Images,&CreationDate))[:5].{id:ImageId,date:CreationDate}'
[
  {
    "id": "ami-0a1b2c3d4e5f60001",
    "date": "2018-11-28T17:16:38.000Z"
  },
  {
    "id": "ami-0a1b2c3d4e5f60002",
    "date": "2018-09-15T13:51:22.000Z"
  },
]
```

```
{
  "id": "ami-0a1b2c3d4e5f60003",
  "date": "2018-08-19T10:22:45.000Z"
},
{
  "id": "ami-0a1b2c3d4e5f60004",
  "date": "2018-05-03T12:04:02.000Z"
},
{
  "id": "ami-0a1b2c3d4e5f60005",
  "date": "2017-12-13T17:16:38.000Z"
}
]
```

Untuk menampilkan instans Auto Scaling yang tidak sehat

Contoh berikut hanya menampilkan InstanceId contoh yang tidak sehat dalam grup Auto Scaling yang ditentukan.

```
$ aws autoscaling describe-auto-scaling-groups \
  --auto-scaling-group-name My-AutoScaling-Group-Name \
  --output text \
  --query 'AutoScalingGroups[*].Instances[?HealthStatus==`Unhealthy`].InstanceId'
```

Untuk menyertakan volume dengan tag yang ditentukan

Contoh berikut menjelaskan semua contoh dengan test tag. Selama ada tag lain di samping yang test melekat pada volume, volume masih dikembalikan dalam hasil.

Ekspresi di bawah ini untuk mengembalikan semua test tag dengan tag dalam array. Setiap tag yang bukan test tag mengandung null nilai.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Tags[?Value == `test`]'
```

Untuk mengecualikan volume dengan tag yang ditentukan

Contoh berikut menjelaskan semua contoh tanpa test tag. Menggunakan ?Value != `test` ekspresi sederhana tidak berfungsi untuk mengecualikan volume karena volume dapat memiliki beberapa tag. Selama ada tag lain di samping yang test melekat pada volume, volume masih dikembalikan dalam hasil.

Untuk mengecualikan semua volume dengan test tag, mulailah dengan ekspresi di bawah ini untuk mengembalikan semua tag dengan test tag dalam array. Setiap tag yang bukan test tag mengandung null nilai.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Tags[?Value == `test`]'
```

Kemudian saring semua test hasil positif menggunakan not\_null fungsi tersebut.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[!not_null(Tags[?Value == `test`].Value)]'
```

Pipa hasil untuk meratakan hasil yang menghasilkan kueri berikut.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[!not_null(Tags[?Value == `test`].Value)] | []'
```

## Menggabungkan penyaringan sisi server dan sisi klien

Anda dapat menggunakan penyaringan sisi server dan sisi klien secara bersamaan. Pemfilteran sisi server diselesaikan terlebih dahulu, yang mengirimkan data ke klien yang kemudian difilter oleh parameter. --query Jika Anda menggunakan kumpulan data besar, menggunakan pemfilteran sisi server terlebih dahulu dapat menurunkan jumlah data yang dikirim ke klien untuk setiap AWS CLI panggilan, sambil tetap menjaga kustomisasi kuat yang disediakan oleh penyaringan sisi klien.

Contoh berikut mencantumkan volume Amazon EC2 menggunakan pemfilteran sisi server dan sisi klien. Layanan memfilter daftar semua volume terlampir di us-west-2a Availability Zone. --queryParameter selanjutnya membatasi output hanya volume dengan Size nilai yang lebih besar dari 50, dan hanya menampilkan bidang tertentu dengan nama yang ditentukan pengguna.

```
$ aws ec2 describe-volumes \
  --filters "Name=availability-zone,Values=us-west-2a" "Name=status,Values=attached" \
  --query 'Volumes[?Size > `50`].{Id:VolumeId,Size:Size,Type:VolumeType}'
[
  {
    "Id": "vol-0be9bb0bf12345678",
    "Size": 80,
    "VolumeType": "gp2"
  }
]
```

```
] ]
```

Contoh berikut mengambil daftar gambar yang memenuhi beberapa kriteria. Kemudian menggunakan `--query` parameter untuk mengurutkan output dengan `CreationDate`, memilih hanya yang terbaru. Akhirnya, ini menampilkan gambar `ImageId` yang satu itu.

```
$ aws ec2 describe-images \
  --owners amazon \
  --filters "Name=name,Values=amzn*gp2" "Name=virtualization-type,Values=hvm"
  "Name=root-device-type,Values=ebs" \
  --query "sort_by(Images, &CreationDate)[-1].ImageId" \
  --output text
ami-00ced3122871a4921
```

Contoh berikut menampilkan jumlah volume yang tersedia yang lebih dari 1000 IOPS dengan menggunakan `length` untuk menghitung berapa banyak dalam daftar.

```
$ aws ec2 describe-volumes \
  --filters "Name=status,Values=available" \
  --query 'length(Volumes[?Iops > `1000`])'
3
```

## Sumber daya tambahan

### AWS CLI autoprompt

Saat mulai menggunakan ekspresi filter, Anda dapat menggunakan fitur prompt otomatis di AWS CLI versi 2. Fitur auto-prompt menyediakan pratinjau ketika Anda menekan tombol F5. Untuk informasi selengkapnya, lihat [the section called “Prompt otomatis”](#).

### Terminal JMESPath

JMESPath Terminal adalah perintah terminal interaktif untuk bereksperimen dengan ekspresi JMESPath yang digunakan untuk pemfilteran sisi klien. Menggunakan `jpterm` perintah, terminal menampilkan hasil kueri langsung saat Anda mengetik. Anda dapat langsung AWS CLI menyalurkan output ke terminal, memungkinkan eksperimen kueri lanjutan.

Berikut contoh pipa `aws ec2 describe-volumes` output langsung ke JMESPath Terminal.

```
$ aws ec2 describe-volumes | jpterm
```



Untuk informasi selengkapnya tentang Terminal JMESPath dan petunjuk penginstalan, lihat Terminal [JMESPath](#) aktif. GitHub

## utilitas jq

jqUtilitas ini memberi Anda cara untuk mengubah output Anda di sisi klien ke format output yang Anda inginkan. Untuk informasi lebih lanjut tentang jq dan petunjuk pemasangan, lihat [jq](#) on GitHub.

## Mengembalikan kode dari AWS CLI

Kode kembali biasanya kode tersembunyi yang dikirim setelah menjalankan perintah AWS Command Line Interface (AWS CLI) yang menggambarkan status perintah. Anda dapat menggunakan echo perintah untuk menampilkan kode yang dikirim dari AWS CLI perintah terakhir dan menggunakan kode ini untuk menentukan apakah perintah berhasil atau gagal, dan mengapa perintah mungkin memiliki kesalahan. Selain kode pengembalian, Anda dapat melihat detail lebih lanjut tentang kegagalan dengan menjalankan perintah Anda dengan `--debug sakelar`. Sakelar ini menghasilkan laporan terperinci tentang langkah-langkah yang AWS CLI digunakan untuk memproses perintah, dan apa hasil dari setiap langkah itu.

Untuk menentukan kode pengembalian suatu AWS CLI perintah, jalankan salah satu perintah berikut segera setelah menjalankan perintah CLI.

### Linux and macOS

```
$ echo $?  
0
```

### Windows PowerShell

```
PS> echo $lastexitcode  
0
```

### Windows Command Prompt

```
C:\> echo %errorlevel%  
0
```

Berikut ini adalah nilai kode kembali yang dapat dikembalikan pada akhir menjalankan perintah AWS Command Line Interface (AWS CLI).

Code	Arti
0	Layanan merespons dengan kode status respons HTTP 200 yang menunjukkan bahwa tidak ada kesalahan yang dihasilkan oleh AWS CLI dan AWS layanan permintaan dikirim.
1	Satu atau lebih operasi transfer Amazon S3 gagal. Terbatas untuk perintah S3.
2	Arti dari kode pengembalian ini tergantung pada perintah: <ul style="list-style-type: none"> <li>Berlaku untuk semua AWS CLI perintah — perintah yang dimasukkan tidak dapat diuraikan. Kegagalan penguraian dapat disebabkan oleh, tetapi tidak terbatas pada, subperintah atau argumen yang diperlukan yang hilang, atau menggunakan perintah atau argumen yang tidak diketahui.</li> <li>Terbatas pada perintah S3 — Satu atau lebih file yang ditandai untuk transfer dilewati selama proses transfer. Namun, semua file lain yang ditandai untuk transfer berhasil ditransfer. File yang dilewati selama proses transfer meliputi: file yang tidak ada; file yang merupakan perangkat khusus karakter, blok perangkat khusus, antrian FIFO, atau soket; dan file yang pengguna tidak memiliki izin baca.</li> </ul>
130	Perintah itu terganggu oleh SIGINT. Ini adalah sinyal yang Anda kirim untuk membatalkan perintah dengan <code>Ctrl +C</code> .
252	Sintaks perintah tidak valid, parameter yang tidak diketahui disediakan, atau nilai parameter tidak benar dan mencegah perintah berjalan.
253	Lingkungan atau konfigurasi sistem tidak valid. Meskipun perintah yang diberikan mungkin valid secara sintaksis, konfigurasi atau kredensial yang hilang mencegah perintah berjalan.
254	Perintah berhasil diurai dan permintaan dibuat ke layanan yang ditentukan tetapi layanan mengembalikan kesalahan. Ini umumnya akan menunjukkan penggunaan API yang salah atau masalah khusus layanan lainnya.
255	Perintah gagal. Ada kesalahan yang dihasilkan oleh AWS CLI atau oleh AWS layanan tempat permintaan dikirim.

## Perintah interaktif menggunakan AWS CLI penyihir

The AWS Command Line Interface (AWS CLI) menyediakan kemampuan untuk menggunakan wizard untuk beberapa perintah. Untuk berkontribusi atau melihat daftar lengkap AWS CLI wizard yang tersedia, lihat [folder AWS CLI wizard](#) di GitHub

### Cara kerjanya

Mirip dengan AWS konsol, AWS CLI memiliki wizard UI yang memandu Anda mengelola AWS sumber daya Anda. Untuk menggunakan wizard, Anda memanggil `wizard` subperintah dan nama wizard setelah nama layanan dalam perintah. Struktur perintah adalah sebagai berikut:

Sintaksis:

```
$ aws <command> wizard <wizardName>
```

Contoh berikut adalah memanggil wizard untuk membuat dynamodb tabel baru.

```
$ aws dynamodb wizard new-table
```

`aws configure` adalah satu-satunya penyihir yang tidak memiliki nama penyihir. Saat menjalankan wizard, jalankan `aws configure wizard` perintah seperti yang ditunjukkan contoh berikut:

```
$ aws configure wizard
```

Setelah memanggil wizard, formulir di shell ditampilkan. Untuk setiap parameter, Anda diberikan daftar opsi untuk dipilih atau diminta untuk memasukkan string. Untuk memilih dari daftar, gunakan tombol panah atas dan bawah dan tekan ENTER. Untuk melihat detail pada opsi, tekan tombol panah kanan. Setelah selesai mengisi parameter, tekan ENTER.

```
$ aws configure wizard
What would you like to configure
> Static Credentials
  Assume Role
  Process Provider
  Additional CLI configuration
Enter the name of the profile:
Enter your Access Key Id:
```

```
Enter your Secret Access Key:
```

Untuk mengedit petunjuk sebelumnya, gunakan SHIFT + TAB. Untuk beberapa penyihir, setelah mengisi semua petunjuk, Anda dapat melihat pratinjau AWS CloudFormation templat atau AWS CLI perintah yang diisi dengan informasi Anda. Mode pratinjau ini berguna untuk mempelajari API layanan AWS CLI, dan membuat templat untuk skrip.

Tekan ENTER setelah melihat pratinjau atau prompt terakhir untuk menjalankan perintah terakhir.

```
$ aws configure wizard
What would you like to configure
Enter the name of the profile: testWizard
Enter your Access Key Id: AB1C2D3EF4GH5I678J90K
Enter your Secret Access Key: ab1c2def34gh5i67j8k90l1mnop2qr3s45tu678v90
<ENTER>
```

## Buat dan gunakan pintasan AWS CLI perintah yang disebut alias

Alias adalah pintasan yang dapat Anda buat di AWS Command Line Interface (AWS CLI) untuk mempersingkat perintah atau skrip yang sering Anda gunakan. Anda membuat alias dalam `alias` file yang terletak di folder konfigurasi Anda.

### Topik

- [Prasyarat](#)
- [Langkah 1: Membuat file alias](#)
- [Langkah 2: Membuat alias](#)
- [Langkah 3: Memanggil alias](#)
- [Contoh repositori alias](#)
- [Sumber daya](#)

## Prasyarat

Untuk menggunakan perintah alias, Anda harus menyelesaikan yang berikut ini:

- Instal dan konfigurasi AWS CLI. Lihat informasi yang lebih lengkap di [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).

- Gunakan AWS CLI versi minimum 1.11.24 atau 2.0.0.
- (Opsional) Untuk menggunakan skrip AWS CLI alias bash, Anda harus menggunakan terminal yang kompatibel dengan bash.

## Langkah 1: Membuat file alias

Untuk membuat alias file, Anda dapat menggunakan navigasi file dan editor teks, atau menggunakan terminal pilihan Anda dengan menggunakan step-by-step prosedur. Untuk membuat file alias Anda dengan cepat, gunakan blok perintah berikut.

Linux and macOS

```
$ mkdir -p ~/.aws/cli  
$ echo '[toplevel]' > ~/.aws/cli/alias
```

Windows

```
C:\> md %USERPROFILE%\aws\cli  
C:\> echo [toplevel] > %USERPROFILE%\aws\cli\alias
```

Untuk membuat file alias

1. Buat folder bernama `cli` di folder AWS CLI konfigurasi Anda. Secara default folder konfigurasi ada `~/.aws/` di Linux atau macOS dan `%USERPROFILE%\aws\` di Windows. Anda dapat membuat ini melalui navigasi file Anda atau dengan menggunakan perintah berikut.

Linux and macOS

```
$ mkdir -p ~/.aws/cli
```

Windows

```
C:\> md %USERPROFILE%\aws\cli
```

Jalur default `cli` folder yang dihasilkan ada `~/.aws/cli/` di Linux atau macOS dan `%USERPROFILE%\aws\cli` di Windows.

- Di `cli` folder, buat file teks bernama `alias` tanpa ekstensi dan tambahkan `[toplevel]` ke baris pertama. Anda dapat membuat file ini melalui editor teks pilihan Anda atau menggunakan perintah berikut.

Linux and macOS

```
$ echo '[toplevel]' > ~/.aws/cli/alias
```

Windows

```
C:\> echo [toplevel] > %USERPROFILE%\cli/alias
```

## Langkah 2: Membuat alias

Anda dapat membuat alias menggunakan perintah dasar atau bash scripting.

### Membuat alias perintah dasar

Anda dapat membuat alias Anda dengan menambahkan perintah menggunakan sintaks berikut dalam `alias` file yang Anda buat pada langkah sebelumnya.

Sintaks

```
aliasname = command [--options]
```

*Nama alias* adalah apa yang Anda sebut alias Anda. *Perintah* adalah perintah yang ingin Anda panggil, yang dapat menyertakan alias lainnya. Anda dapat menyertakan opsi atau parameter dalam alias Anda, atau menambahkannya saat memanggil alias Anda.

Contoh berikut membuat alias bernama `aws whoami` menggunakan perintah. Karena alias ini memanggil AWS CLI perintah yang ada, Anda dapat menulis perintah tanpa `aws` awalan.

```
whoami = sts get-caller-identity
```

Contoh berikut mengambil `whoami` contoh sebelumnya dan menambahkan output opsi `Account` filter dan teks.

```
whoami2 = sts get-caller-identity --query Account --output text
```

## Membuat alias sub-perintah

### Note

Fitur alias sub-perintah memerlukan AWS CLI versi minimum 1.11.24 atau 2.0.0

Anda dapat membuat alias untuk sub-perintah dengan menambahkan perintah menggunakan sintaks berikut dalam `alias` file yang Anda buat pada langkah sebelumnya.

### Sintaks

```
[command commandGroup]  
aliasname = command [--options]
```

*CommandGroup* adalah namespace perintah, misalnya Perintah `aws ec2 describe-regions` berada di bawah grup perintah. `ec2 Nama alias` adalah apa yang Anda sebut alias Anda. *Perintah* adalah perintah yang ingin Anda panggil, yang dapat menyertakan alias lainnya. Anda dapat menyertakan opsi atau parameter dalam alias Anda, atau menambahkannya saat memanggil alias Anda.

Contoh berikut membuat alias bernama `aws ec2 regions` menggunakan perintah. Karena alias ini memanggil AWS CLI perintah yang ada di bawah namespace `ec2` perintah, Anda dapat menulis perintah tanpa awalan. `aws ec2`

```
[command ec2]  
regions = describe-regions --query Regions[].RegionName
```

Untuk membuat alias dari perintah di luar namespace perintah,awali perintah lengkap dengan tanda seru. Contoh berikut membuat alias bernama `aws ec2 instance-profiles` menggunakan perintah.

```
[command ec2]  
instance-profiles = !aws iam list-instance-profiles
```

**Note**

Alias hanya menggunakan ruang nama perintah yang ada dan Anda tidak dapat membuat yang baru. misalnya Anda tidak dapat membuat alias dengan `[command johnsmith]` bagian tersebut karena namespace `johnsmith` perintah belum ada.

## Membuat alias skrip bash

**Warning**

Untuk menggunakan skrip AWS CLI alias bash, Anda harus menggunakan terminal yang kompatibel dengan bash

Anda dapat membuat alias menggunakan skrip bash untuk proses yang lebih maju menggunakan sintaks berikut.

### Sintaks

```
aliasname =  
    !f() {  
        script content  
    }; f
```

*Nama alias* adalah apa yang Anda sebut alias Anda dan *konten skrip* adalah skrip yang ingin Anda jalankan ketika Anda memanggil alias.

Contoh berikut digunakan `opendns` untuk menampilkan alamat IP Anda saat ini. Karena Anda dapat menggunakan alias di alias lain, alias berikut berguna untuk mengizinkan atau mencabut akses untuk alamat IP Anda dari dalam `myip` alias lain.

```
myip =  
    !f() {  
        dig +short myip.opendns.com @resolver1.opendns.com  
    }; f
```

Contoh skrip berikut memanggil `aws myip` alias sebelumnya untuk mengotorisasi alamat IP Anda untuk masuknya grup keamanan Amazon EC2.



```
authorize-my-ip =
  !f() {
    ip=$(aws myip)
    aws ec2 authorize-security-group-ingress --group-id ${1} --cidr $ip/32 --protocol
    tcp --port 22
  }; f
```

Saat Anda memanggil alias yang menggunakan skrip bash, variabel selalu diteruskan dalam urutan yang Anda masukkan. Dalam skrip bash, nama variabel tidak dipertimbangkan, hanya urutan kemunculannya. Dalam contoh `textalert` alias berikut, variabel untuk `--message` opsi adalah yang pertama dan `--phone-number` opsi kedua.

```
textalert =
  !f() {
    aws sns publish --message "${1}" --phone-number ${2}
  }; f
```

### Langkah 3: Memanggil alias

Untuk menjalankan alias yang Anda buat dalam `alias` file Anda gunakan sintaks berikut. Anda dapat menambahkan opsi tambahan saat memanggil alias Anda.

Sintaks

```
$ aws aliasname
```

Contoh berikut menggunakan alias `aws whoami` perintah.

```
$ aws
whoami
{
  "UserId": "A12BCD34E5FGHI6JKLM",
  "Account": "1234567890987",
  "Arn": "arn:aws:iam::1234567890987:user/userName"
}
```

Contoh berikut menggunakan `aws whoami` alias dengan opsi tambahan untuk hanya mengembalikan Account nomor dalam text output.

```
$ aws whoami --query Account --output
text
1234567890987
```

Contoh berikut menggunakan [alias aws ec2 regions sub-perintah](#).

```
$ aws ec2
regions
[
  "ap-south-1",
  "eu-north-1",
  "eu-west-3",
  "eu-west-2",
  ...
```

## Memanggil alias menggunakan variabel skrip bash

Saat Anda memanggil alias yang menggunakan skrip bash, variabel dilewatkan dalam urutan yang dimasukkan. Dalam bash scripting, nama variabel tidak dipertimbangkan, hanya urutan mereka muncul. Misalnya, dalam `textalert` alias berikut, variabel untuk opsi `--message` adalah yang pertama dan `--phone-number` kedua.

```
textalert =
!f() {
  aws sns publish --message "${1}" --phone-number ${2}
}; f
```

Saat Anda memanggil `textalert` alias, Anda harus meneruskan variabel dalam urutan yang sama seperti yang dijalankan di alias. Dalam contoh berikut kita menggunakan variabel `$message` dan `$phone`. `$message` variabel dilewatkan seperti `${1}` untuk `--message` opsi dan `$phone` variabel dilewatkan seperti `${2}` untuk `--phone-number` opsi. Ini menghasilkan berhasil memanggil `textalert` alias untuk mengirim pesan.

```
$ aws textalert $message
$phone
{
  "MessageId": "1ab2cd3e4-fg56-7h89-i01j-2klmn34567"
}
```

Dalam contoh berikut, urutan dialihkan saat memanggil alias ke `$phone` dan `$message`. Variabel `$phone` dilewatkan seperti `${1}` untuk `--message` opsi dan variabel `$message` dilewatkan seperti `${2}` untuk `--phone-number` opsi. Karena variabel rusak, alias melewati variabel secara tidak benar. Ini menyebabkan kesalahan karena konten `$message` tidak sesuai dengan persyaratan pemformatan nomor telepon untuk `--phone-number` opsi tersebut.

```
$ aws textalert $phone
$message
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

aws help
aws <command> help
aws <command> <subcommand> help

Unknown options: text
```

## Contoh repositori alias

[Repositori AWS CLI alias](#) di GitHub berisi contoh AWS CLI alias yang dibuat oleh tim AWS CLI pengembang dan komunitas. Anda dapat menggunakan seluruh contoh alias file atau mengambil alias individual untuk Anda gunakan sendiri.

### Warning

Menjalankan perintah di bagian ini akan menghapus alias file yang ada. Untuk menghindari penipaan file alias yang ada, ubah lokasi unduhan Anda.

Untuk menggunakan alias dari repositori

1. Instal Git. Untuk petunjuk penginstalan, lihat [Memulai - Menginstal Git](#) di Dokumentasi Git.
2. Instal `jq` perintah. `jq` Perintah ini digunakan dalam `tostring` alias. Untuk petunjuk penginstalan, lihat [JMESPath \(jq\) README.md](#) pada GitHub
3. Instal `jq` perintah. `jq` Perintah ini digunakan dalam `tostring-with-jq` alias. Untuk petunjuk penginstalan, lihat [prosesor JSON \(jq\) aktif](#). GitHub
4. Unduh alias file dengan melakukan salah satu hal berikut:

- Jalankan perintah berikut yang mengunduh dari repositori dan menyalin alias file ke folder konfigurasi Anda.

Linux and macOS

```
$ git clone https://github.com/awslabs/awscli-aliases.git
$ mkdir -p ~/.aws/cli
$ cp awscli-aliases/alias ~/.aws/cli/alias
```

Windows

```
C:\> git clone https://github.com/awslabs/awscli-aliases.git
C:\> md %USERPROFILE%\aws\cli
C:\> copy awscli-aliases\alias %USERPROFILE%\aws\cli
```

- Unduh langsung dari repositori dan simpan ke cli folder di folder AWS CLI konfigurasi Anda. Secara default folder konfigurasi ada ~/.aws/ di Linux atau macOS dan %USERPROFILE%\aws\ di Windows.

5. Untuk memverifikasi alias berfungsi, jalankan alias berikut.

```
$ aws whoami
```

Ini menampilkan respons yang sama dengan `aws sts get-caller-identity` perintah:

```
{
  "Account": "012345678901",
  "UserId": "AIUAINBADX2VEG2TC6HD6",
  "Arn": "arn:aws:iam::012345678901:user/myuser"
}
```

## Sumber daya

- [Repositori AWS CLI alias](#) pada GitHub berisi contoh AWS CLI alias yang dibuat oleh tim AWS CLI pengembang dan kontribusi komunitas. AWS CLI
- Pengumuman fitur alias dari [AWS re:Invent 2016: The Effective User on AWS CLI](#) YouTube
- [aws sts get-caller-identity](#)
- [aws ec2 describe-instances](#)

- [aws sns publish](#)

# Contoh kode

Bab ini menyediakan kumpulan contoh yang menunjukkan cara menggunakan AWS Command Line Interface (AWS CLI) dengan Layanan AWS.

Ini AWS CLI memiliki jenis contoh berikut dalam panduan ini:

- [Contoh perintah terpandu](#) - Contoh perintah terpandu untuk Panduan AWS CLI Pengguna tentang cara menggunakan AWS CLI dengan beberapa Layanan AWS. Ini seringkali merupakan contoh yang lebih rinci daripada contoh dari [referensi AWS CLI versi 2](#).
- [AWS CLI contoh perintah - Contoh](#) perintah sumber terbuka yang juga tersedia dalam [referensi AWS CLI versi 2](#). Contoh perintah di-host di [AWS CLI](#) repositori pada GitHub
- [AWS CLI menggunakan contoh kode skrip Bash - Contoh](#) skrip bash sumber terbuka. Contoh skrip Bash di-host di [Repositori Contoh AWS Kode](#) di GitHub

## Contoh umpan balik

Tidak dapat menemukan apa yang Anda butuhkan? Minta contoh perintah dengan menggunakan tautan Berikan umpan balik di bagian bawah halaman ini atau pada halaman perintah yang relevan dalam [versi 2 panduan referensi](#).

Ingin berkontribusi? Kontribusikan contoh AWS CLI perintah di [Repositori Contoh AWS Kode](#) di GitHub Untuk informasi selengkapnya tentang kontribusi, lihat [contoh AWS CLI kode langkah cepat kontribusi](#) pada GitHub halaman.

# Contoh AWS CLI perintah terpandu

Bagian ini memberikan contoh yang menunjukkan cara menggunakan AWS Command Line Interface (AWS CLI) untuk mengakses berbagai AWS layanan.

## Note

Untuk referensi lengkap dari semua perintah yang tersedia untuk setiap layanan, lihat [versi 2 panduan referensi](#), atau gunakan bantuan baris perintah bawaan. Untuk informasi selengkapnya, lihat [Dapatkan bantuan dengan AWS CLI](#).

## Layanan

- [Gunakan Amazon DynamoDB dengan AWS CLI](#)
- [Gunakan Amazon EC2 dengan AWS CLI](#)
- [Gunakan Amazon S3 Glacier dengan AWS CLI](#)
- [Gunakan AWS Identity and Access Management dari AWS CLI](#)
- [Gunakan Amazon S3 dengan AWS CLI](#)
- [Gunakan Amazon SNS dengan AWS CLI](#)

## Gunakan Amazon DynamoDB dengan AWS CLI

### Pengantar Amazon DynamoDB

#### [Apa itu Amazon DynamoDB?](#)

The AWS Command Line Interface (AWS CLI) menyediakan dukungan untuk semua layanan AWS database, termasuk Amazon DynamoDB. Anda dapat menggunakan AWS CLI untuk operasi dadakan, seperti membuat tabel. Anda juga dapat menggunakannya untuk menanamkan operasi DynamoDB dalam skrip utilitas.

Untuk informasi selengkapnya tentang menggunakan DynamoDB AWS CLI with, [\*dynamodb\*](#) lihat di AWS CLI Command Reference.

Untuk membuat daftar AWS CLI perintah untuk DynamoDB, gunakan perintah berikut.

```
$ aws dynamodb help
```

### Topik

- [Prasyarat](#)
- [Membuat dan menggunakan tabel DynamoDB](#)
- [Menggunakan DynamoDB Lokal](#)
- [Sumber daya](#)

## Prasyarat

Untuk menjalankan dynamodb perintah, Anda perlu:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, silakan lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).

## Membuat dan menggunakan tabel DynamoDB

Format baris perintah terdiri dari nama perintah DynamoDB, diikuti oleh parameter untuk perintah itu. AWS CLI mendukung [sintaks singkatan CLI untuk nilai parameter, dan](#) JSON penuh.

Contoh berikut membuat tabel bernama MusicCollection.

```
$ aws dynamodb create-table \  
  --table-name MusicCollection \  
  --attribute-definitions AttributeName=Artist,AttributeType=S  
  AttributeName=SongTitle,AttributeType=S \  
  --key-schema AttributeName=Artist,KeyType=HASH  
  AttributeName=SongTitle,KeyType=RANGE \  
  --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1
```

Anda dapat menambahkan baris baru ke tabel dengan perintah yang mirip dengan yang ditunjukkan pada contoh berikut. Contoh-contoh ini menggunakan kombinasi sintaks singkatan dan JSON.

```
$ aws dynamodb put-item \  
  --table-name MusicCollection \  
  --item '{  
    "Artist": {"S": "No One You Know"},  
    "SongTitle": {"S": "Call Me Today"} ,  
    "AlbumTitle": {"S": "Somewhat Famous"}  
  }' \  
  --return-consumed-capacity TOTAL  
{  
  "ConsumedCapacity": {  
    "CapacityUnits": 1.0,  
    "TableName": "MusicCollection"  
  }  
}
```

```
$ aws dynamodb put-item \  

```



```

--table-name MusicCollection \
--item '{
  "Artist": {"S": "Acme Band"},
  "SongTitle": {"S": "Happy Day"} ,
  "AlbumTitle": {"S": "Songs About Life"}
}' \
--return-consumed-capacity TOTAL
{
  "ConsumedCapacity": {
    "CapacityUnits": 1.0,
    "TableName": "MusicCollection"
  }
}

```

Mungkin sulit untuk membuat JSON yang valid dalam perintah satu baris. Untuk membuat ini lebih mudah, AWS CLI dapat membaca file JSON. Misalnya, pertimbangkan cuplikan JSON berikut, yang disimpan dalam file bernama `expression-attributes.json`

```

{
  ":v1": {"S": "No One You Know"},
  ":v2": {"S": "Call Me Today"}
}

```

Anda dapat menggunakan file itu untuk mengeluarkan query permintaan menggunakan file AWS CLI. Dalam contoh berikut, konten `expression-attributes.json` file digunakan sebagai nilai untuk `--expression-attribute-values` parameter.

```

$ aws dynamodb query --table-name MusicCollection \
  --key-condition-expression "Artist = :v1 AND SongTitle = :v2" \
  --expression-attribute-values file://expression-attributes.json
{
  "Count": 1,
  "Items": [
    {
      "AlbumTitle": {
        "S": "Somewhat Famous"
      },
      "SongTitle": {
        "S": "Call Me Today"
      },
      "Artist": {
        "S": "No One You Know"
      }
    }
  ]
}

```

```
    }  
  }  
],  
"ScannedCount": 1,  
"ConsumedCapacity": null  
}
```

## Menggunakan DynamoDB Lokal

Selain DynamoDB, Anda dapat menggunakan AWS CLI dengan DynamoDB Local. DynamoDB Local adalah database sisi klien kecil dan server yang meniru layanan DynamoDB. DynamoDB Local memungkinkan Anda untuk menulis aplikasi yang menggunakan DynamoDB API, tanpa memanipulasi tabel atau data apa pun dalam layanan web DynamoDB. Sebagai gantinya, semua tindakan API dialihkan ke database lokal. Ini memungkinkan Anda menghemat throughput yang disediakan, penyimpanan data, dan biaya transfer data.

Untuk informasi selengkapnya tentang DynamoDB Local dan cara menggunakannya dengan, lihat bagian berikut AWS CLI dari Panduan Pengembang Amazon [DynamoDB](#):

- [DynamoDB Lokal](#)
- [Menggunakan AWS CLI dengan DynamoDB Local](#)

## Sumber daya

AWS CLI referensi:

- [aws dynamodb](#)
- [aws dynamodb create-table](#)
- [aws dynamodb put-item](#)
- [aws dynamodb query](#)

Referensi layanan:

- [DynamoDB Lokal di Panduan Pengembang Amazon DynamoDB](#)
- [Menggunakan AWS CLI with DynamoDB Local di Panduan Pengembang Amazon DynamoDB](#)

# Gunakan Amazon EC2 dengan AWS CLI

## Pengantar Amazon Elastic Compute Cloud

### [Pengantar Amazon EC2 - Server Cloud Elastis dan Hosting dengan AWS](#)

Anda dapat mengakses fitur Amazon Elastic Compute Cloud (Amazon EC2) menggunakan (). AWS Command Line Interface AWS CLI Untuk membuat daftar AWS CLI perintah untuk Amazon EC2, gunakan perintah berikut.

```
aws ec2 help
```

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Topik ini menunjukkan contoh AWS CLI perintah singkat yang melakukan tugas umum untuk Amazon EC2.

Untuk contoh AWS CLI perintah bentuk panjang, lihat [repositori contoh AWS CLI kode](#) di GitHub

Topik

- [Membuat, menampilkan, dan menghapus pasangan kunci Amazon EC2](#)
- [Membuat, mengonfigurasi, dan menghapus grup keamanan untuk Amazon EC2](#)
- [Luncurkan, daftar, dan hentikan instans Amazon EC2](#)
- [Mengubah jenis instans Amazon EC2 dengan skrip bash](#)

## Membuat, menampilkan, dan menghapus pasangan kunci Amazon EC2

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk membuat, menampilkan, dan menghapus pasangan kunci untuk Amazon Elastic Compute Cloud (Amazon EC2). Anda menggunakan pasangan kunci untuk terhubung ke instans Amazon EC2.

Anda harus menyediakan key pair ke Amazon EC2 saat membuat instance, lalu gunakan key pair tersebut untuk mengautentikasi saat Anda terhubung ke instans.

**Note**

Untuk contoh perintah tambahan, lihat [referensi](#).

## Topik

- [Prasyarat](#)
- [Membuat pasangan kunci](#)
- [Menampilkan key pair](#)
- [Menghapus pasangan kunci Anda](#)
- [Referensi](#)

## Prasyarat

Untuk menjalankan ec2 perintah, Anda perlu:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Setel izin IAM Anda untuk memungkinkan akses Amazon EC2. Untuk informasi selengkapnya tentang izin IAM untuk Amazon EC2, [lihat kebijakan IAM untuk Amazon EC2 di Panduan Pengguna Amazon EC2](#).

## Membuat pasangan kunci

Untuk membuat key pair, gunakan [aws ec2 create-key-pair](#) perintah dengan `--query` opsi, dan `--output text` opsi untuk menyalurkan kunci pribadi Anda langsung ke file.

```
$ aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text  
> MyKeyPair.pem
```

Untuk PowerShell, > file pengalihan default ke pengkodean UTF-8, yang tidak dapat digunakan dengan beberapa klien SSH. Jadi, Anda harus mengonversi output dengan menyalurkannya ke `out-file` perintah dan secara eksplisit mengatur pengkodean ke `ascii`

```
PS C:\>aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text  
| out-file -encoding ascii -filepath MyKeyPair.pem
```

`MyKeyPair.pem` yang dihasilkan terlihat mirip dengan yang berikut ini.

```
-----BEGIN RSA PRIVATE KEY-----
EXAMPLEKEYKCAQEAY7WZhaDsrA1W3mRlQtvhwyORRX8gnxgDAfRt/gx42kWXsT4rXE/b5CpSgie/
vBoU7jLxx92pNHoFnByP+Dc21eyyz6CvjTmWA0JwfWiW5/akH7i05dSrvC7dQkW2duV5QuUdE0QW
Z/aNxMniGQE6XAgfwlnXVBwrerrQo+ZWQeqiUwwMkuEbLeJFLhMCvYURpUMSC1oehm449ilx9X1F
G50TCFe0zf18dqqCP6GzbPaIjiU19xX/az0R9V+tpU0zEL+wmXnZt3/nHPQ5xvD20JH67km6SuPW
oPzev/D8V+x4+bHthfSjR9Y7DvQFjfbVwHXigBdtZcU2/wei8D/HYwIDAQABAoIBAGZ1kaEvntrqu
/uler7vgIn5m71N5LKw4hJLAIW6tUT/fzvtcCHK0SbkQCQXuriHmQ2MQyJX/0kn2NfjLV/ufGxbl1
mb5qwMGUnEpJaZD6QSSs3kICLwUUYUiGfc0uisbmJoap/GTLU0W5Mfcv36PaBUNy5p53V6G7hXb2
bahyWyJNfjLe4M86yd2YK3V2CmK+X/B0sShnJ36+hjrXPPWmV3N9zEmCdJjA+K15DYmhm/tJWSD9
81oGk9TopEp7CkIfatEATyyZiVqoRq6k64iuM9JkA30zdXzMqexXVJ1TLZVEH0E7bh1Y9d801ozR
oQs/FiZNAx2iijCWyv01pjE73+kCgYEA9mZtyhkHkFDpwrSM1APaL8oNAbbjwEy7Z5Mqfq1+lIp1
YkriL0DbLXlvRAH+yHPRit2hH0jtUNZ4Aaxv+cpg09qbUI3+43eEy24B7G/Uh+GTfbjsXs0xQx/x
p9otyVvc7hsQ5TA5PZb+mvkJ50BEKzet9XcKw0NBVELGhnEPe7cCgYEA06Vgov6YH1eHui9kHuws
ayav0elc5zkxjF9nfHFJRry21R1trw2Vdpn+9g481URrpzWV0Eihvm+xTtmaZlSp//lkq75XDwnU
WA8gkn603QE3fq2yN98BURsAKdJfJ5RL1HvGQvTe10HLYYXpJnEkHv+Un12ajLivWUt5pbBrKbUC
gYBjb0+0Zk0sCcpZ29sbzjYjpIddErySIyRX5gV2uNQwAjLdp9PfN295yQ+BxMBXiIycWVQiw0bH
oMo7yykABY70zd5wQewBQ4AdSlWSX4nGDtsiFxiI5sKuAAe0CbTosy1s8w8fxoJ5Tz1sdoxNeGs
Arq6Wv/G16zQuAE9zK9vVwKBgF+09VI/1wJBirsDGz9whVwFFPrTkJNvJZzYt69qezx1sjgFKshy
WBhd4xHZtmCqpBP1AymEjr/T01bxyARMXmNIOWIANXMGb4KGSy11mzSVAoQ+fqR+cJ3d0dyP11j
jjb0Ed/NY8fr1NDxAVHE8BSkdsx2f6ELEyBKJSRr9snRAoGAMrTwYneXzvTskF/S5Fyu0i0egLda
NWUH38v/nDCgEpIXD5Hn3qAEcju1Ijmbw1vtW+nY2jVhv7UGd8MjwUTNGItdb6nsYqM2asrnF3qS
VRkAKKKYeGjKpUfVTrW0YFjXkfcR/V+QFL50ndHAKJXjW7a4ejJLncTzmZSpYzwApc=
-----END RSA PRIVATE KEY-----
```

Kunci pribadi Anda tidak disimpan AWS dan hanya dapat diambil ketika dibuat. Anda tidak dapat memulihkannya nanti. Sebaliknya, jika Anda kehilangan kunci pribadi, Anda harus membuat key pair baru.

Jika Anda terhubung ke instans Anda dari komputer Linux, kami sarankan Anda menggunakan perintah berikut untuk mengatur izin file kunci pribadi Anda sehingga hanya Anda yang dapat membacanya.

```
$ chmod 400 MyKeyPair.pem
```

Menampilkan key pair

“Sidik jari” dihasilkan dari key pair Anda, dan Anda dapat menggunakannya untuk memverifikasi bahwa kunci pribadi yang Anda miliki di mesin lokal Anda cocok dengan kunci publik yang disimpan di dalamnya AWS.

Sidik jari adalah hash SHA1 yang diambil dari salinan kunci pribadi yang dikodekan DER. Nilai ini ditangkap ketika key pair dibuat, dan disimpan AWS dengan kunci publik. Anda dapat melihat sidik jari di konsol Amazon EC2 atau dengan menjalankan perintah. AWS CLI [aws ec2 describe-key-pairs](#)

Contoh berikut menampilkan sidik jari untuk `MyKeyPair`.

```
$ aws ec2 describe-key-pairs --key-name MyKeyPair
{
  "KeyPairs": [
    {
      "KeyName": "MyKeyPair",
      "KeyFingerprint":
        "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f"
    }
  ]
}
```

Untuk informasi selengkapnya tentang kunci dan sidik jari, lihat Pasangan [Kunci Amazon EC2](#) di Panduan Pengguna Amazon EC2.

Menghapus pasangan kunci Anda

Untuk menghapus key pair, jalankan [aws ec2 delete-key-pair](#) perintah, ganti `MyKeyPair` dengan nama pasangan yang akan dihapus.

```
$ aws ec2 delete-key-pair --key-name MyKeyPair
```

Referensi

AWS CLI Referensi:

- [aws ec2](#)
- [aws ec2 create-key-pair](#)
- [aws ec2 delete-key-pair](#)
- [aws ec2 describe-key-pairs](#)

Referensi lainnya:

- [Dokumentasi Cloud Komputasi Elastis Amazon](#)

- Untuk melihat dan berkontribusi pada AWS SDK dan contoh AWS CLI kode, lihat [Repositori Contoh AWS Kode](#) di GitHub

## Membuat, mengonfigurasi, dan menghapus grup keamanan untuk Amazon EC2

Anda dapat membuat grup keamanan untuk instans Amazon Elastic Compute Cloud (Amazon EC2) yang pada dasarnya beroperasi sebagai firewall, dengan aturan yang menentukan lalu lintas jaringan apa yang dapat masuk dan keluar.

Gunakan AWS Command Line Interface (AWS CLI) untuk membuat grup keamanan, menambahkan aturan ke grup keamanan yang ada, dan menghapus grup keamanan.

### Note

Untuk contoh perintah tambahan, lihat [referensi](#).

### Topik

- [Prasyarat](#)
- [Membuat grup keamanan](#)
- [Tambahkan aturan ke grup keamanan Anda](#)
- [Hapus grup keamanan Anda](#)
- [Referensi](#)

### Prasyarat

Untuk menjalankan ec2 perintah, Anda perlu:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Setel izin IAM Anda untuk memungkinkan akses Amazon EC2. Untuk informasi selengkapnya tentang izin IAM untuk Amazon EC2, [lihat kebijakan IAM untuk Amazon EC2 di Panduan Pengguna Amazon EC2](#).

### Membuat grup keamanan

Anda dapat membuat grup keamanan yang terkait dengan virtual private cloud (VPC).

[aws ec2 create-security-group](#) Contoh berikut menunjukkan cara membuat grup keamanan untuk VPC tertentu.

```
$ aws ec2 create-security-group --group-name my-sg --description "My security group" --  
vpc-id vpc-1a2b3c4d  
{  
  "GroupId": "sg-903004f8"  
}
```

Untuk melihat informasi awal untuk grup keamanan, jalankan [aws ec2 describe-security-groups](#) perintah. Anda dapat mereferensikan grup keamanan EC2-VPC hanya dengan `namanyavpc-id`, bukan namanya.

```
$ aws ec2 describe-security-groups --group-ids sg-903004f8  
{  
  "SecurityGroups": [  
    {  
      "IpPermissionsEgress": [  
        {  
          "IpProtocol": "-1",  
          "IpRanges": [  
            {  
              "CidrIp": "0.0.0.0/0"  
            }  
          ],  
          "UserIdGroupPairs": []  
        }  
      ],  
      "Description": "My security group"  
      "IpPermissions": [],  
      "GroupName": "my-sg",  
      "VpcId": "vpc-1a2b3c4d",  
      "OwnerId": "123456789012",  
      "GroupId": "sg-903004f8"  
    }  
  ]  
}
```

Tambahkan aturan ke grup keamanan Anda

Saat menjalankan instans Amazon EC2, Anda harus mengaktifkan aturan di grup keamanan untuk mengizinkan lalu lintas jaringan masuk untuk sarana Anda menghubungkan ke gambar.



Misalnya, jika Anda meluncurkan instance Windows, Anda biasanya menambahkan aturan untuk mengizinkan lalu lintas masuk pada port TCP 3389 untuk mendukung Remote Desktop Protocol (RDP). Jika Anda meluncurkan instance Linux, Anda biasanya menambahkan aturan untuk mengizinkan lalu lintas masuk pada port TCP 22 untuk mendukung koneksi SSH.

Gunakan [aws ec2 authorize-security-group-ingress](#) perintah untuk menambahkan aturan ke grup keamanan Anda. Parameter yang diperlukan dari perintah ini adalah alamat IP publik komputer Anda, atau jaringan (dalam bentuk rentang alamat) yang dilampirkan komputer Anda, dalam notasi [CIDR](#).

### Note

Kami menyediakan layanan berikut, <https://checkip.amazonaws.com/>, untuk memungkinkan Anda menentukan alamat IP publik Anda. Untuk menemukan layanan lain yang dapat membantu Anda mengidentifikasi alamat IP Anda, gunakan browser Anda untuk mencari "apa alamat IP saya". Jika Anda terhubung melalui ISP atau dari belakang firewall Anda menggunakan alamat IP dinamis (melalui gateway NAT dari jaringan pribadi), alamat Anda dapat berubah secara berkala. Dalam hal ini, Anda harus mengetahui kisaran alamat IP yang digunakan oleh komputer klien.

Contoh berikut menunjukkan cara menambahkan aturan untuk RDP (port TCP 3389) ke grup keamanan EC2-VPC dengan ID menggunakan alamat IP Anda. `sg-903004f8`

Untuk memulai, temukan alamat IP Anda.

```
$ curl https://checkip.amazonaws.com  
x.x.x.x
```

Anda kemudian dapat menambahkan alamat IP ke grup keamanan Anda dengan menjalankan [aws ec2 authorize-security-group-ingress](#) perintah.

```
$ aws ec2 authorize-security-group-ingress --group-id sg-903004f8 --protocol tcp --port  
3389 --cidr x.x.x.x/x
```

Perintah berikut menambahkan aturan lain untuk mengaktifkan SSH ke instance dalam grup keamanan yang sama.

```
$ aws ec2 authorize-security-group-ingress --group-id sg-903004f8 --protocol tcp --port 22 --cidr x.x.x.x/x
```

Untuk melihat perubahan pada grup keamanan, jalankan [aws ec2 describe-security-groups](#) perintah.

```
$ aws ec2 describe-security-groups --group-ids sg-903004f8
{
  "SecurityGroups": [
    {
      "IpPermissionsEgress": [
        {
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0"
            }
          ],
          "UserIdGroupPairs": []
        }
      ],
      "Description": "My security group"
      "IpPermissions": [
        {
          "ToPort": 22,
          "IpProtocol": "tcp",
          "IpRanges": [
            {
              "CidrIp": "x.x.x.x/x"
            }
          ],
          "UserIdGroupPairs": [],
          "FromPort": 22
        }
      ],
      "GroupName": "my-sg",
      "OwnerId": "123456789012",
      "GroupId": "sg-903004f8"
    }
  ]
}
```

## Hapus grup keamanan Anda

Untuk menghapus grup keamanan, jalankan [aws ec2 delete-security-group](#) perintah.

### Note

Anda tidak dapat menghapus grup keamanan jika saat ini dilampirkan ke lingkungan.

Contoh perintah berikut menghapus grup keamanan EC2-VPC.

```
$ aws ec2 delete-security-group --group-id sg-903004f8
```

## Referensi

AWS CLI referensi:

- [aws ec2](#)
- [aws ec2 authorize-security-group-ingress](#)
- [aws ec2 create-security-group](#)
- [aws ec2 delete-security-group](#)
- [aws ec2 describe-security-groups](#)

Referensi lainnya:

- [Dokumentasi Cloud Komputasi Elastis Amazon](#)
- Untuk melihat dan berkontribusi pada AWS SDK dan contoh AWS CLI kode, lihat [Repositori Contoh AWS Kode](#) di GitHub

## Luncurkan, daftar, dan hentikan instans Amazon EC2

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk meluncurkan, mencantumkan, dan menghentikan instans Amazon Elastic Compute Cloud (Amazon EC2). Jika meluncurkan instans yang tidak berada dalam Tingkat AWS Gratis, Anda akan ditagih setelah meluncurkan instans dan dikenakan biaya untuk waktu instans berjalan, meskipun instans tersebut tetap menganggur.

 Note

Untuk contoh perintah tambahan, lihat [referensi](#).

## Topik

- [Prasyarat](#)
- [Luncurkan instans Anda](#)
- [Menambahkan perangkat blokir ke instans Anda](#)
- [Tambahkan tag ke instance Anda](#)
- [Terhubung ke instans Anda.](#)
- [Buat daftar instans Anda](#)
- [Akhiri instans Anda](#)
- [Referensi](#)

## Prasyarat

Untuk menjalankan ec2 perintah dalam topik ini, Anda perlu:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Setel izin IAM Anda untuk memungkinkan akses Amazon EC2. Untuk informasi selengkapnya tentang izin IAM untuk Amazon EC2, [lihat kebijakan IAM untuk Amazon EC2 di Panduan Pengguna Amazon EC2](#).
- Buat [key pair](#) dan [grup keamanan](#).
- Pilih Amazon Machine Image (AMI) dan catat ID AMI. Untuk informasi selengkapnya, lihat [Menemukan AMI yang Sesuai](#) di Panduan Pengguna Amazon EC2.

## Luncurkan instans Anda

Untuk meluncurkan instans Amazon EC2 menggunakan AMI yang Anda pilih, gunakan perintah. [aws ec2 run-instances](#) Anda dapat meluncurkan instance ke virtual private cloud (VPC).

Awalnya, instance Anda muncul di pending negara bagian, tetapi berubah ke running status setelah beberapa menit.

Contoh berikut menunjukkan cara meluncurkan t2.micro instance di subnet yang ditentukan dari VPC. Ganti nilai parameter yang *dicetak miring* dengan milik Anda sendiri.

```
$ aws ec2 run-instances --image-id ami-xxxxxxx --count 1 --instance-type t2.micro --
key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e
{
  "OwnerId": "123456789012",
  "ReservationId": "r-5875ca20",
  "Groups": [
    {
      "GroupName": "my-sg",
      "GroupId": "sg-903004f8"
    }
  ],
  "Instances": [
    {
      "Monitoring": {
        "State": "disabled"
      },
      "PublicDnsName": null,
      "Platform": "windows",
      "State": {
        "Code": 0,
        "Name": "pending"
      },
      "EbsOptimized": false,
      "LaunchTime": "2013-07-19T02:42:39.000Z",
      "PrivateIpAddress": "10.0.1.114",
      "ProductCodes": [],
      "VpcId": "vpc-1a2b3c4d",
      "InstanceId": "i-5203422c",
      "ImageId": "ami-173d747e",
      "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
      "KeyName": "MyKeyPair",
      "SecurityGroups": [
        {
          "GroupName": "my-sg",
          "GroupId": "sg-903004f8"
        }
      ],
      "ClientToken": null,
      "SubnetId": "subnet-6e7f829e",
      "InstanceType": "t2.micro",
```

```
"NetworkInterfaces": [
  {
    "Status": "in-use",
    "SourceDestCheck": true,
    "VpcId": "vpc-1a2b3c4d",
    "Description": "Primary network interface",
    "NetworkInterfaceId": "eni-a7edb1c9",
    "PrivateIpAddresses": [
      {
        "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
        "Primary": true,
        "PrivateIpAddress": "10.0.1.114"
      }
    ],
    "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
    "Attachment": {
      "Status": "attached",
      "DeviceIndex": 0,
      "DeleteOnTermination": true,
      "AttachmentId": "eni-attach-52193138",
      "AttachTime": "2013-07-19T02:42:39.000Z"
    },
    "Groups": [
      {
        "GroupName": "my-sg",
        "GroupId": "sg-903004f8"
      }
    ],
    "SubnetId": "subnet-6e7f829e",
    "OwnerId": "123456789012",
    "PrivateIpAddress": "10.0.1.114"
  }
],
"SourceDestCheck": true,
"Placement": {
  "Tenancy": "default",
  "GroupName": null,
  "AvailabilityZone": "us-west-2b"
},
"Hypervisor": "xen",
"BlockDeviceMappings": [
  {
    "DeviceName": "/dev/sda1",
    "Ebs": {
```

```

        "Status": "attached",
        "DeleteOnTermination": true,
        "VolumeId": "vol-877166c8",
        "AttachTime": "2013-07-19T02:42:39.000Z"
      }
    ],
    "Architecture": "x86_64",
    "StateReason": {
      "Message": "pending",
      "Code": "pending"
    },
    "RootDeviceName": "/dev/sda1",
    "VirtualizationType": "hvm",
    "RootDeviceType": "ebs",
    "Tags": [
      {
        "Value": "MyInstance",
        "Key": "Name"
      }
    ],
    "AmiLaunchIndex": 0
  }
]
}

```

## Menambahkan perangkat blok ke instans Anda

Setiap instance yang Anda luncurkan memiliki volume perangkat root terkait. Anda dapat menggunakan pemetaan perangkat blok untuk menentukan volume Amazon Elastic Block Store (Amazon EBS) tambahan atau volume penyimpanan instans untuk dilampirkan ke instance saat diluncurkan.

Untuk menambahkan perangkat blok ke instans Anda, tentukan `--block-device-mappings` opsi saat Anda menggunakan `run-instances`.

Parameter contoh berikut menyediakan volume Amazon EBS standar yang berukuran 20 GB, dan memetakannya ke instans Anda menggunakan pengenal `/dev/sdf`.

```

--block-device-mappings "[{\"DeviceName\":\"/dev/sdf\",\"Ebs\":{\"VolumeSize\":20,
\"DeleteOnTermination\":false}}]"

```

Contoh berikut menambahkan volume Amazon EBS, dipetakan ke `/dev/sdf`, berdasarkan snapshot yang ada. Snapshot mewakili gambar yang dimuat ke volume untuk Anda. Saat Anda menentukan snapshot, Anda tidak perlu menentukan ukuran volume; itu akan cukup besar untuk menampung gambar Anda. Namun, jika Anda menentukan ukuran, itu harus lebih besar dari atau sama dengan ukuran snapshot.

```
--block-device-mappings "[{"DeviceName":"/dev/sdf","Ebs":{"SnapshotId":"snap-a1b2c3d4"}}]"
```

Contoh berikut menambahkan dua volume ke instance Anda. Jumlah volume yang tersedia untuk instans Anda bergantung pada jenis instance-nya.

```
--block-device-mappings [{"DeviceName":"/dev/sdf","VirtualName":"ephemeral0"}, {"DeviceName":"/dev/sdg","VirtualName":"ephemeral1"}]
```

Contoh berikut membuat pemetaan (`/dev/sdj`), tetapi tidak menyediakan volume untuk instance.

```
--block-device-mappings [{"DeviceName":"/dev/sdj","NoDevice":""}]"
```

Untuk informasi selengkapnya, lihat [Memblokir Pemetaan Perangkat](#) di Panduan Pengguna Amazon EC2.

Tambahkan tag ke instance Anda

Tag adalah label yang Anda tetapkan ke AWS sumber daya. Ini memungkinkan Anda untuk menambahkan metadata ke sumber daya Anda yang dapat Anda gunakan untuk berbagai tujuan. Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Anda](#) di Panduan Pengguna Amazon EC2.

Contoh berikut menunjukkan bagaimana menambahkan tag dengan nama kunci "Name" dan nilai "MyInstance" untuk contoh yang ditentukan, dengan menggunakan [aws ec2 create-tags](#) perintah.

```
$ aws ec2 create-tags --resources i-5203422c --tags Key=Name,Value=MyInstance
```

Terhubung ke instans Anda.

Ketika instans Anda berjalan, Anda dapat terhubung ke sana dan menggunakannya sama seperti Anda akan menggunakan komputer yang duduk di depan Anda. Untuk informasi selengkapnya, lihat [Connect ke Instans Amazon EC2 Anda](#) di Panduan Pengguna Amazon EC2.



## Buat daftar instans Anda

Anda dapat menggunakan AWS CLI untuk daftar instance Anda dan melihat informasi tentang mereka. Anda dapat membuat daftar semua instance Anda, atau memfilter hasil berdasarkan instance yang Anda minati.

Contoh berikut menunjukkan cara menggunakan [aws ec2 describe-instances](#) perintah.

Perintah berikut mencantumkan semua instance Anda.

```
$ aws ec2 describe-instances
```

Perintah berikut memfilter daftar ke hanya t2.micro instance Anda dan hanya mengeluarkan InstanceId nilai untuk setiap kecocokan.

```
$ aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query
"Reservations[].Instances[].InstanceId"
[
  "i-05e998023d9c69f9a"
]
```

Perintah berikut mencantumkan salah satu instance Anda yang memiliki tagName=MyInstance.

```
$ aws ec2 describe-instances --filters "Name=tag:Name,Values=MyInstance"
```

Perintah berikut mencantumkan instance Anda yang diluncurkan menggunakan salah satu dari AMI berikut:ami-x0123456,ami-y0123456, danami-z0123456.

```
$ aws ec2 describe-instances --filters "Name=image-id,Values=ami-x0123456,ami-
y0123456,ami-z0123456"
```

## Akhiri instans Anda

Mengakhiri sebuah instance menghapusnya. Anda tidak dapat menyambung kembali ke sebuah instance setelah Anda menghentikannya.

Segera setelah status instans berubah menjadi shutting-down atauterminated, Anda berhenti menimbulkan biaya untuk contoh itu. Jika Anda ingin menyambung kembali ke instance nanti,

gunakan [stop-instance](#) sebagai gantinya. `terminate-instances` Untuk informasi selengkapnya, lihat [Menghentikan Instans Anda](#) di Panduan Pengguna Amazon EC2.

Untuk menghapus instance, Anda menggunakan perintah [aws ec2 terminate-instances](#) untuk menghapusnya.

```
$ aws ec2 terminate-instances --instance-ids i-5203422c
{
  "TerminatingInstances": [
    {
      "InstanceId": "i-5203422c",
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}
```

## Referensi

### AWS CLI Referensi:

- [aws ec2](#)
- [aws ec2 create-tags](#)
- [aws ec2 describe-instances](#)
- [aws ec2 run-instances](#)
- [aws ec2 terminate-instances](#)

### Referensi lainnya:

- [Dokumentasi Cloud Komputasi Elastis Amazon](#)
- Untuk melihat dan berkontribusi pada AWS SDK dan contoh AWS CLI kode, lihat [Repositori Contoh AWS Kode](#) di GitHub

## Mengubah jenis instans Amazon EC2 dengan skrip bash

Contoh skrip bash untuk Amazon EC2 ini mengubah jenis instans untuk instans Amazon EC2 menggunakan (). AWS Command Line Interface AWS CLI Ini menghentikan instance jika sedang berjalan, mengubah jenis instance, dan kemudian, jika diminta, memulai ulang instance. Skrip Shell adalah program yang dirancang untuk berjalan di antarmuka baris perintah.

### Note

Untuk contoh perintah tambahan, lihat [referensi](#).

### Topik

- [Sebelum Anda mulai](#)
- [Tentang contoh ini](#)
- [Parameter](#)
- [Berkas](#)
- [Referensi](#)

### Sebelum Anda mulai

Sebelum Anda dapat menjalankan salah satu contoh di bawah ini, hal-hal berikut perlu diselesaikan.

- Instal dan konfigurasi AWS CLI. Lihat informasi yang lebih lengkap di [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Profil yang Anda gunakan harus memiliki izin yang memungkinkan AWS operasi dilakukan oleh contoh.
- Instans Amazon EC2 yang berjalan di akun yang Anda memiliki izin untuk berhenti dan memodifikasinya. Jika Anda menjalankan skrip pengujian, ia meluncurkan instance untuk Anda, menguji mengubah jenisnya, dan kemudian mengakhiri instance.
- Sebagai praktik AWS terbaik, berikan kode ini hak istimewa paling sedikit, atau hanya izin yang diperlukan untuk melakukan tugas. Untuk informasi selengkapnya, lihat [Memberikan Hak Istimewa Paling Sedikit](#) di Panduan Pengguna AWS Identity and Access Management (IAM).
- Kode ini belum diuji di semua AWS Wilayah. Beberapa AWS layanan hanya tersedia di Wilayah tertentu. Untuk informasi selengkapnya, lihat [Titik Akhir Layanan dan Kuota di Panduan Referensi AWS Umum](#).

- Menjalankan kode ini dapat mengakibatkan biaya ke AWS akun Anda. Adalah tanggung jawab Anda untuk memastikan bahwa sumber daya apa pun yang dibuat oleh skrip ini dihapus ketika Anda selesai dengan mereka.

Tentang contoh ini

Contoh ini ditulis sebagai fungsi dalam file skrip shell `change_ec2_instance_type.sh` yang Anda dapat `source` dari skrip lain atau dari baris perintah. Setiap file skrip berisi komentar yang menjelaskan masing-masing fungsi. Setelah fungsi dalam memori, Anda dapat memanggilnya dari baris perintah. Misalnya, perintah berikut mengubah jenis instance yang ditentukan menjadi `t2.nano`:

```
$ source ./change_ec2_instance_type.sh
$ ./change_ec2_instance_type -i *instance-id* -t new-type
```

Untuk contoh lengkap dan file skrip yang dapat diunduh, lihat [Mengubah Jenis Instans Amazon EC2](#) di Repositori AWS Contoh Kode. GitHub

Parameter

- i - (string) Menentukan ID contoh untuk memodifikasi.
- t - (string) Menentukan jenis instans Amazon EC2 untuk beralih ke.
- r - (switch) Secara default, ini tidak disetel. Jika `-r` disetel, restart instance setelah sakelar tipe.
- f - (switch) Secara default, skrip meminta pengguna untuk mengonfirmasi mematikan instance sebelum beralih. Jika `-f` disetel, fungsi tidak meminta pengguna sebelum mematikan instance untuk membuat sakelar tipe
- v - (switch) Secara default, skrip beroperasi secara diam-diam dan menampilkan output hanya jika terjadi kesalahan. Jika `-v` diatur, fungsi menampilkan status di seluruh operasinya.

Berkas

### **change\_ec2\_instance\_type.sh**

File skrip utama berisi `change_ec2_instance_type()` fungsi yang melakukan tugas-tugas berikut:

- Memverifikasi bahwa instans Amazon EC2 yang ditentukan ada.
- Kecuali `-f` dipilih, memperingatkan pengguna sebelum menghentikan instance.

- Mengubah jenis instance
- Jika Anda menyetel `-r`, restart instance dan mengonfirmasi bahwa instance sedang berjalan

Lihat kode untuk [change\\_ec2\\_instance\\_type.sh](#) aktif GitHub.

## **test\_change\_ec2\_instance\_type.sh**

`test_change_ec2_instance_type.sh` Skrip file menguji berbagai jalur kode untuk `change_ec2_instance_type` fungsi tersebut. Jika semua langkah dalam skrip pengujian berfungsi dengan benar, skrip pengujian menghapus semua sumber daya yang dibuatnya.

Anda dapat menjalankan skrip pengujian dengan parameter berikut:

- `-v` - (switch) Setiap tes menunjukkan status lulus/kegagalan saat dijalankan. Secara default, tes berjalan diam-diam dan output hanya mencakup status lulus/kegagalan keseluruhan akhir.
- `-i` - (switch) Script berhenti setelah setiap tes untuk memungkinkan Anda menelusuri hasil antara setiap langkah. Memungkinkan Anda memeriksa status instans saat ini menggunakan konsol Amazon EC2. Script melanjutkan ke langkah berikutnya setelah Anda menekan ENTER pada prompt.

Lihat kode untuk [test\\_change\\_ec2\\_instance\\_type.sh](#) aktif GitHub.

## **awsdocs\_general.sh**

File skrip `awsdocs_general.sh` menyimpan fungsi tujuan umum yang digunakan di seluruh contoh lanjutan untuk file AWS CLI.

Lihat kode untuk [awsdocs\\_general.sh](#) aktif GitHub.

## Referensi

AWS CLI referensi:

- [aws ec2](#)
- [aws ec2 describe-instances](#)
- [aws ec2 modify-instance-attribute](#)
- [aws ec2 start-instances](#)
- [aws ec2 stop-instances](#)

- [aws ec2 wait instance-running](#)
- [aws ec2 wait instance-stopped](#)

Referensi lainnya:

- [Dokumentasi Cloud Komputasi Elastis Amazon](#)
- Untuk melihat dan berkontribusi pada AWS SDK dan contoh AWS CLI kode, lihat [Repositori Contoh AWS Kode](#) di GitHub

## Gunakan Amazon S3 Glacier dengan AWS CLI

Pengantar Amazon S3 Glacier

[Pengantar Gletser Amazon S3](#)

Topik ini menunjukkan contoh AWS CLI perintah yang melakukan tugas umum untuk S3 Glacier. Contoh menunjukkan cara menggunakan file untuk mengunggah file besar AWS CLI ke S3 Glacier dengan membaginya menjadi bagian-bagian yang lebih kecil dan mengunggahnya dari baris perintah.

Anda dapat mengakses fitur Amazon S3 Glacier menggunakan (). AWS Command Line Interface AWS CLI Untuk membuat daftar AWS CLI perintah untuk S3 Glacier, gunakan perintah berikut.

```
aws glacier help
```

### Note

Untuk referensi perintah dan contoh tambahan, lihat [aws glacier](#) di AWS CLICommand Reference.

Topik

- [Prasyarat](#)
- [Buat lemari besi Amazon S3 Glacier](#)
- [Siapkan file untuk diunggah](#)

- [Memulai upload dan meng-upload file multipart](#)
- [Selesaikan unggahan](#)
- [Sumber daya](#)

## Prasyarat

Untuk menjalankan `glacier` perintah, Anda perlu:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, silakan lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Tutorial ini menggunakan beberapa alat baris perintah yang biasanya sudah diinstal sebelumnya pada sistem operasi mirip Unix, termasuk Linux dan macOS. Pengguna Windows dapat menggunakan alat yang sama dengan menginstal [Cygwin](#) dan menjalankan perintah dari terminal Cygwin. Kami mencatat perintah dan utilitas asli Windows yang melakukan fungsi yang sama jika tersedia.

## Buat lemari besi Amazon S3 Glacier

Buat lemari besi dengan [create-vault](#) perintah.

```
$ aws glacier create-vault --account-id - --vault-name myvault
{
  "location": "/123456789012/vaults/myvault"
}
```

### Note

Semua perintah S3 Glacier memerlukan parameter ID akun. Gunakan karakter tanda hubung (`--account-id -`) untuk menggunakan akun saat ini.

## Siapkan file untuk diunggah

Buat file untuk unggahan pengujian. Perintah berikut membuat file bernama *largefile* yang berisi persis 3 MiB data acak.

Linux atau macOS

```
$ dd if=/dev/urandom of=largefile bs=3145728 count=1
1+0 records in
1+0 records out
3145728 bytes (3.1 MB) copied, 0.205813 s, 15.3 MB/s
```

dd adalah utilitas yang menyalin sejumlah byte dari file input ke file output. Contoh sebelumnya menggunakan file perangkat sistem /dev/urandom sebagai sumber data acak. fsutil melakukan fungsi serupa di Windows.

Jendela

```
C:\> fsutil file createnew largefile 3145728
File C:\temp\largefile is created
```

Selanjutnya, bagi file menjadi potongan 1 MiB (1.048.576 byte).

```
$ split -b 1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'
```

#### Note

[HJ-Split](#) adalah file splitter gratis untuk Windows dan banyak platform lainnya.

## Memulai upload dan meng-upload file multipart

Buat unggahan multibagian di Amazon S3 Glacier dengan menggunakan perintah. [initiate-multipart-upload](#)

```
$ aws glacier initiate-multipart-upload --account-id - --archive-description "multipart upload test" --part-size 1048576 --vault-name myvault
{
  "uploadId": "19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ",
  "location": "/123456789012/vaults/myvault/multipart-
uploads/19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
```



```
}

```

S3 Glacier memerlukan ukuran setiap bagian dalam byte (1 MiB dalam contoh ini), nama vault Anda, dan ID akun untuk mengonfigurasi unggahan multibagian. AWS CLI Output ID upload ketika operasi selesai. Simpan ID unggahan ke variabel shell untuk digunakan nanti.

Linux atau macOS

```
$ UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
```

Jendela

```
C:\> set UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
```

Selanjutnya, gunakan [upload-multipart-part](#) perintah untuk mengunggah masing-masing dari tiga bagian.

```
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkaa --range 'bytes
0-1048575/*' --account-id - --vault-name myvault
{
  "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkab --range 'bytes
1048576-2097151/*' --account-id - --vault-name myvault
{
  "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkac --range 'bytes
2097152-3145727/*' --account-id - --vault-name myvault
{
  "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
```

### Note

Contoh sebelumnya menggunakan tanda dolar (\$) untuk referensi isi variabel UPLOADID shell di Linux. Pada baris perintah Windows, gunakan tanda persen (%) di kedua sisi nama variabel (misalnya,%UPLOADID%).

Anda harus menentukan rentang byte setiap bagian saat Anda mengunggahnya sehingga S3 Glacier dapat memasangnya kembali dalam urutan yang benar. Setiap bagian adalah 1.048.576 byte, sehingga bagian pertama menempati byte 0-1048575, yang kedua 1048576-2097151, dan yang ketiga 2097152-3145727.

## Selesaikan unggahan

Amazon S3 Glacier memerlukan hash pohon dari file asli untuk mengonfirmasi bahwa semua bagian yang diunggah mencapai utuh. AWS

Untuk menghitung hash pohon, Anda harus membagi file menjadi 1 bagian MiB dan menghitung hash SHA-256 biner dari setiap bagian. Kemudian Anda membagi daftar hash menjadi pasangan, menggabungkan dua hash biner di setiap pasangan, dan mengambil hash dari hasilnya. Ulangi proses ini sampai hanya ada satu hash yang tersisa. Jika ada jumlah ganjil hash di tingkat mana pun, promosikan ke tingkat berikutnya tanpa memodifikasinya.

Kunci untuk menghitung hash pohon dengan benar saat menggunakan utilitas baris perintah adalah menyimpan setiap hash dalam format biner dan mengkonversi ke heksadesimal hanya pada langkah terakhir. Menggabungkan atau hashing versi heksadesimal dari setiap hash di pohon akan menyebabkan hasil yang salah.

### Note

Pengguna Windows dapat menggunakan type perintah sebagai penggantinya. [OpenSSL tersedia untuk Windows di OpenSSL.org.](#)

Untuk menghitung hash pohon

1. Jika Anda belum melakukannya, pisahkan file asli menjadi 1 bagian MiB.

```
$ split --bytes=1048576 --verbose largefile chunk
creating file `chunkaa`
creating file `chunkab`
creating file `chunkac`
```

2. Hitung dan simpan hash biner SHA-256 dari setiap potongan.

```
$ openssl dgst -sha256 -binary chunkaa > hash1
$ openssl dgst -sha256 -binary chunkab > hash2
```

```
$ openssl dgst -sha256 -binary chunkac > hash3
```

- Gabungkan dua hash pertama dan ambil hash biner hasilnya.

```
$ cat hash1 hash2 > hash12
$ openssl dgst -sha256 -binary hash12 > hash12hash
```

- Gabungkan hash induk dari potongan aa dan ab dengan hash potongan dan hash hasilnya, kali ini mengeluarkan ac heksadesimal. Simpan hasilnya dalam variabel shell.

```
$ cat hash12hash hash3 > hash123
$ openssl dgst -sha256 hash123
SHA256(hash123)= 9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
$ TREEHASH=9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
```

Terakhir, selesaikan unggahan dengan [complete-multipart-upload](#) perintah. Perintah ini mengambil ukuran file asli dalam byte, nilai hash pohon akhir dalam heksadesimal, dan ID akun dan nama vault Anda.

```
$ aws glacier complete-multipart-upload --checksum $TREEHASH --archive-size 3145728 --
upload-id $UPLOADID --account-id - --vault-name myvault
{
  "archiveId": "d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrALLGAAONJAzo5QdP-
N83MKqd96Unspoa5H51ItWX-sK8-QS0ZhwsyGiu9-R-
kWUyS1dSB1mgPPWkEbeFfqDSav053rU7FvVLHfRc6hg",
  "checksum": "9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
  "location": "/123456789012/vaults/myvault/archives/
d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrALLGAAONJAzo5QdP-N83MKqd96Unspoa5H51ItWX-sK8-
QS0ZhwsyGiu9-R-kWUyS1dSB1mgPPWkEbeFfqDSav053rU7FvVLHfRc6hg"
}
```

Anda juga dapat memeriksa status brankas menggunakan [describe-vault](#) perintah.

```
$ aws glacier describe-vault --account-id - --vault-name myvault
{
  "SizeInBytes": 3178496,
  "VaultARN": "arn:aws:glacier:us-west-2:123456789012:vaults/myvault",
  "LastInventoryDate": "2018-12-07T00:26:19.028Z",
  "NumberOfArchives": 1,
  "CreationDate": "2018-12-06T21:23:45.708Z",
  "VaultName": "myvault"
```

```
}
```

### Note

Status Vault diperbarui sekitar sekali per hari. Lihat [Bekerja dengan Vaults untuk informasi lebih lanjut](#).

Sekarang aman untuk menghapus file chunk dan hash yang Anda buat.

```
$ rm chunk* hash*
```

Untuk informasi selengkapnya tentang unggahan multibagian, lihat [Mengunggah Arsip Besar dalam Checksum Suku Cadang dan Komputasi di Panduan Pengembang Gletser](#) Amazon S3.

## Sumber daya

AWS CLIREferensi:

- [aws glacier](#)
- [aws glacier complete-multipart-upload](#)
- [aws glacier create-vault](#)
- [aws glacier describe-vault](#)
- [aws glacier initiate-multipart-upload](#)

Referensi layanan:

- [Panduan Pengembang Amazon S3 Glacier](#)
- [Mengunggah Arsip Besar di Bagian dalam Panduan](#) Pengembang Gletser Amazon S3
- [Checksum Komputasi](#) di Panduan Pengembang Amazon S3 Glacier
- [Bekerja dengan Vaults di Panduan Pengembang](#) Amazon S3 Glacier

# Gunakan AWS Identity and Access Management dari AWS CLI

## Pengantar AWS Identity and Access Management

### [Pengantar AWS Identity and Access Management](#)

Anda dapat mengakses fitur AWS Identity and Access Management (IAM) menggunakan AWS Command Line Interface (AWS CLI). Untuk membuat daftar AWS CLI perintah untuk IAM, gunakan perintah berikut.

```
aws iam help
```

Topik ini menunjukkan contoh AWS CLI perintah yang melakukan tugas-tugas umum untuk IAM.

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Untuk informasi selengkapnya tentang layanan IAM, lihat [Panduan AWS Identity and Access Management Pengguna](#).

### Topik

- [Buat pengguna dan grup IAM](#)
- [Lampirkan kebijakan terkelola IAM ke pengguna](#)
- [Tetapkan kata sandi awal untuk pengguna IAM](#)
- [Buat kunci akses untuk pengguna IAM](#)

## Buat pengguna dan grup IAM

Topik ini menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) perintah untuk membuat grup AWS Identity and Access Management (IAM) dan pengguna baru, lalu menambahkan pengguna ke grup. Untuk informasi selengkapnya tentang layanan IAM, lihat [Panduan AWS Identity and Access Management Pengguna](#).

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Untuk membuat grup dan menambahkan pengguna baru ke dalamnya

1. Gunakan perintah untuk membuat grup.

```
$ aws iam create-group --group-name MyIamGroup
{
  "Group": {
    "GroupName": "MyIamGroup",
    "CreateDate": "2018-12-14T03:03:52.834Z",
    "GroupId": "AGPAJNUJ2W4IJVEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/MyIamGroup",
    "Path": "/"
  }
}
```

2. Gunakan perintah untuk membuat pengguna.

```
$ aws iam create-user --user-name MyUser
{
  "User": {
    "UserName": "MyUser",
    "Path": "/",
    "CreateDate": "2018-12-14T03:13:02.581Z",
    "UserId": "AIDAJY2PE5XUZ4EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/MyUser"
  }
}
```

3. Gunakan perintah untuk menambahkan pengguna ke grup.

```
$ aws iam add-user-to-group --user-name MyUser --group-name MyIamGroup
```

4. Untuk memverifikasi bahwa *MyIamGroup* grup berisi *MyUser*, gunakan perintah.

```
$ aws iam get-group --group-name MyIamGroup
{
  "Group": {
    "GroupName": "MyIamGroup",
    "CreateDate": "2018-12-14T03:03:52Z",
    "GroupId": "AGPAJNUJ2W4IJVEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/MyIamGroup",
    "Path": "/"
  },
}
```

```

    "Users": [
      {
        "UserName": "MyUser",
        "Path": "/",
        "CreateDate": "2018-12-14T03:13:02Z",
        "UserId": "AIDAJY2PE5XUZ4EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:user/MyUser"
      }
    ],
    "IsTruncated": "false"
  }

```

## Lampirkan kebijakan terkelola IAM ke pengguna

Topik ini menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) perintah untuk melampirkan kebijakan AWS Identity and Access Management (IAM) ke pengguna. Kebijakan dalam contoh ini memberi pengguna “Akses Pengguna Daya”. Untuk informasi selengkapnya tentang layanan IAM, lihat [Panduan AWS Identity and Access Management Pengguna](#).

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Untuk melampirkan kebijakan terkelola IAM ke pengguna

1. Tentukan Nama Sumber Daya Amazon (ARN) dari kebijakan yang akan dilampirkan. Perintah berikut digunakan `list-policies` untuk menemukan ARN kebijakan dengan nama `PowerUserAccess` Kemudian menyimpan ARN itu dalam variabel lingkungan.

```

$ export POLICYARN=$(aws iam list-policies --query 'Policies[?
PolicyName==`PowerUserAccess`].{ARN:Arn}' --output text) ~
$ echo $POLICYARN
arn:aws:iam::aws:policy/PowerUserAccess

```

2. Untuk melampirkan kebijakan, gunakan perintah, dan referensi variabel lingkungan yang memegang ARN kebijakan.

```

$ aws iam attach-user-policy --user-name MyUser --policy-arn $POLICYARN

```

3. Verifikasi bahwa kebijakan dilampirkan ke pengguna dengan menjalankan perintah.

```

$ aws iam list-attached-user-policies --user-name MyUser

```

```
{
  "AttachedPolicies": [
    {
      "PolicyName": "PowerUserAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/PowerUserAccess"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Sumber Daya Manajemen Akses](#). Topik ini menyediakan tautan ke ikhtisar izin dan kebijakan, serta tautan ke contoh kebijakan untuk mengakses Amazon S3, Amazon EC2, dan layanan lainnya.

## Tetapkan kata sandi awal untuk pengguna IAM

Topik ini menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) perintah untuk menetapkan kata sandi awal untuk pengguna AWS Identity and Access Management (IAM). Untuk informasi selengkapnya tentang layanan IAM, lihat [Panduan AWS Identity and Access Management Pengguna](#).

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Perintah berikut digunakan [create-login-profile](#) untuk mengatur kata sandi awal pada pengguna yang ditentukan. Saat pengguna masuk untuk pertama kalinya, pengguna diminta untuk mengubah kata sandi menjadi sesuatu yang hanya diketahui pengguna.

```
$ aws iam create-login-profile --user-name MyUser --password My!User1Login8P@ssword --password-reset-required
{
  "LoginProfile": {
    "UserName": "MyUser",
    "CreateDate": "2018-12-14T17:27:18Z",
    "PasswordResetRequired": true
  }
}
```

Anda dapat menggunakan `update-login-profile` perintah untuk mengubah kata sandi untuk pengguna.



```
$ aws iam update-login-profile --user-name MyUser --password My!User1ADifferentP@ssword
```

## Buat kunci akses untuk pengguna IAM

Topik ini menjelaskan cara menggunakan AWS Command Line Interface (AWS CLI) perintah untuk membuat satu set kunci akses untuk pengguna AWS Identity and Access Management (IAM). Untuk informasi selengkapnya tentang layanan IAM, lihat [Panduan AWS Identity and Access Management Pengguna](#).

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Anda dapat menggunakan perintah untuk membuat kunci akses bagi pengguna. Kunci akses adalah seperangkat kredensi keamanan yang terdiri dari ID kunci akses dan kunci rahasia.

Seorang pengguna hanya dapat membuat dua kunci akses pada satu waktu. Jika Anda mencoba membuat set ketiga, perintah mengembalikan LimitExceeded kesalahan.

```
$ aws iam create-access-key --user-name MyUser
{
  "AccessKey": {
    "UserName": "MyUser",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "Status": "Active",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "CreateDate": "2018-12-14T17:34:16Z"
  }
}
```

Gunakan perintah untuk menghapus kunci akses bagi pengguna. Tentukan kunci akses mana yang akan dihapus dengan menggunakan ID kunci akses.

```
$ aws iam delete-access-key --user-name MyUser --access-key-id AKIAIOSFODNN7EXAMPLE
```

# Gunakan Amazon S3 dengan AWS CLI

## Pengantar Amazon Simple Storage Service (Amazon S3)

### [Pengantar Layanan Penyimpanan Sederhana Amazon \(Amazon S3 - Penyimpanan Cloud aktif AWS\)](#)

Anda dapat mengakses fitur Amazon Simple Storage Service (Amazon S3) Simple Storage Service (). AWS Command Line Interface AWS CLI AWS CLI ini menyediakan dua tingkatan perintah untuk mengakses Amazon S3:

- `s3` — Perintah tingkat tinggi yang menyederhanakan melakukan tugas-tugas umum, seperti membuat, memanipulasi, dan menghapus objek dan ember.
- `s3api` - Mengekspos akses langsung ke semua operasi API Amazon S3 yang memungkinkan Anda untuk melakukan operasi lanjutan.

Topik dalam panduan ini:

- [Gunakan perintah tingkat tinggi \(`s3`\) dengan AWS CLI](#)
- [Gunakan perintah API-level \(`s3api`\) dengan AWS CLI](#)
- [Contoh skrip operasi siklus hidup bucket Amazon S3](#)

## Gunakan perintah tingkat tinggi (`s3`) dengan AWS CLI

Topik ini menjelaskan beberapa perintah yang dapat Anda gunakan untuk mengelola bucket dan objek Amazon S3 menggunakan perintah `aws s3`. Untuk perintah yang tidak tercakup dalam topik ini dan contoh perintah tambahan, lihat perintah di [AWS CLI Referensi](#).

`aws s3` Perintah tingkat tinggi menyederhanakan pengelolaan objek Amazon S3. Perintah ini memungkinkan Anda untuk mengelola konten Amazon S3 di dalam dirinya sendiri dan dengan direktori lokal.

Topik

- [Prasyarat](#)
- [Sebelum Anda mulai](#)
- [Membuat bucket](#)

- [Daftar ember dan benda](#)
- [Hapus ember](#)
- [Hapus objek](#)
- [Pindahkan objek](#)
- [Salin objek](#)
- [Sinkronkan objek](#)
- [Opsi yang sering digunakan untuk perintah s3](#)
- [Sumber daya](#)

## Prasyarat

Untuk menjalankan s3 perintah, Anda perlu:

- Instal dan konfigurasi AWS CLI. Lihat informasi yang lebih lengkap di [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Profil yang Anda gunakan harus memiliki izin yang memungkinkan AWS operasi dilakukan oleh contoh.
- Pahami istilah Amazon S3 ini:
  - Bucket - Folder Amazon S3 tingkat atas.
  - Awalan - Folder Amazon S3 dalam ember.
  - Objek - Item apa pun yang di-host di ember Amazon S3.

## Sebelum Anda mulai

Bagian ini menjelaskan beberapa hal yang perlu diperhatikan sebelum Anda menggunakan `aws s3` perintah.

## Unggahan objek besar

Saat Anda menggunakan `aws s3` perintah untuk mengunggah objek besar ke bucket Amazon S3, AWS CLI secara otomatis melakukan pengunggahan multibagian. Anda tidak dapat melanjutkan unggahan yang gagal saat menggunakan `aws s3` perintah ini.

Jika unggahan multipart gagal karena batas waktu, atau jika Anda membatalkan secara manual AWS CLI, maka akan AWS CLI menghentikan unggahan dan membersihkan file apa pun yang dibuat. Proses ini dapat memakan waktu beberapa menit.

Jika proses upload atau pembersihan multipart dibatalkan oleh perintah kill atau kegagalan sistem, file yang dibuat tetap berada di bucket Amazon S3. [Untuk membersihkan unggahan multipart, gunakan perintah `s3api . abort-multipart-upload`](#)

## Properti file dan tag dalam salinan multipart

Saat Anda menggunakan perintah AWS CLI versi 1 di `aws s3` namespace untuk menyalin file dari satu lokasi bucket Amazon S3 ke lokasi bucket Amazon S3 lainnya, dan operasi tersebut [menggunakan salinan multibagian](#), tidak ada properti file dari objek sumber yang disalin ke objek tujuan.

Secara default, perintah AWS CLI versi 2 di `s3` namespace yang melakukan salinan multipart mentransfer semua tag dan kumpulan properti berikut dari sumber ke salinan tujuan: `content-type,,,,content-language, content-encoding content-dispositioncache-control, expires` dan `metadata`

Ini dapat menghasilkan panggilan AWS API tambahan ke titik akhir Amazon S3 yang tidak akan dibuat jika Anda menggunakan AWS CLI versi 1. Ini dapat mencakup: `HeadObject,GetObjectTagging, danPutObjectTagging`.

Jika Anda perlu mengubah perilaku default ini dalam perintah AWS CLI versi 2, gunakan `--copy-props` parameter untuk menentukan salah satu opsi berikut:

- `default` - Nilai default. Menentukan bahwa salinan mencakup semua tag yang dilampirkan ke objek sumber dan properti yang dicakup oleh `--metadata-directive` parameter yang digunakan untuk salinan non-multipart: `content-type,,,,, content-languagecontent-encoding, content-disposition` dan `cache-control expires metadata`
- `metadata-directive` - Menentukan bahwa salinan hanya mencakup properti yang dicakup oleh parameter yang digunakan untuk salinan non-multipart. `--metadata-directive` itu tidak menyalin tag apa pun.
- `none` - Menentukan bahwa salinan termasuk tidak ada properti dari objek sumber.

## Membuat bucket

Gunakan perintah untuk membuat ember. Nama bucket harus unik secara global (unik di semua Amazon S3) dan harus sesuai dengan DNS.

Nama bucket dapat berisi huruf kecil, angka, tanda hubung, dan titik. Nama bucket dapat dimulai dan diakhiri hanya dengan huruf atau angka, dan tidak dapat berisi titik di sebelah tanda hubung atau periode lain.

## Sintaksis

```
$ aws s3 mb <target> [--options]
```

contoh s3 mb

Contoh berikut membuat `s3://bucket-name` bucket.

```
$ aws s3 mb s3://bucket-name
```

## Daftar ember dan benda

Untuk membuat daftar bucket, folder, atau objek Anda, gunakan perintah. Menggunakan perintah tanpa target atau opsi mencantumkan semua ember.

## Sintaksis

```
$ aws s3 ls <target> [--options]
```

Untuk beberapa opsi umum untuk digunakan dengan perintah ini, dan contoh, lihat [Opsi yang sering digunakan untuk perintah s3](#). Untuk daftar lengkap opsi yang tersedia, lihat di Referensi AWS CLI Perintah.

contoh s3 ls

Contoh berikut mencantumkan semua bucket Amazon S3 Anda.

```
$ aws s3 ls
2018-12-11 17:08:50 my-bucket
2018-12-14 14:55:44 my-bucket2
```

Perintah berikut mencantumkan semua objek dan awalan dalam ember. Dalam contoh output ini, awalan `example/` memiliki satu file bernama `MyFile1.txt`.

```
$ aws s3 ls s3://bucket-name
PRE example/
```

```
2018-12-04 19:05:48      3 MyFile1.txt
```

Anda dapat memfilter output ke awalan tertentu dengan memasukkannya ke dalam perintah. *Perintah berikut mencantumkan objek dalam bucket-name/example/ (yaitu, objek dalam bucket-name disaring oleh awalan example/).*

```
$ aws s3 ls s3://bucket-name/example/  
2018-12-06 18:59:32      3 MyFile1.txt
```

## Hapus ember

Untuk menghapus ember, gunakan perintah.

### Sintaksis

```
$ aws s3 rb <target> [--options]
```

contoh s3 rb

Contoh berikut menghapus s3://bucket-name ember.

```
$ aws s3 rb s3://bucket-name
```

Secara default, bucket harus kosong agar operasi berhasil. Untuk menghapus ember yang tidak kosong, Anda harus menyertakan `--force` opsi. Jika Anda menggunakan bucket berversi yang berisi objek yang sebelumnya dihapus—tetapi dipertahankan—, perintah ini tidak memungkinkan Anda untuk menghapus bucket. Anda harus terlebih dahulu menghapus semua konten.

Contoh berikut menghapus semua objek dan awalan di bucket, dan kemudian menghapus bucket.

```
$ aws s3 rb s3://bucket-name --force
```

## Hapus objek

Untuk menghapus objek dalam ember atau direktori lokal Anda, gunakan perintah.

### Sintaksis

```
$ aws s3 rm <target> [--options]
```

Untuk beberapa opsi umum untuk digunakan dengan perintah ini, dan contoh, lihat [Opsi yang sering digunakan untuk perintah s3](#). Untuk daftar lengkap opsi, lihat di Referensi AWS CLI Perintah.

s3 rm contoh

Contoh berikut menghapus `filename.txt` dari `s3://bucket-name/example`.

```
$ aws s3 rm s3://bucket-name/example/filename.txt
```

Contoh berikut menghapus semua objek dari `s3://bucket-name/example` menggunakan `--recursive` opsi.

```
$ aws s3 rm s3://bucket-name/example --recursive
```

Pindahkan objek

Gunakan perintah untuk memindahkan objek dari ember atau direktori lokal. `s3 mv` Perintah menyalin objek sumber atau file ke tujuan yang ditentukan dan kemudian menghapus objek sumber atau file.

Sintaksis

```
$ aws s3 mv <source> <target> [--options]
```

Untuk beberapa opsi umum untuk digunakan dengan perintah ini, dan contoh, lihat [Opsi yang sering digunakan untuk perintah s3](#). Untuk daftar lengkap opsi yang tersedia, lihat di Referensi AWS CLI Perintah.

#### Warning

Jika Anda menggunakan semua jenis ARN titik akses atau alias titik akses di URI sumber atau tujuan Amazon S3 Anda, Anda harus berhati-hati agar URI Amazon S3 sumber dan tujuan Anda diselesaikan ke bucket dasar yang berbeda. Jika bucket sumber dan tujuan sama, file sumber atau objek dapat dipindahkan ke dirinya sendiri, yang dapat mengakibatkan penghapusan file sumber atau objek Anda secara tidak sengaja. Untuk memverifikasi bahwa bucket sumber dan tujuan tidak sama, gunakan `--validate-same-s3-paths` parameter, atau atur variabel lingkungan [AWS\\_CLI\\_S3\\_MV\\_VALIDATE\\_SAME\\_S3\\_PATHS](#) ke `true`.

## s3 mv contoh

Contoh berikut memindahkan semua objek dari `s3://bucket-name/example` ke `s3://my-bucket/`.

```
$ aws s3 mv s3://bucket-name/example s3://my-bucket/
```

Contoh berikut memindahkan file lokal dari direktori kerja Anda saat ini ke bucket Amazon S3 dengan perintah. `s3 mv`

```
$ aws s3 mv filename.txt s3://bucket-name
```

Contoh berikut memindahkan file dari bucket Amazon S3 ke direktori kerja saat ini, tempat `./` menentukan direktori kerja Anda saat ini.

```
$ aws s3 mv s3://bucket-name/filename.txt ./
```

## Salin objek

Gunakan perintah untuk menyalin objek dari ember atau direktori lokal.

## Sintaksis

```
$ aws s3 cp <source> <target> [--options]
```

Anda dapat menggunakan parameter dasbor untuk streaming file ke input standar (`stdin`) atau output standar (`stdout`).

### Warning

Jika Anda menggunakan PowerShell, shell mungkin mengubah pengkodean CRLF atau menambahkan CRLF ke input atau output pipa, atau output yang dialihkan.

`s3 cp` Perintah menggunakan sintaks berikut untuk mengunggah aliran file dari `stdin` bucket tertentu.

## Sintaksis



```
$ aws s3 cp - <target> [--options]
```

s3 cp Perintah menggunakan sintaks berikut untuk mengunduh aliran file Amazon S3 untuk. stdout

### Sintaksis

```
$ aws s3 cp <target> [--options] -
```

Untuk beberapa opsi umum untuk digunakan dengan perintah ini, dan contoh, lihat [Opsi yang sering digunakan untuk perintah s3](#). Untuk daftar lengkap opsi, lihat di Referensi AWS CLI Perintah.

### s3 cp contoh

Contoh berikut menyalin semua objek dari s3://bucket-name/example ke s3://my-bucket/.

```
$ aws s3 cp s3://bucket-name/example s3://my-bucket/
```

Contoh berikut menyalin file lokal dari direktori kerja Anda saat ini ke bucket Amazon S3 dengan perintah. s3 cp

```
$ aws s3 cp filename.txt s3://bucket-name
```

Contoh berikut menyalin file dari bucket Amazon S3 Anda ke direktori kerja Anda saat ini, yang ./ menentukan direktori kerja Anda saat ini.

```
$ aws s3 cp s3://bucket-name/filename.txt ./
```

Contoh berikut menggunakan echo untuk mengalirkan teks "hello world" ke s3://bucket-name/filename.txt file.

```
$ echo "hello world" | aws s3 cp - s3://bucket-name/filename.txt
```

Contoh berikut mengalirkan s3://bucket-name/filename.txt file ke stdout dan mencetak konten ke konsol.

```
$ aws s3 cp s3://bucket-name/filename.txt -  
hello world
```

Contoh berikut mengalirkan konten `s3://bucket-name/pre` ke `tostdout`, menggunakan `bzip2` perintah untuk mengompres file, dan mengunggah file terkompresi baru yang diberi nama `key.bz2` ke `s3://bucket-name`

```
$ aws s3 cp s3://bucket-name/pre - | bzip2 --best | aws s3 cp - s3://bucket-name/key.bz2
```

## Sinkronkan objek

Perintah menyinkronkan isi ember dan direktori, atau isi dua ember. Biasanya, `s3 sync` menyalin file atau objek yang hilang atau usang antara sumber dan target. Namun, Anda juga dapat menyediakan `--delete` opsi untuk menghapus file atau objek dari target yang tidak ada di sumbernya.

## Sintaksis

```
$ aws s3 sync <source> <target> [--options]
```

Untuk beberapa opsi umum untuk digunakan dengan perintah ini, dan contoh, lihat [Opsi yang sering digunakan untuk perintah s3](#). Untuk daftar lengkap opsi, lihat di Referensi AWS CLI Perintah.

## contoh sinkronisasi s3

Contoh berikut menyinkronkan konten jalur bernama awalan Amazon S3 di bucket bernama `my-bucket` dengan direktori kerja saat ini.

`s3 sync` memperbarui file apa pun yang memiliki ukuran atau waktu modifikasi yang berbeda dari file dengan nama yang sama di tempat tujuan. Output menampilkan operasi tertentu yang dilakukan selama sinkronisasi. Perhatikan bahwa operasi secara rekursif menyinkronkan subdirektori `MySubdirectory` dan isinya dengan `s3://my-bucket/path/MySubdirectory`

```
$ aws s3 sync . s3://my-bucket/path
upload: MySubdirectory\MyFile3.txt to s3://my-bucket/path/MySubdirectory/MyFile3.txt
upload: MyFile2.txt to s3://my-bucket/path/MyFile2.txt
upload: MyFile1.txt to s3://my-bucket/path/MyFile1.txt
```

Contoh berikut, yang memperluas yang sebelumnya, menunjukkan cara menggunakan `--delete` opsi.

```
// Delete local file
$ rm ./MyFile1.txt
```

```
// Attempt sync without --delete option - nothing happens
$ aws s3 sync . s3://my-bucket/path

// Sync with deletion - object is deleted from bucket
$ aws s3 sync . s3://my-bucket/path --delete
delete: s3://my-bucket/path/MyFile1.txt

// Delete object from bucket
$ aws s3 rm s3://my-bucket/path/MySubdirectory/MyFile3.txt
delete: s3://my-bucket/path/MySubdirectory/MyFile3.txt

// Sync with deletion - local file is deleted
$ aws s3 sync s3://my-bucket/path . --delete
delete: MySubdirectory\MyFile3.txt

// Sync with Infrequent Access storage class
$ aws s3 sync . s3://my-bucket/path --storage-class STANDARD_IA
```

Saat menggunakan `--delete` opsi, `--include` opsi `--exclude` dan dapat memfilter file atau objek yang akan dihapus selama `s3 sync` operasi. Dalam hal ini, string parameter harus menentukan file yang akan dikecualikan dari, atau menyertakan untuk, penghapusan dalam konteks direktori target atau bucket. Bagian berikut menunjukkan satu contoh.

```
Assume local directory and s3://my-bucket/path currently in sync and each contains 3
files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt
...

// Sync with delete, excluding files that match a pattern. MyFile88.txt is deleted,
while remote MyFile1.txt is not.
$ aws s3 sync . s3://my-bucket/path --delete --exclude "path/MyFile?.txt"
delete: s3://my-bucket/path/MyFile88.txt
...

// Sync with delete, excluding MyFile2.rtf - local file is NOT deleted
$ aws s3 sync s3://my-bucket/path . --delete --exclude "./MyFile2.rtf"
download: s3://my-bucket/path/MyFile1.txt to MyFile1.txt
...

// Sync with delete, local copy of MyFile2.rtf is deleted
```

```
$ aws s3 sync s3://my-bucket/path . --delete
delete: MyFile2.rtf
```

Opsi yang sering digunakan untuk perintah s3

Opsi berikut sering digunakan untuk perintah yang dijelaskan dalam topik ini. Untuk daftar lengkap opsi yang dapat Anda gunakan pada perintah, lihat perintah khusus dalam [versi 2 panduan referensi](#).

acl

s3 sync dan s3 cp dapat menggunakan --acl opsi. Ini memungkinkan Anda untuk mengatur izin akses untuk file yang disalin ke Amazon S3. --acl Opsi menerima private, public-read, dan public-read-write nilai. Untuk informasi selengkapnya, lihat [ACL Kalengan](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

```
$ aws s3 sync . s3://my-bucket/path --acl public-read
```

mengecualikan

Saat Anda menggunakan s3 cp, s3 mv, s3 sync, atau s3 rm perintah, Anda dapat memfilter hasil dengan menggunakan --include opsi --exclude or. --exclude Opsi menetapkan aturan untuk hanya mengecualikan objek dari perintah, dan opsi berlaku dalam urutan yang ditentukan. Seperti yang ditunjukkan dalam contoh berikut.

```
Local directory contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt

// Exclude all .txt files, resulting in only MyFile2.rtf being copied
$ aws s3 cp . s3://my-bucket/path --exclude "*.txt"

// Exclude all .txt files but include all files with the "MyFile*.txt" format,
resulting in, MyFile1.txt, MyFile2.rtf, MyFile88.txt being copied
$ aws s3 cp . s3://my-bucket/path --exclude "*.txt" --include "MyFile*.txt"

// Exclude all .txt files, but include all files with the "MyFile*.txt" format,
but exclude all files with the "MyFile?.txt" format resulting in, MyFile2.rtf and
MyFile88.txt being copied
$ aws s3 cp . s3://my-bucket/path --exclude "*.txt" --include "MyFile*.txt" --
exclude "MyFile?.txt"
```

## memasukkan

Saat Anda menggunakan `cp`, `mv`, `sync`, atau `rm` perintah, Anda dapat memfilter hasil menggunakan `--include` opsi `--exclude` or. `--include` Opsi menetapkan aturan untuk hanya menyertakan objek yang ditentukan untuk perintah, dan opsi berlaku dalam urutan yang ditentukan. Seperti yang ditunjukkan dalam contoh berikut.

```
Local directory contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt

// Include all .txt files, resulting in MyFile1.txt and MyFile88.txt being copied
$ aws s3 cp . s3://my-bucket/path --include "*.txt"

// Include all .txt files but exclude all files with the "MyFile*.txt" format,
resulting in no files being copied
$ aws s3 cp . s3://my-bucket/path --include "*.txt" --exclude "MyFile*.txt"

// Include all .txt files, but exclude all files with the "MyFile*.txt" format, but
include all files with the "MyFile?.txt" format resulting in MyFile1.txt being
copied

$ aws s3 cp . s3://my-bucket/path --include "*.txt" --exclude "MyFile*.txt" --
include "MyFile?.txt"
```

## pemberian izin

`sync` Perintah `cp`, `mv`, dan menyertakan `--grants` opsi yang dapat Anda gunakan untuk memberikan izin pada objek kepada pengguna atau grup tertentu. Atur `--grants` opsi ke daftar izin menggunakan sintaks berikut. Ganti `Permission`, `Grantee_Type`, dan `Grantee_ID` dengan nilai Anda sendiri.

### Sintaksis

```
--grants Permission=Grantee_Type=Grantee_ID
        [Permission=Grantee_Type=Grantee_ID ...]
```

Setiap nilai mengandung elemen-elemen berikut:

- *Izin* - Menentukan izin yang diberikan. Dapat diatur `read`, `readacl`, `writeacl`, atau `full`.

- *Grantee\_Type* - Menentukan bagaimana mengidentifikasi penerima hibah. Dapat diatur ke `uri`, `emailaddress`, atau `id`.
- *Grantee\_ID* - Menentukan penerima berdasarkan *Grantee\_Type*.
  - `uri`— URI kelompok. Untuk informasi lebih lanjut, lihat [Siapa penerima hibah?](#)
  - `emailaddress`— Alamat email akun.
  - `id`— ID kanonik akun.

Untuk informasi selengkapnya tentang kontrol akses Amazon S3, lihat Kontrol [akses](#).

Contoh berikut menyalin objek ke dalam ember. Ini memberikan `read` izin pada objek untuk semua orang, dan `full` izin (`readacl`, dan `writeacl`) ke akun yang terkait dengan `user@example.com`

```
$ aws s3 cp file.txt s3://my-bucket/ --grants read=uri=http://acs.amazonaws.com/groups/global/AllUsers full=emailaddress=user@example.com
```

Anda juga dapat menentukan kelas penyimpanan nondefault (`REDUCED_REDUNDANCY` atau `STANDARD_IA`) untuk objek yang Anda unggah ke Amazon S3. Untuk melakukan ini, gunakan `--storage-class` opsi.

```
$ aws s3 cp file.txt s3://my-bucket/ --storage-class REDUCED_REDUNDANCY
```

## rekursif

Bila Anda menggunakan opsi ini, perintah dilakukan pada semua file atau objek di bawah direktori atau awalan yang ditentukan. Contoh berikut menghapus `s3://my-bucket/path` dan semua isinya.

```
$ aws s3 rm s3://my-bucket/path --recursive
```

## Sumber daya

### AWS CLI Referensi:

- [aws s3](#)
- [aws s3 cp](#)
- [aws s3 mb](#)

- [aws s3 mv](#)
- [aws s3 ls](#)
- [aws s3 rb](#)
- [aws s3 rm](#)
- [aws s3 sync](#)

Referensi layanan:

- [Bekerja dengan bucket Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon
- [Bekerja dengan objek Amazon S3 di Panduan](#) Pengguna Layanan Penyimpanan Sederhana Amazon
- [Kunci daftar secara hierarkis menggunakan awalan dan pembatas dalam Panduan Pengguna](#) Layanan Penyimpanan Sederhana Amazon
- [Batalkan unggahan multibagian ke bucket S3 menggunakan AWS SDK for .NET \(tingkat rendah\) di](#) Panduan Pengguna Layanan Penyimpanan Sederhana Amazon

## Gunakan perintah API-level (s3api) dengan AWS CLI

Perintah API-level (terkandung dalam set `s3api` perintah) menyediakan akses langsung ke Amazon Simple Storage Service (Amazon S3) API, dan mengaktifkan beberapa operasi yang tidak diekspos dalam perintah tingkat tinggi. `s3` Perintah ini setara dengan AWS layanan lain yang menyediakan akses tingkat API ke fungsionalitas layanan. Untuk informasi selengkapnya tentang `s3` perintah, lihat [Gunakan perintah tingkat tinggi \(s3\) dengan AWS CLI](#)

Topik ini memberikan contoh yang menunjukkan cara menggunakan perintah tingkat rendah yang dipetakan ke API Amazon S3. Selain itu, Anda dapat menemukan contoh untuk setiap perintah API S3 di `s3api` bagian [referensi](#).

Topik

- [Prasyarat](#)
- [Menerapkan ACL khusus](#)
- [Konfigurasi kebijakan pencatatan](#)
- [Sumber daya](#)

## Prasyarat

Untuk menjalankan `s3api` perintah, Anda perlu:

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, silakan lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Profil yang Anda gunakan harus memiliki izin yang memungkinkan AWS operasi dilakukan oleh contoh.
- Pahami istilah Amazon S3 ini:
  - Bucket - Folder Amazon S3 tingkat atas.
  - Awalan - Folder Amazon S3 dalam ember.
  - Objek - Item apa pun yang di-host di ember Amazon S3.

## Menerapkan ACL khusus

Dengan perintah tingkat tinggi, Anda dapat menggunakan `--acl` opsi untuk menerapkan daftar kontrol akses (ACL) yang telah ditentukan sebelumnya ke objek Amazon S3. Tetapi Anda tidak dapat menggunakan perintah itu untuk mengatur ACL di seluruh ember. Namun, Anda dapat melakukan ini dengan menggunakan perintah [put-bucket-acl](#) API-level.

Contoh berikut menunjukkan cara memberikan kontrol penuh kepada dua AWS pengguna (`user1@example.com` dan `user2@example.com`) dan izin baca untuk semua orang. Pengidentifikasi untuk “semua orang” berasal dari URI khusus yang Anda lewati sebagai parameter.

```
$ aws s3api put-bucket-acl --bucket MyBucket --grant-full-control  
'emailaddress="user1@example.com",emailaddress="user2@example.com"' --grant-read  
'uri="http://acs.amazonaws.com/groups/global/AllUsers"'
```

Untuk detail tentang cara membuat ACL, lihat [PUT Bucket acl](#) di Referensi API Amazon Simple Storage Service. Perintah `s3api` ACL di CLI, `put-bucket-acl` seperti, menggunakan notasi argumen [singkatan yang sama](#).

## Konfigurasi kebijakan pencatatan

Perintah API `put-bucket-logging` mengonfigurasi kebijakan pencatatan bucket.

Dalam contoh berikut, AWS pengguna `user@example.com` diberikan kontrol penuh atas file log, dan semua pengguna telah membaca akses ke file tersebut. Perhatikan bahwa `put-bucket-acl`



perintah juga diperlukan untuk memberikan sistem pengiriman log Amazon S3 (ditentukan oleh URI) izin yang diperlukan untuk membaca dan menulis log ke bucket.

```
$ aws s3api put-bucket-acl --bucket MyBucket --grant-read-acp 'URI="http://acs.amazonaws.com/groups/s3/LogDelivery"' --grant-write 'URI="http://acs.amazonaws.com/groups/s3/LogDelivery"'
$ aws s3api put-bucket-logging --bucket MyBucket --bucket-logging-status file://logging.json
```

logging.jsonFile dalam perintah sebelumnya memiliki konten berikut.

```
{
  "LoggingEnabled": {
    "TargetBucket": "MyBucket",
    "TargetPrefix": "MyBucketLogs/",
    "TargetGrants": [
      {
        "Grantee": {
          "Type": "AmazonCustomerByEmail",
          "EmailAddress": "user@example.com"
        },
        "Permission": "FULL_CONTROL"
      },
      {
        "Grantee": {
          "Type": "Group",
          "URI": "http://acs.amazonaws.com/groups/global/AllUsers"
        },
        "Permission": "READ"
      }
    ]
  }
}
```

Sumber daya

AWS CLI referensi:

- [aws s3api](#)
- [aws s3api put-bucket-acl](#)
- [aws s3api put-bucket-logging](#)

## Referensi layanan:

- [Bekerja dengan bucket Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon
- [Bekerja dengan objek Amazon S3 di Panduan](#) Pengguna Layanan Penyimpanan Sederhana Amazon
- [Kunci daftar secara hierarkis menggunakan awalan dan pembatas dalam Panduan Pengguna](#) Layanan Penyimpanan Sederhana Amazon
- [Batalkan unggahan multibagian ke bucket S3 menggunakan AWS SDK for .NET \(tingkat rendah\)](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon

## Contoh skrip operasi siklus hidup bucket Amazon S3

Topik ini menggunakan contoh skrip bash untuk operasi siklus hidup bucket Amazon S3 menggunakan (). AWS Command Line Interface AWS CLI Contoh scripting ini menggunakan set perintah. Skrip Shell adalah program yang dirancang untuk berjalan di antarmuka baris perintah.

### Topik

- [Sebelum Anda mulai](#)
- [Tentang contoh ini](#)
- [Berkas](#)
- [References](#)

### Sebelum Anda mulai

Sebelum Anda dapat menjalankan salah satu contoh di bawah ini, hal-hal berikut perlu diselesaikan.

- Instal dan konfigurasi AWS CLI. Untuk informasi selengkapnya, silakan lihat [the section called “Instal/Perbarui”](#) dan [Otentikasi dan akses kredensial](#).
- Profil yang Anda gunakan harus memiliki izin yang memungkinkan AWS operasi dilakukan oleh contoh.
- Sebagai praktik AWS terbaik, berikan kode ini hak istimewa paling sedikit, atau hanya izin yang diperlukan untuk melakukan tugas. Untuk informasi lebih lanjut, lihat [Berikan hak istimewa terkecil](#) dalam Panduan Pengguna IAM.

- Kode ini belum diuji di semua AWS Wilayah. Beberapa AWS layanan hanya tersedia di Wilayah tertentu. Untuk informasi selengkapnya, lihat [Titik Akhir Layanan dan Kuota di Panduan Referensi AWS Umum](#).
- Menjalankan kode ini dapat mengakibatkan biaya ke AWS akun Anda. Adalah tanggung jawab Anda untuk memastikan bahwa sumber daya apa pun yang dibuat oleh skrip ini dihapus ketika Anda selesai dengan mereka.

Layanan Amazon S3 menggunakan istilah berikut:

- Bucket - Folder Amazon S3 tingkat atas.
- Awalan - Folder Amazon S3 dalam ember.
- Objek - Item apa pun yang dihosting di ember Amazon S3.

Tentang contoh ini

Contoh ini menunjukkan bagaimana berinteraksi dengan beberapa operasi Amazon S3 dasar menggunakan serangkaian fungsi dalam file skrip shell. Fungsinya terletak di file skrip shell bernama `bucket-operations.sh`. Anda dapat memanggil fungsi-fungsi ini di file lain. Setiap file skrip berisi komentar yang menjelaskan masing-masing fungsi.

Untuk melihat hasil antara setiap langkah, jalankan skrip dengan `-i` parameter. Anda dapat melihat status bucket saat ini atau isinya menggunakan konsol Amazon S3. Script hanya melanjutkan ke langkah berikutnya ketika Anda menekan enter pada prompt.

Untuk contoh lengkap dan file skrip yang dapat diunduh, lihat Operasi [Siklus Hidup Bucket Amazon S3 di](#) Repositori Contoh KodeAWS. GitHub

Berkas

Contoh berisi file-file berikut:

`bucket-operations.sh`

File skrip utama ini dapat bersumber dari file lain. Ini termasuk fungsi yang melakukan tugas-tugas berikut:

- Membuat ember dan memverifikasi bahwa itu ada
- Menyalin file dari komputer lokal ke ember
- Menyalin file dari satu lokasi bucket ke lokasi bucket yang berbeda

- Daftar isi ember
- Menghapus file dari ember
- Menghapus bucket

Lihat kode untuk [bucket-operations.sh](#) aktif GitHub.

#### test-bucket-operations.sh

File skrip shell `test-bucket-operations.sh` menunjukkan cara memanggil fungsi dengan mencari `bucket-operations.sh` file dan memanggil masing-masing fungsi. Setelah memanggil fungsi, skrip pengujian menghapus semua sumber daya yang dibuatnya.

Lihat kode untuk [test-bucket-operations.sh](#) aktif GitHub.

#### awsdocs-general.sh

File skrip `awsdocs-general.sh` menyimpan fungsi tujuan umum yang digunakan di seluruh contoh kode lanjutan untuk file AWS CLI.

Lihat kode untuk [awsdocs-general.sh](#) aktif GitHub.

## References

### AWS CLI referensi:

- [aws s3api](#)
- [aws s3api create-bucket](#)
- [aws s3api copy-object](#)
- [aws s3api delete-bucket](#)
- [aws s3api delete-object](#)
- [aws s3api head-bucket](#)
- [aws s3api list-objects](#)
- [aws s3api put-object](#)

### Referensi lainnya:

- [Bekerja dengan bucket Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon
- [Bekerja dengan objek Amazon S3 di Panduan](#) Pengguna Layanan Penyimpanan Sederhana Amazon
- Untuk melihat dan berkontribusi pada AWS SDK dan contoh AWS CLI kode, lihat [Repositori Contoh AWS Kode](#) di. GitHub

## Gunakan Amazon SNS dengan AWS CLI

Anda dapat mengakses fitur Amazon Simple Notification Service (Amazon SNS) menggunakan AWS Command Line Interface (). AWS CLI Untuk membuat daftar AWS CLI perintah untuk Amazon SNS, gunakan perintah berikut.

```
aws sns help
```

Sebelum Anda menjalankan perintah apa pun, atur kredensi default Anda. Untuk informasi selengkapnya, lihat [Konfigurasi AWS CLI](#).

Topik ini menunjukkan contoh AWS CLI perintah yang melakukan tugas umum untuk Amazon SNS.

Topik

- [Buat topik](#)
- [Berlangganan topik](#)
- [Publikasikan ke topik](#)
- [Berhenti berlangganan dari suatu topik](#)
- [Hapus topik](#)

### Buat topik

Untuk membuat topik, gunakan perintah dan tentukan nama yang akan ditetapkan ke topik.

```
$ aws sns create-topic --name my-topic
{
  "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
}
```

Catat respons `TopicArn`, yang Anda gunakan nanti untuk mempublikasikan pesan.

## Berlangganan topik

Untuk berlangganan topik, gunakan perintah.

Contoh berikut menentukan email protokol dan alamat email untuk. `notification-endpoint`

```
$ aws sns subscribe --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic --
protocol email --notification-endpoint saanvi@example.com
{
  "SubscriptionArn": "pending confirmation"
}
```

AWS segera mengirim pesan konfirmasi melalui email ke alamat yang Anda tentukan dalam `subscribe` perintah. Pesan email memiliki teks berikut.

```
You have chosen to subscribe to the topic:
arn:aws:sns:us-west-2:123456789012:my-topic
To confirm this subscription, click or visit the following link (If this was in error
no action is necessary):
Confirm subscription
```

Setelah penerima mengklik tautan Konfirmasi langganan, browser penerima menampilkan pesan notifikasi dengan informasi yang mirip dengan yang berikut ini.

```
Subscription confirmed!

You have subscribed saanvi@example.com to the topic:my-topic.

Your subscription's id is:
arn:aws:sns:us-west-2:123456789012:my-topic:1328f057-de93-4c15-512e-8bb22EXAMPLE

If it was not your intention to subscribe, click here to unsubscribe.
```

## Publikasikan ke topik

Untuk mengirim pesan ke semua pelanggan suatu topik, gunakan perintah.

Contoh berikut mengirimkan pesan "Hello World!" untuk semua pelanggan dari topik yang ditentukan.

```
$ aws sns publish --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic --
message "Hello World!"
```

```
{  
  "MessageId": "4e41661d-5eec-5ddf-8dab-2c867EXAMPLE"  
}
```

Dalam contoh ini, AWS kirim pesan email dengan teks “Hello World!” ke `ksaanvi@example.com`.

## Berhenti berlangganan dari suatu topik

Untuk berhenti berlangganan dari topik dan berhenti menerima pesan yang dipublikasikan ke topik itu, gunakan perintah dan tentukan ARN dari topik yang ingin Anda hentikan berlangganan.

```
$ aws sns unsubscribe --subscription-arn arn:aws:sns:us-west-2:123456789012:my-  
topic:1328f057-de93-4c15-512e-8bb22EXAMPLE
```

Untuk memverifikasi bahwa Anda berhasil berhenti berlangganan, gunakan perintah untuk mengonfirmasi bahwa ARN tidak lagi muncul dalam daftar.

```
$ aws sns list-subscriptions
```

## Hapus topik

Untuk menghapus topik, jalankan perintah.

```
$ aws sns delete-topic --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic
```

Untuk memverifikasi bahwa topik AWS berhasil dihapus, gunakan perintah untuk mengonfirmasi bahwa topik tidak lagi muncul dalam daftar.

```
$ aws sns list-topics
```

## AWS CLI contoh perintah

Contoh kode dalam topik ini menunjukkan cara menggunakan AWS Command Line Interface with AWS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Contoh lintas layanan adalah contoh aplikasi yang bekerja di beberapa Layanan AWS.

Contoh

- [Tindakan dan skenario menggunakan AWS CLI](#)

## Tindakan dan skenario menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Layanan AWS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Layanan

- [Contoh ACM menggunakan AWS CLI](#)
- [Contoh API Gateway menggunakan AWS CLI](#)
- [API Gateway HTTP dan contoh WebSocket API menggunakan AWS CLI](#)
- [Contoh API Gateway Management API menggunakan AWS CLI](#)
- [Contoh App Mesh menggunakan AWS CLI](#)
- [Contoh App Runner menggunakan AWS CLI](#)
- [AWS AppConfig contoh menggunakan AWS CLI](#)
- [Contoh Application Auto Scaling menggunakan AWS CLI](#)
- [Contoh Application Discovery Service menggunakan AWS CLI](#)
- [AppRegistry contoh menggunakan AWS CLI](#)
- [Athena contoh menggunakan AWS CLI](#)
- [Contoh Auto Scaling menggunakan AWS CLI](#)
- [Contoh Paket Auto Scaling menggunakan AWS CLI](#)
- [AWS Backup contoh menggunakan AWS CLI](#)



- [AWS Batch contoh menggunakan AWS CLI](#)
- [AWS Budgets contoh menggunakan AWS CLI](#)
- [Contoh Amazon Chime menggunakan AWS CLI](#)
- [Contoh Cloud Control API menggunakan AWS CLI](#)
- [AWS Cloud Map contoh menggunakan AWS CLI](#)
- [AWS Cloud9 contoh menggunakan AWS CLI](#)
- [AWS CloudFormation contoh menggunakan AWS CLI](#)
- [CloudFront contoh menggunakan AWS CLI](#)
- [CloudSearch Contoh Amazon menggunakan AWS CLI](#)
- [CloudTrail contoh menggunakan AWS CLI](#)
- [CloudWatch contoh menggunakan AWS CLI](#)
- [CloudWatch Log contoh menggunakan AWS CLI](#)
- [CloudWatch Contoh Pemantauan Jaringan menggunakan AWS CLI](#)
- [CodeArtifact contoh menggunakan AWS CLI](#)
- [CodeBuild contoh menggunakan AWS CLI](#)
- [CodeCommit contoh menggunakan AWS CLI](#)
- [CodeDeploy contoh menggunakan AWS CLI](#)
- [CodeGuru Contoh reviewer menggunakan AWS CLI](#)
- [CodePipeline contoh menggunakan AWS CLI](#)
- [AWS CodeStar contoh menggunakan AWS CLI](#)
- [AWS CodeStar Contoh pemberitahuan menggunakan AWS CLI](#)
- [CodeConnections contoh menggunakan AWS CLI](#)
- [Contoh Identitas Amazon Cognito menggunakan AWS CLI](#)
- [Contoh Penyedia Identitas Amazon Cognito menggunakan AWS CLI](#)
- [Amazon Comprehend contoh menggunakan AWS CLI](#)
- [Amazon Comprehend Medical contoh menggunakan AWS CLI](#)
- [AWS Config contoh menggunakan AWS CLI](#)
- [Contoh Amazon Connect menggunakan AWS CLI](#)
- [AWS Cost and Usage Report contoh menggunakan AWS CLI](#)
- [Contoh Layanan Cost Explorer menggunakan AWS CLI](#)

- [Contoh Firehose menggunakan AWS CLI](#)
- [Contoh Amazon Data Lifecycle Manager menggunakan AWS CLI](#)
- [AWS Data Pipeline contoh menggunakan AWS CLI](#)
- [DataSync contoh menggunakan AWS CLI](#)
- [Contoh DAX menggunakan AWS CLI](#)
- [Contoh Detektif menggunakan AWS CLI](#)
- [Contoh Device Farm menggunakan AWS CLI](#)
- [AWS Direct Connect contoh menggunakan AWS CLI](#)
- [AWS Directory Service contoh menggunakan AWS CLI](#)
- [AWS DMS contoh menggunakan AWS CLI](#)
- [Amazon DocumentDB contoh menggunakan AWS CLI](#)
- [Contoh DynamoDB menggunakan AWS CLI](#)
- [Contoh DynamoDB Streams menggunakan AWS CLI](#)
- [Contoh Amazon EC2 menggunakan AWS CLI](#)
- [Contoh Amazon EC2 Instance Connect menggunakan AWS CLI](#)
- [Contoh Amazon ECR menggunakan AWS CLI](#)
- [Contoh Amazon ECS menggunakan AWS CLI](#)
- [Amazon EFS contoh menggunakan AWS CLI](#)
- [Amazon EKS contoh menggunakan AWS CLI](#)
- [Contoh Elastic Beanstalk menggunakan AWS CLI](#)
- [Elastic Load Balancing - Contoh versi 1 menggunakan AWS CLI](#)
- [Elastic Load Balancing - Contoh versi 2 menggunakan AWS CLI](#)
- [Contoh Elastic Transcoder menggunakan AWS CLI](#)
- [ElastiCache contoh menggunakan AWS CLI](#)
- [MediaStore contoh menggunakan AWS CLI](#)
- [Amazon EMR contoh menggunakan AWS CLI](#)
- [Amazon EMR pada contoh EKS menggunakan AWS CLI](#)
- [EventBridge contoh menggunakan AWS CLI](#)
- [Contoh Firewall Manager menggunakan AWS CLI](#)
- [AWS FIS contoh menggunakan AWS CLI](#)

- [GameLift Contoh Amazon menggunakan AWS CLI](#)
- [Contoh Global Accelerator menggunakan AWS CLI](#)
- [AWS Glue contoh menggunakan AWS CLI](#)
- [GuardDuty contoh menggunakan AWS CLI](#)
- [AWS Health contoh menggunakan AWS CLI](#)
- [HealthImaging contoh menggunakan AWS CLI](#)
- [HealthLake contoh menggunakan AWS CLI](#)
- [HealthOmics contoh menggunakan AWS CLI](#)
- [Contoh IAM menggunakan AWS CLI](#)
- [Contoh IAM Access Analyzer menggunakan AWS CLI](#)
- [Contoh Image Builder menggunakan AWS CLI](#)
- [Contoh Manajer Insiden menggunakan AWS CLI](#)
- [Contoh Kontak Manajer Insiden menggunakan AWS CLI](#)
- [Contoh Amazon Inspector menggunakan AWS CLI](#)
- [AWS IoT contoh menggunakan AWS CLI](#)
- [AWS IoT 1-Click Contoh perangkat menggunakan AWS CLI](#)
- [AWS IoT 1-Click Contoh proyek menggunakan AWS CLI](#)
- [AWS IoT Analytics contoh menggunakan AWS CLI](#)
- [Contoh Device Advisor menggunakan AWS CLI](#)
- [AWS IoT data contoh menggunakan AWS CLI](#)
- [AWS IoT Events contoh menggunakan AWS CLI](#)
- [AWS IoT Events-Data contoh menggunakan AWS CLI](#)
- [AWS IoT Greengrass contoh menggunakan AWS CLI](#)
- [AWS IoT Greengrass V2 contoh menggunakan AWS CLI](#)
- [AWS IoT Jobs SDK release contoh menggunakan AWS CLI](#)
- [AWS IoT SiteWise contoh menggunakan AWS CLI](#)
- [AWS IoT Things Graph contoh menggunakan AWS CLI](#)
- [AWS IoT Wireless contoh menggunakan AWS CLI](#)
- [Contoh Amazon IVS menggunakan AWS CLI](#)
- [Contoh Obrolan Amazon IVS menggunakan AWS CLI](#)

- [Contoh Streaming Waktu Nyata Amazon IVS menggunakan AWS CLI](#)
- [Amazon Kendra contoh menggunakan AWS CLI](#)
- [Contoh Kinesis menggunakan AWS CLI](#)
- [AWS KMS contoh menggunakan AWS CLI](#)
- [Contoh Lake Formation menggunakan AWS CLI](#)
- [Contoh Lambda menggunakan AWS CLI](#)
- [Contoh License Manager menggunakan AWS CLI](#)
- [Contoh Lightsail menggunakan AWS CLI](#)
- [Contoh Macie menggunakan AWS CLI](#)
- [Contoh Grafana yang Dikelola Amazon menggunakan AWS CLI](#)
- [MediaConnect contoh menggunakan AWS CLI](#)
- [MediaConvert contoh menggunakan AWS CLI](#)
- [MediaLive contoh menggunakan AWS CLI](#)
- [MediaPackage contoh menggunakan AWS CLI](#)
- [MediaPackage Contoh VOD menggunakan AWS CLI](#)
- [MediaStore Contoh Data Plane menggunakan AWS CLI](#)
- [MediaTailor contoh menggunakan AWS CLI](#)
- [Contoh MemoryDB menggunakan AWS CLI](#)
- [Amazon MSK contoh menggunakan AWS CLI](#)
- [Contoh Network Manager menggunakan AWS CLI](#)
- [Contoh Nimble Studio menggunakan AWS CLI](#)
- [OpenSearch Contoh layanan menggunakan AWS CLI](#)
- [AWS OpsWorks contoh menggunakan AWS CLI](#)
- [AWS OpsWorks CM contoh menggunakan AWS CLI](#)
- [Organizations contoh menggunakan AWS CLI](#)
- [AWS Outposts contoh menggunakan AWS CLI](#)
- [AWS Payment Cryptography contoh menggunakan AWS CLI](#)
- [AWS Payment Cryptography Contoh Data Plane menggunakan AWS CLI](#)
- [Amazon Pinpoint contoh menggunakan AWS CLI](#)
- [Contoh Amazon Polly menggunakan AWS CLI](#)

- [Daftar Harga AWS contoh menggunakan AWS CLI](#)
- [AWS Private CA contoh menggunakan AWS CLI](#)
- [AWS Proton contoh menggunakan AWS CLI](#)
- [Contoh QLDB menggunakan AWS CLI](#)
- [Contoh Amazon RDS menggunakan AWS CLI](#)
- [Contoh Layanan Data Amazon RDS menggunakan AWS CLI](#)
- [Contoh Performance Insights Amazon RDS menggunakan AWS CLI](#)
- [Contoh Amazon Redshift menggunakan AWS CLI](#)
- [Contoh Rekognition Amazon menggunakan AWS CLI](#)
- [AWS RAM contoh menggunakan AWS CLI](#)
- [Contoh Resource Explorer menggunakan AWS CLI](#)
- [Contoh Resource Groups menggunakan AWS CLI](#)
- [Contoh API Penandaan Resource Groups menggunakan AWS CLI](#)
- [AWS RoboMaker contoh menggunakan AWS CLI](#)
- [Route 53 contoh menggunakan AWS CLI](#)
- [Route 53 contoh pendaftaran domain menggunakan AWS CLI](#)
- [Contoh Route 53 Resolver menggunakan AWS CLI](#)
- [Contoh Amazon S3 menggunakan AWS CLI](#)
- [Contoh Kontrol Amazon S3 menggunakan AWS CLI](#)
- [Contoh S3 Glacier menggunakan AWS CLI](#)
- [Secrets Manager contoh menggunakan AWS CLI](#)
- [Contoh Security Hub menggunakan AWS CLI](#)
- [AWS Serverless Application Repository contoh menggunakan AWS CLI](#)
- [Contoh Service Catalog menggunakan AWS CLI](#)
- [Contoh Service Quotas menggunakan AWS CLI](#)
- [Amazon SES contoh menggunakan AWS CLI](#)
- [Contoh Shield menggunakan AWS CLI](#)
- [Contoh penandatanganan menggunakan AWS CLI](#)
- [Contoh Snowball menggunakan AWS CLI](#)
- [Contoh Amazon SNS menggunakan AWS CLI](#)

- [Contoh Amazon SQS menggunakan AWS CLI](#)
- [Contoh Storage Gateway menggunakan AWS CLI](#)
- [AWS STS contoh menggunakan AWS CLI](#)
- [AWS Support contoh menggunakan AWS CLI](#)
- [Contoh Amazon SWF menggunakan AWS CLI](#)
- [Contoh Systems Manager menggunakan AWS CLI](#)
- [Contoh Amazon Texttract menggunakan AWS CLI](#)
- [Contoh Amazon Transcribe menggunakan AWS CLI](#)
- [Contoh Amazon Translate menggunakan AWS CLI](#)
- [Trusted Advisor contoh menggunakan AWS CLI](#)
- [Contoh Izin Terverifikasi menggunakan AWS CLI](#)
- [Contoh VPC Lattice menggunakan AWS CLI](#)
- [AWS WAF Classic contoh menggunakan AWS CLI](#)
- [AWS WAF Classic Regional contoh menggunakan AWS CLI](#)
- [AWS WAFV2 contoh menggunakan AWS CLI](#)
- [WorkDocs Contoh Amazon menggunakan AWS CLI](#)
- [WorkMail Contoh Amazon menggunakan AWS CLI](#)
- [Contoh Amazon WorkMail Message Flow menggunakan AWS CLI](#)
- [WorkSpaces contoh menggunakan AWS CLI](#)
- [Contoh X-Ray menggunakan AWS CLI](#)

## Contoh ACM menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan ACM AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags-to-certificate**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-certificate`.

AWS CLI

Untuk menambahkan tag ke Sertifikat ACM yang ada

`add-tags-to-certificate` Perintah berikut menambahkan dua tag ke sertifikat yang ditentukan. Gunakan spasi untuk memisahkan beberapa tag:

```
aws acm add-tags-to-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --tags
Key=Admin,Value=Alice Key=Purpose,Value=Website
```

- Untuk detail API, lihat [AddTagsToCertificate](#) di Referensi AWS CLI Perintah.

### **delete-certificate**

Contoh kode berikut menunjukkan cara menggunakan `delete-certificate`.

AWS CLI

Untuk menghapus sertifikat ACM dari akun Anda

`delete-certificate` Perintah berikut menghapus sertifikat dengan ARN yang ditentukan:

```
aws acm delete-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

- Untuk detail API, lihat [DeleteCertificate](#) di Referensi AWS CLI Perintah.

## describe-certificate

Contoh kode berikut menunjukkan cara menggunakan `describe-certificate`.

### AWS CLI

Untuk mengambil bidang yang terdapat dalam sertifikat ACM

`describe-certificate` Perintah berikut mengambil semua bidang untuk sertifikat dengan ARN yang ditentukan:

```
aws acm describe-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Output yang mirip dengan berikut ini ditampilkan:

```
{
  "Certificate": {
    "CertificateArn":
"arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
    "CreatedAt": 1446835267.0,
    "DomainName": "www.example.com",
    "DomainValidationOptions": [
      {
        "DomainName": "www.example.com",
        "ValidationDomain": "www.example.com",
        "ValidationEmails": [
          "hostmaster@example.com",
          "admin@example.com",
          "owner@example.com.whoisprivacyservice.org",
          "tech@example.com.whoisprivacyservice.org",
          "admin@example.com.whoisprivacyservice.org",
          "postmaster@example.com",
          "webmaster@example.com",
          "administrator@example.com"
        ]
      },
      {
        "DomainName": "www.example.net",
        "ValidationDomain": "www.example.net",
        "ValidationEmails": [
          "postmaster@example.net",
          "admin@example.net",

```



```

    "owner@example.net.whoisprivacyservice.org",
    "tech@example.net.whoisprivacyservice.org",
    "admin@example.net.whoisprivacyservice.org",
    "hostmaster@example.net",
    "administrator@example.net",
    "webmaster@example.net"
  ]
}
],
"InUseBy": [],
"IssuedAt": 1446835815.0,
"Issuer": "Amazon",
"KeyAlgorithm": "RSA-2048",
"NotAfter": 1478433600.0,
"NotBefore": 1446768000.0,
"Serial": "0f:ac:b0:a3:8d:ea:65:52:2d:7d:01:3a:39:36:db:d6",
"SignatureAlgorithm": "SHA256WITHRSA",
"Status": "ISSUED",
"Subject": "CN=www.example.com",
"SubjectAlternativeNames": [
  "www.example.com",
  "www.example.net"
]
}
}

```

- Untuk detail API, lihat [DescribeCertificate](#) di Referensi AWS CLI Perintah.

## export-certificate

Contoh kode berikut menunjukkan cara menggunakan `export-certificate`.

### AWS CLI

Untuk mengekspor sertifikat pribadi yang dikeluarkan oleh CA pribadi.

`export-certificate` Perintah berikut mengekspor sertifikat pribadi, rantai sertifikat, dan kunci pribadi ke layar Anda:

```

aws acm export-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
passphrase file://path-to-passphrase-file

```

Untuk mengekspor sertifikat, rantai, dan kunci pribadi ke file lokal, gunakan perintah berikut:

```
aws acm export-certificate --certificate-arn
arn:aws:acm:region:sccount:certificate/12345678-1234-1234-1234-123456789012 --
passphrase file://path-to-passphrase-file > c:\temp\export.txt
```

- Untuk detail API, lihat [ExportCertificate](#) di Referensi AWS CLI Perintah.

## get-certificate

Contoh kode berikut menunjukkan cara menggunakan `get-certificate`.

### AWS CLI

Untuk mengambil sertifikat ACM

`get-certificate` Perintah berikut mengambil sertifikat untuk ARN yang ditentukan dan rantai sertifikat:

```
aws acm get-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Output yang mirip dengan berikut ini ditampilkan:

```
{
  "Certificate": "-----BEGIN CERTIFICATE-----
MIICiTCCAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGFT
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZncvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
```

```

"CertificateChain": "-----BEGIN CERTIFICATE-----
MIICiTCCAFICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJiI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
MIICiTCCAFICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJiI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
MIICiTCCAFICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4

```

```
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----"
}
```

- Untuk detail API, lihat [GetCertificate](#) di Referensi AWS CLI Perintah.

## import-certificate

Contoh kode berikut menunjukkan cara menggunakan `import-certificate`.

### AWS CLI

Untuk mengimpor sertifikat ke ACM.

`import-certificate` Perintah berikut mengimpor sertifikat ke ACM. Ganti nama file dengan nama Anda sendiri:

```
aws acm import-certificate --certificate file://Certificate.pem --certificate-chain
file://CertificateChain.pem --private-key file://PrivateKey.pem
```

- Untuk detail API, lihat [ImportCertificate](#) di Referensi AWS CLI Perintah.

## list-certificates

Contoh kode berikut menunjukkan cara menggunakan `list-certificates`.

### AWS CLI

Untuk mencantumkan sertifikat ACM untuk akun AWS

`list-certificates` Perintah berikut mencantumkan ARN sertifikat di akun Anda:

```
aws acm list-certificates
```

Perintah sebelumnya menghasilkan output yang mirip dengan berikut ini:

```
{
  "CertificateSummaryList": [
```

```

    {
      "CertificateArn":
"arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
      "DomainName": "www.example.com"
    },
    {
      "CertificateArn": "arn:aws:acm:region:account:certificate/aaaaaaaa-bbbb-
cccc-dddd-eeeeeeeeeeee",
      "DomainName": "www.example.net"
    }
  ]
}

```

Anda dapat memutuskan berapa banyak sertifikat yang ingin Anda tampilkan setiap kali Anda menelepon `list-certificates`. Misalnya, jika Anda memiliki empat sertifikat dan Anda ingin menampilkan tidak lebih dari dua sekaligus, atur `max-items` argumen ke 2 seperti pada contoh berikut:

```
aws acm list-certificates --max-items 2
```

Dua ARN sertifikat dan `NextToken` nilai akan ditampilkan:

```

"CertificateSummaryList": [
  {
    "CertificateArn": "arn:aws:acm:region:account: \
      certificate/12345678-1234-1234-1234-123456789012",
    "DomainName": "www.example.com"
  },
  {
    "CertificateArn": "arn:aws:acm:region:account: \
      certificate/aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee",
    "DomainName": "www.example.net"
  }
],
"NextToken": "9f4d9f69-275a-41fe-b58e-2b837bd9ba48"

```

Untuk menampilkan dua sertifikat berikutnya di akun Anda, tetapkan `NextToken` nilai ini dalam panggilan berikutnya:

```
aws acm list-certificates --max-items 2 --next-token 9f4d9f69-275a-41fe-
b58e-2b837bd9ba48
```

Anda dapat memfilter output Anda dengan menggunakan `certificate-statuses` argumen. Perintah berikut menampilkan sertifikat yang memiliki status `PENDING_VALIDATION`:

```
aws acm list-certificates --certificate-statuses PENDING_VALIDATION
```

Anda juga dapat memfilter output Anda dengan menggunakan `includes` argumen. Perintah berikut menampilkan sertifikat yang difilter pada properti berikut. Sertifikat yang akan ditampilkan:

- Specify that the RSA algorithm and a 2048 bit key are used to generate key pairs.
- Contain a Key Usage extension that specifies that the certificates can be used to create digital signatures.
- Contain an Extended Key Usage extension that specifies that the certificates can be used for code signing.

```
aws acm list-certificates --max-items 10 --includes  
extendedKeyUsage=CODE_SIGNING,keyUsage=DIGITAL_SIGNATURE,keyTypes=RSA_2048
```

- Untuk detail API, lihat [ListCertificates](#) di Referensi AWS CLI Perintah.

## list-tags-for-certificate

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-certificate`.

### AWS CLI

Untuk mencantumkan tag yang diterapkan pada Sertifikat ACM

`list-tags-for-certificate` Perintah berikut mencantumkan tag yang diterapkan pada sertifikat di akun Anda:

```
aws acm list-tags-for-certificate --certificate-arn  
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Perintah sebelumnya menghasilkan output yang mirip dengan berikut ini:

```
{  
  "Tags": [  
    {  
      "Value": "Website",  
      "Key": "Purpose"  
    },  
  ],  
}
```

```
{
  "Value": "Alice",
  "Key": "Admin"
}
]
```

- Untuk detail API, lihat [ListTagsForCertificate](#) di Referensi AWS CLI Perintah.

## **remove-tags-from-certificate**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-certificate`.

### AWS CLI

Untuk menghapus tag dari Sertifikat ACM

`remove-tags-from-certificate` Perintah berikut menghapus dua tag dari sertifikat yang ditentukan. Gunakan spasi untuk memisahkan beberapa tag:

```
aws acm remove-tags-from-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --tags
Key=Admin,Value=Alice Key=Purpose,Value=Website
```

- Untuk detail API, lihat [RemoveTagsFromCertificate](#) di Referensi AWS CLI Perintah.

## **request-certificate**

Contoh kode berikut menunjukkan cara menggunakan `request-certificate`.

### AWS CLI

Untuk meminta sertifikat ACM baru

`request-certificate` Perintah berikut meminta sertifikat baru untuk domain `www.example.com` menggunakan validasi DNS:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS
```

Anda dapat memasukkan token idempotensi untuk membedakan antara panggilan ke: `request-certificate`

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS --  
idempotency-token 91adc45q
```

Anda dapat memasukkan satu atau beberapa nama alternatif subjek untuk meminta sertifikat yang akan melindungi lebih dari satu domain puncak:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
idempotency-token 91adc45q --subject-alternative-names www.example.net
```

Anda dapat memasukkan nama alternatif yang juga dapat digunakan untuk menjangkau situs web Anda:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
idempotency-token 91adc45q --subject-alternative-names www.example.com
```

Anda dapat menggunakan tanda bintang (\*) sebagai wildcard untuk membuat sertifikat untuk beberapa subdomain dalam domain yang sama:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
idempotency-token 91adc45q --subject-alternative-names *.example.com
```

Anda juga dapat memasukkan beberapa nama alternatif:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
subject-alternative-names b.example.com c.example.com d.example.com
```

Jika Anda menggunakan email untuk validasi, Anda dapat memasukkan opsi validasi domain untuk menentukan domain tempat email validasi akan dikirim:

```
aws acm request-certificate --domain-name example.com --validation-method  
EMAIL --subject-alternative-names www.example.com --domain-validation-options  
DomainName=example.com,ValidationDomain=example.com
```

Perintah berikut memilih keluar dari pencatatan transparansi sertifikat saat Anda meminta sertifikat baru:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS --  
options CertificateTransparencyLoggingPreference=DISABLED --idempotency-token 184627
```



- Untuk detail API, lihat [RequestCertificate](#) di Referensi AWS CLI Perintah.

## resend-validation-email

Contoh kode berikut menunjukkan cara menggunakan `resend-validation-email`.

### AWS CLI

Untuk mengirim ulang email validasi untuk permintaan sertifikat ACM Anda

`resend-validation-email` Perintah berikut memberi tahu otoritas sertifikat Amazon untuk mengirim email validasi ke alamat yang sesuai:

```
aws acm resend-validation-email --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
domain www.example.com --validation-domain example.com
```

- Untuk detail API, lihat [ResendValidationEmail](#) di Referensi AWS CLI Perintah.

## update-certificate-options

Contoh kode berikut menunjukkan cara menggunakan `update-certificate-options`.

### AWS CLI

Untuk memperbarui opsi sertifikat

`update-certificate-options` Perintah berikut memilih keluar dari pencatatan transparansi sertifikat:

```
aws acm update-certificate-options --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
options CertificateTransparencyLoggingPreference=DISABLED
```

- Untuk detail API, lihat [UpdateCertificateOptions](#) di Referensi AWS CLI Perintah.

## Contoh API Gateway menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan API Gateway AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-api-key**

Contoh kode berikut menunjukkan cara menggunakan `create-api-key`.

AWS CLI

Untuk membuat kunci API yang diaktifkan untuk API dan Stage yang ada

Perintah:

```
aws apigateway create-api-key --name 'Dev API Key' --description 'Used for development' --enabled --stage-keys restApiId='a1b2c3d4e5',stageName='dev'
```

- Untuk detail API, lihat [CreateApiKey](#) di Referensi AWS CLI Perintah.

### **create-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `create-authorizer`.

AWS CLI

Contoh 1: Untuk membuat Otorisasi Kustom API Gateway berbasis token untuk API

`create-authorizer` Contoh berikut membuat authorizer berbasis token.

```
aws apigateway create-authorizer \
```

```

--rest-api-id 1234123412 \
--name 'First-Token-Custom-Authorizer' \
--type TOKEN \
--authorizer-uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations' \
--identity-source 'method.request.header.Authorization' \
--authorizer-result-ttl-in-seconds 300

```

**Output:**

```

{
  "authType": "custom",
  "name": "First-Token-Custom-Authorizer",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations",
  "authorizerResultTtlInSeconds": 300,
  "identitySource": "method.request.header.Authorization",
  "type": "TOKEN",
  "id": "z40xj0"
}

```

Contoh 2: Untuk membuat Otorisasi Kustom API Gateway berbasis Cognito User Pools untuk API `create-authorizer` Contoh berikut membuat Otorisasi Kustom API Gateway berbasis Cognito User Pools.

```

aws apigateway create-authorizer \
--rest-api-id 1234123412 \
--name 'First-Cognito-Custom-Authorizer' \
--type COGNITO_USER_POOLS \
--provider-arns 'arn:aws:cognito-idp:us-east-1:123412341234:userpool/us-
east-1_aWcZeQbuD' \
--identity-source 'method.request.header.Authorization'

```

**Output:**

```

{
  "authType": "cognito_user_pools",
  "identitySource": "method.request.header.Authorization",
  "name": "First-Cognito-Custom-Authorizer",
  "providerARNs": [
    "arn:aws:cognito-idp:us-east-1:342398297714:userpool/us-east-1_qWbZzQhzE"
  ]
}

```

```

    ],
    "type": "COGNITO_USER_POOLS",
    "id": "5yid1t"
  }

```

Contoh 3: Untuk membuat API Gateway Custom Authorizer berbasis permintaan untuk API `create-authorizer` Contoh berikut membuat authorizer berbasis permintaan.

```

aws apigateway create-authorizer \
  --rest-api-id 1234123412 \
  --name 'First_Request_Custom_Authorizer' \
  --type REQUEST \
  --authorizer-uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations' \
  --identity-source 'method.request.header.Authorization,context.accountId' \
  --authorizer-result-ttl-in-seconds 300

```

Output:

```

{
  "id": "z40xj0",
  "name": "First_Request_Custom_Authorizer",
  "type": "REQUEST",
  "authType": "custom",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations",
  "identitySource": "method.request.header.Authorization,context.accountId",
  "authorizerResultTtlInSeconds": 300
}

```

- Untuk detail API, lihat [CreateAuthorizer](#) di Referensi AWS CLI Perintah.

## create-base-path-mapping

Contoh kode berikut menunjukkan cara menggunakan `create-base-path-mapping`.

### AWS CLI

Untuk membuat pemetaan jalur dasar untuk nama domain kustom

Perintah:

```
aws apigateway create-base-path-mapping --domain-name subdomain.domain.tld --rest-api-id 1234123412 --stage prod --base-path v1
```

- Untuk detail API, lihat [CreateBasePathMapping](#) di Referensi AWS CLI Perintah.

## create-deployment

Contoh kode berikut menunjukkan cara menggunakan `create-deployment`.

### AWS CLI

Untuk menerapkan sumber daya yang dikonfigurasi untuk API ke Tahap baru

Perintah:

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --stage-description 'Development Stage' --description 'First deployment to the dev stage'
```

Untuk menerapkan sumber daya yang dikonfigurasi untuk API ke tahap yang ada

Perintah:

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --description 'Second deployment to the dev stage'
```

Untuk menerapkan sumber daya yang dikonfigurasi untuk API ke tahap yang ada dengan Variabel Tahap

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --description 'Penerapan ketiga ke tahap dev' --variable key='value', otherKey='otherValue'
```

- Untuk detail API, lihat [CreateDeployment](#) di Referensi AWS CLI Perintah.

## create-domain-name

Contoh kode berikut menunjukkan cara menggunakan `create-domain-name`.

### AWS CLI

Untuk membuat nama domain kustom

Perintah:

```
aws apigateway create-domain-name --domain-name 'my.domain.tld' --
certificate-name 'my.domain.tld cert' --certificate-arn 'arn:aws:acm:us-
east-1:012345678910:certificate/fb1b9770-a305-495d-aefb-27e5e101ff3'
```

- Untuk detail API, lihat [CreateDomainName](#) di Referensi AWS CLI Perintah.

## create-model

Contoh kode berikut menunjukkan cara menggunakan `create-model`.

### AWS CLI

Untuk membuat model untuk API

Perintah:

```
aws apigateway create-model --rest-api-id 1234123412 --name 'firstModel' --
description 'The First Model' --content-type 'application/json' --schema
'{ "$schema": "http://json-schema.org/draft-04/schema#", "title": "firstModel",
"type": "object", "properties": { "firstProperty" : { "type": "object",
"properties": { "key": { "type": "string" } } } } }'
```

Output:

```
{
  "contentType": "application/json",
  "description": "The First Model",
  "name": "firstModel",
  "id": "2rzg01",
  "schema": "{ \"$schema\": \"http://json-schema.org/draft-04/schema#\", \"title
\": \"firstModel\", \"type\": \"object\", \"properties\": { \"firstProperty
\": { \"type\": \"object\", \"properties\": { \"key\": { \"type\": \"string
\" } } } } }"
```

- Untuk detail API, lihat [CreateModel](#) di Referensi AWS CLI Perintah.

## create-resource

Contoh kode berikut menunjukkan cara menggunakan `create-resource`.

## AWS CLI

Untuk membuat sumber daya di API

Perintah:

```
aws apigateway create-resource --rest-api-id 1234123412 --parent-id a1b2c3 --path-part 'new-resource'
```

- Untuk detail API, lihat [CreateResource](#) di Referensi AWS CLI Perintah.

## create-rest-api

Contoh kode berikut menunjukkan cara menggunakan `create-rest-api`.

### AWS CLI

Untuk membuat API

Perintah:

```
aws apigateway create-rest-api --name 'My First API' --description 'This is my first API'
```

Untuk membuat API duplikat dari API yang ada

Perintah:

```
aws apigateway create-rest-api --name 'Copy of My First API' --description 'This is a copy of my first API' --clone-from 1234123412
```

- Untuk detail API, lihat [CreateRestApi](#) di Referensi AWS CLI Perintah.

## create-stage

Contoh kode berikut menunjukkan cara menggunakan `create-stage`.

### AWS CLI

Untuk membuat tahapan di API yang akan berisi penerapan yang ada

Perintah:

```
aws apigateway create-stage --rest-api-id 1234123412 --stage-name 'dev' --description 'Development stage' --deployment-id a1b2c3
```

Untuk membuat tahap dalam API yang akan berisi penerapan yang ada dan Variabel Tahap kustom

Perintah:

```
aws apigateway create-stage --rest-api-id 1234123412 --stage-name 'dev' --description 'Development stage' --deployment-id a1b2c3 --variables key='value',otherKey='otherValue'
```

- Untuk detail API, lihat [CreateStage](#) di Referensi AWS CLI Perintah.

## **create-usage-plan-key**

Contoh kode berikut menunjukkan cara menggunakan `create-usage-plan-key`.

AWS CLI

Kaitkan kunci API yang ada dengan Paket Penggunaan

Perintah:

```
aws apigateway create-usage-plan-key --usage-plan-id a1b2c3 --key-type "API_KEY" --key-id 4vq3yryqm5
```

- Untuk detail API, lihat [CreateUsagePlanKey](#) di Referensi AWS CLI Perintah.

## **create-usage-plan**

Contoh kode berikut menunjukkan cara menggunakan `create-usage-plan`.

AWS CLI

Untuk membuat paket penggunaan dengan batas throttle dan kuota yang disetel ulang pada awal bulan

Perintah:



```
aws apigateway create-usage-plan --name "New Usage Plan" --description "A new usage plan" --throttle burstLimit=10,rateLimit=5 --quota limit=500,offset=0,period=MONTH
```

- Untuk detail API, lihat [CreateUsagePlan](#) di Referensi AWS CLI Perintah.

## **delete-api-key**

Contoh kode berikut menunjukkan cara menggunakan `delete-api-key`.

### AWS CLI

Untuk menghapus kunci API

Perintah:

```
aws apigateway delete-api-key --api-key 8bk1k8b11k3sB38D9B310enyWT8c09B301kq0b1k
```

- Untuk detail API, lihat [DeleteApiKey](#) di Referensi AWS CLI Perintah.

## **delete-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `delete-authorizer`.

### AWS CLI

Untuk menghapus Authorizer Kustom di API

Perintah:

```
aws apigateway delete-authorizer --rest-api-id 1234123412 --authorizer-id 7gkfbo
```

- Untuk detail API, lihat [DeleteAuthorizer](#) di Referensi AWS CLI Perintah.

## **delete-base-path-mapping**

Contoh kode berikut menunjukkan cara menggunakan `delete-base-path-mapping`.

### AWS CLI

Untuk menghapus pemetaan jalur dasar untuk nama domain kustom

Perintah:

```
aws apigateway delete-base-path-mapping --domain-name 'api.domain.tld' --base-path 'dev'
```

- Untuk detail API, lihat [DeleteBasePathMapping](#) di Referensi AWS CLI Perintah.

## **delete-client-certificate**

Contoh kode berikut menunjukkan cara menggunakan `delete-client-certificate`.

AWS CLI

Untuk menghapus sertifikat klien

Perintah:

```
aws apigateway delete-client-certificate --client-certificate-id a1b2c3
```

- Untuk detail API, lihat [DeleteClientCertificate](#) di Referensi AWS CLI Perintah.

## **delete-deployment**

Contoh kode berikut menunjukkan cara menggunakan `delete-deployment`.

AWS CLI

Untuk menghapus penerapan di API

Perintah:

```
aws apigateway delete-deployment --rest-api-id 1234123412 --deployment-id a1b2c3
```

- Untuk detail API, lihat [DeleteDeployment](#) di Referensi AWS CLI Perintah.

## **delete-domain-name**

Contoh kode berikut menunjukkan cara menggunakan `delete-domain-name`.

## AWS CLI

Untuk menghapus nama domain kustom

Perintah:

```
aws apigateway delete-domain-name --domain-name 'api.domain.tld'
```

- Untuk detail API, lihat [DeleteDomainName](#) di Referensi AWS CLI Perintah.

## **delete-integration-response**

Contoh kode berikut menunjukkan cara menggunakan `delete-integration-response`.

### AWS CLI

Untuk menghapus respons integrasi untuk sumber daya, metode, dan kode status tertentu dalam API

Perintah:

```
aws apigateway delete-integration-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 200
```

- Untuk detail API, lihat [DeleteIntegrationResponse](#) di Referensi AWS CLI Perintah.

## **delete-integration**

Contoh kode berikut menunjukkan cara menggunakan `delete-integration`.

### AWS CLI

Untuk menghapus integrasi untuk sumber daya dan metode tertentu dalam API

Perintah:

```
aws apigateway delete-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET
```

- Untuk detail API, lihat [DeleteIntegration](#) di Referensi AWS CLI Perintah.

## **delete-method-response**

Contoh kode berikut menunjukkan cara menggunakan `delete-method-response`.

### AWS CLI

Untuk menghapus respons metode untuk sumber daya, metode, dan kode status yang diberikan dalam API

Perintah:

```
aws apigateway delete-method-response --rest-api-id 1234123412 --resource-id a1b2c3
--http-method GET --status-code 200
```

- Untuk detail API, lihat [DeleteMethodResponse](#) di Referensi AWS CLI Perintah.

## **delete-method**

Contoh kode berikut menunjukkan cara menggunakan `delete-method`.

### AWS CLI

Untuk menghapus metode untuk sumber daya yang diberikan dalam API

Perintah:

```
aws apigateway delete-method --rest-api-id 1234123412 --resource-id a1b2c3 --http-
method GET
```

- Untuk detail API, lihat [DeleteMethod](#) di Referensi AWS CLI Perintah.

## **delete-model**

Contoh kode berikut menunjukkan cara menggunakan `delete-model`.

### AWS CLI

Untuk menghapus model di API yang diberikan

Perintah:

```
aws apigateway delete-model --rest-api-id 1234123412 --model-name 'customModel'
```

- Untuk detail API, lihat [DeleteModel](#) di Referensi AWS CLI Perintah.

## **delete-resource**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource`.

### AWS CLI

Untuk menghapus sumber daya di API

Perintah:

```
aws apigateway delete-resource --rest-api-id 1234123412 --resource-id a1b2c3
```

- Untuk detail API, lihat [DeleteResource](#) di Referensi AWS CLI Perintah.

## **delete-rest-api**

Contoh kode berikut menunjukkan cara menggunakan `delete-rest-api`.

### AWS CLI

Untuk menghapus API

Perintah:

```
aws apigateway delete-rest-api --rest-api-id 1234123412
```

- Untuk detail API, lihat [DeleteRestApi](#) di Referensi AWS CLI Perintah.

## **delete-stage**

Contoh kode berikut menunjukkan cara menggunakan `delete-stage`.

### AWS CLI

Untuk menghapus tahapan di API

Perintah:

```
aws apigateway delete-stage --rest-api-id 1234123412 --stage-name 'dev'
```

- Untuk detail API, lihat [DeleteStage](#) di Referensi AWS CLI Perintah.

## **delete-usage-plan-key**

Contoh kode berikut menunjukkan cara menggunakan `delete-usage-plan-key`.

AWS CLI

Untuk menghapus kunci API dari Paket Penggunaan

Perintah:

```
aws apigateway delete-usage-plan-key --usage-plan-id a1b2c3 --key-id  
1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu
```

- Untuk detail API, lihat [DeleteUsagePlanKey](#) di Referensi AWS CLI Perintah.

## **delete-usage-plan**

Contoh kode berikut menunjukkan cara menggunakan `delete-usage-plan`.

AWS CLI

Untuk menghapus Paket Penggunaan

Perintah:

```
aws apigateway delete-usage-plan --usage-plan-id a1b2c3
```

- Untuk detail API, lihat [DeleteUsagePlan](#) di Referensi AWS CLI Perintah.

## **flush-stage-authorizers-cache**

Contoh kode berikut menunjukkan cara menggunakan `flush-stage-authorizers-cache`.

## AWS CLI

Untuk membersihkan semua entri cache otorisasi di atas panggung

Perintah:

```
aws apigateway flush-stage-authorizers-cache --rest-api-id 1234123412 --stage-name dev
```

- Untuk detail API, lihat [FlushStageAuthorizersCached](#) di Referensi AWS CLI Perintah.

### **flush-stage-cache**

Contoh kode berikut menunjukkan cara menggunakan `flush-stage-cache`.

## AWS CLI

Untuk membersihkan cache untuk tahap API

Perintah:

```
aws apigateway flush-stage-cache --rest-api-id 1234123412 --stage-name dev
```

- Untuk detail API, lihat [FlushStageCache](#) di Referensi AWS CLI Perintah.

### **generate-client-certificate**

Contoh kode berikut menunjukkan cara menggunakan `generate-client-certificate`.

## AWS CLI

Untuk membuat Sertifikat SSL Sisi Klien

Perintah:

```
aws apigateway generate-client-certificate --description 'My First Client Certificate'
```

- Untuk detail API, lihat [GenerateClientCertificate](#) di Referensi AWS CLI Perintah.

## get-account

Contoh kode berikut menunjukkan cara menggunakan `get-account`.

### AWS CLI

Untuk mendapatkan pengaturan akun API Gateway

Perintah:

```
aws apigateway get-account
```

Output:

```
{
  "cloudwatchRoleArn": "arn:aws:iam::123412341234:role/
APIGatewayToCloudWatchLogsRole",
  "throttleSettings": {
    "rateLimit": 500.0,
    "burstLimit": 1000
  }
}
```

- Untuk detail API, lihat [GetAccount](#) di Referensi AWS CLI Perintah.

## get-api-key

Contoh kode berikut menunjukkan cara menggunakan `get-api-key`.

### AWS CLI

Untuk mendapatkan informasi tentang kunci API tertentu

Perintah:

```
aws apigateway get-api-key --api-key 8bk1k8b11k3sB38D9B310enyWT8c09B301kq0b1k
```

Output:

```
{
```



```
"description": "My first key",
"enabled": true,
"stageKeys": [
  "a1b2c3d4e5/dev",
  "e5d4c3b2a1/dev"
],
"lastUpdatedDate": 1456184515,
"createdDate": 1456184452,
"id": "8bk1k8b11k3sB38D9B310enyWT8c09B301kq0b1k",
"name": "My key"
}
```

- Untuk detail API, lihat [GetApiKey](#) di Referensi AWS CLI Perintah.

## get-api-keys

Contoh kode berikut menunjukkan cara menggunakan `get-api-keys`.

### AWS CLI

Untuk mendapatkan daftar kunci API

Perintah:

```
aws apigateway get-api-keys
```

Output:

```
{
  "items": [
    {
      "description": "My first key",
      "enabled": true,
      "stageKeys": [
        "a1b2c3d4e5/dev",
        "e5d4c3b2a1/dev"
      ],
      "lastUpdatedDate": 1456184515,
      "createdDate": 1456184452,
      "id": "8bk1k8b11k3sB38D9B310enyWT8c09B301kq0b1k",
      "name": "My key"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [GetApiKeys](#) di Referensi AWS CLI Perintah.

## get-authorizer

Contoh kode berikut menunjukkan cara menggunakan `get-authorizer`.

### AWS CLI

Untuk mendapatkan pengaturan API Gateway per-API Authorizer

Perintah:

```
aws apigateway get-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3
```

Output:

```
{
  "authorizerResultTtlInSeconds": 300,
  "name": "MyAuthorizer",
  "type": "TOKEN",
  "identitySource": "method.request.header.Authorization",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:authorizer_function/invocations",
  "id": "gfi4n3"
}
```

- Untuk detail API, lihat [GetAuthorizer](#) di Referensi AWS CLI Perintah.

## get-authorizers

Contoh kode berikut menunjukkan cara menggunakan `get-authorizers`.

### AWS CLI

Untuk mendapatkan daftar otorisasi untuk REST API

Perintah:

```
aws apigateway get-authorizers --rest-api-id 1234123412
```

Output:

```
{
  "items": [
    {
      "name": "MyAuthorizer",
      "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/
functions/arn:aws:lambda:us-west-2:123412341234:function:My_Authorizer_Function/
invocations",
      "authorizerResultTtlInSeconds": 300,
      "identitySource": "method.request.header.Authorization",
      "type": "TOKEN",
      "id": "gfi4n3"
    }
  ]
}
```

- Untuk detail API, lihat [GetAuthorizers](#) di Referensi AWS CLI Perintah.

## get-base-path-mapping

Contoh kode berikut menunjukkan cara menggunakan `get-base-path-mapping`.

AWS CLI

Untuk mendapatkan pemetaan jalur dasar untuk nama domain khusus

Perintah:

```
aws apigateway get-base-path-mapping --domain-name subdomain.domain.tld --base-path
v1
```

Output:

```
{
  "basePath": "v1",
  "restApiId": "1234w4321e",
  "stage": "api"
}
```

- Untuk detail API, lihat [GetBasePathMapping](#) di Referensi AWS CLI Perintah.

## get-base-path-mappings

Contoh kode berikut menunjukkan cara menggunakan `get-base-path-mappings`.

### AWS CLI

Untuk mendapatkan pemetaan jalur dasar untuk nama domain kustom

Perintah:

```
aws apigateway get-base-path-mappings --domain-name subdomain.domain.tld
```

Output:

```
{
  "items": [
    {
      "basePath": "(none)",
      "restApiId": "1234w4321e",
      "stage": "dev"
    },
    {
      "basePath": "v1",
      "restApiId": "1234w4321e",
      "stage": "api"
    }
  ]
}
```

- Untuk detail API, lihat [GetBasePathMappings](#) di Referensi AWS CLI Perintah.

## get-client-certificate

Contoh kode berikut menunjukkan cara menggunakan `get-client-certificate`.

### AWS CLI

Untuk mendapatkan sertifikat klien

Perintah:

```
aws apigateway get-client-certificate --client-certificate-id a1b2c3
```

- Untuk detail API, lihat [GetClientCertificate](#) di Referensi AWS CLI Perintah.

## get-client-certificates

Contoh kode berikut menunjukkan cara menggunakan `get-client-certificates`.

### AWS CLI

Untuk mendapatkan daftar sertifikat klien

Perintah:

```
aws apigateway get-client-certificates
```

Output:

```
{
  "items": [
    {
      "pemEncodedCertificate": "-----BEGIN CERTIFICATE----- <certificate
content> -----END CERTIFICATE-----",
      "clientCertificateId": "a1b2c3",
      "expirationDate": 1483556561,
      "description": "My Client Certificate",
      "createdDate": 1452020561
    }
  ]
}
```

- Untuk detail API, lihat [GetClientCertificates](#) di Referensi AWS CLI Perintah.

## get-deployment

Contoh kode berikut menunjukkan cara menggunakan `get-deployment`.

### AWS CLI

Untuk mendapatkan informasi tentang penerapan

**Perintah:**

```
aws apigateway get-deployment --rest-api-id 1234123412 --deployment-id ztt4m2
```

**Output:**

```
{
  "description": "myDeployment",
  "id": "ztt4m2",
  "createdDate": 1455218022
}
```

- Untuk detail API, lihat [GetDeployment](#) di Referensi AWS CLI Perintah.

**get-deployments**

Contoh kode berikut menunjukkan cara menggunakan `get-deployments`.

**AWS CLI**

Untuk mendapatkan daftar penerapan untuk REST API

**Perintah:**

```
aws apigateway get-deployments --rest-api-id 1234123412
```

**Output:**

```
{
  "items": [
    {
      "createdDate": 1453797217,
      "id": "0a2b4c",
      "description": "Deployed my API for the first time"
    }
  ]
}
```

- Untuk detail API, lihat [GetDeployments](#) di Referensi AWS CLI Perintah.

## get-domain-name

Contoh kode berikut menunjukkan cara menggunakan `get-domain-name`.

### AWS CLI

Untuk mendapatkan informasi tentang nama domain khusus

Perintah:

```
aws apigateway get-domain-name --domain-name api.domain.tld
```

Output:

```
{
  "domainName": "api.domain.tld",
  "distributionDomainName": "d1a2f3a4c5o6d.cloudfront.net",
  "certificateName": "uploadedCertificate",
  "certificateUploadDate": 1462565487
}
```

- Untuk detail API, lihat [GetDomainName](#) di Referensi AWS CLI Perintah.

## get-domain-names

Contoh kode berikut menunjukkan cara menggunakan `get-domain-names`.

### AWS CLI

Untuk mendapatkan daftar nama domain khusus

Perintah:

```
aws apigateway get-domain-names
```

Output:

```
{
  "items": [
    {
      "distributionDomainName": "d9511k3109bkd.cloudfront.net",
```

```
    "certificateUploadDate": 1452812505,  
    "certificateName": "my_custom_domain-certificate",  
    "domainName": "subdomain.domain.tld"  
  }  
]  
}
```

- Untuk detail API, lihat [GetDomainNames](#) di Referensi AWS CLI Perintah.

## get-export

Contoh kode berikut menunjukkan cara menggunakan `get-export`.

### AWS CLI

Untuk mendapatkan template JSON Swagger untuk sebuah panggung

Perintah:

```
aws apigateway get-export --rest-api-id a1b2c3d4e5 --stage-name dev --export-type  
swagger /path/to/filename.json
```

Untuk mendapatkan template JSON Swagger+API Gateway Extentions untuk sebuah panggung

Perintah:

```
aws apigateway get-export --parameters extensions='integrations' --rest-api-id  
a1b2c3d4e5 --stage-name dev --export-type swagger /path/to/filename.json
```

Untuk mendapatkan template JSON Swagger+Ekstensi Postman untuk sebuah panggung

Perintah:

```
aws apigateway get-export --parameters extensions='postman' --rest-api-id a1b2c3d4e5  
--stage-name dev --export-type swagger /path/to/filename.json
```

- Untuk detail API, lihat [GetExport](#) di Referensi AWS CLI Perintah.

## get-integration-response

Contoh kode berikut menunjukkan cara menggunakan `get-integration-response`.



## AWS CLI

Untuk mendapatkan konfigurasi respons integrasi untuk metode HTTP yang ditentukan di bawah sumber daya REST API

Perintah:

```
aws apigateway get-integration-response --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET --status-code 200
```

Output:

```
{
  "statusCode": "200",
  "responseTemplates": {
    "application/json": null
  }
}
```

- Untuk detail API, lihat [GetIntegrationResponse](#) di Referensi AWS CLI Perintah.

## get-integration

Contoh kode berikut menunjukkan cara menggunakan `get-integration`.

## AWS CLI

Untuk mendapatkan konfigurasi integrasi untuk metode HTTP yang didefinisikan di bawah sumber daya REST API

Perintah:

```
aws apigateway get-integration --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET
```

Output:

```
{
  "httpMethod": "POST",
```

```

    "integrationResponses": {
      "200": {
        "responseTemplates": {
          "application/json": null
        },
        "statusCode": "200"
      }
    },
    "cacheKeyParameters": [],
    "type": "AWS",
    "uri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:My_Function/invocations",
    "cacheNamespace": "y9h6rt"
  }
}

```

- Untuk detail API, lihat [GetIntegration](#) di Referensi AWS CLI Perintah.

## get-method-response

Contoh kode berikut menunjukkan cara menggunakan `get-method-response`.

### AWS CLI

Untuk mendapatkan konfigurasi sumber daya respons metode untuk metode HTTP yang didefinisikan di bawah sumber daya REST API

Perintah:

```
aws apigateway get-method-response --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET --status-code 200
```

Output:

```

{
  "responseModels": {
    "application/json": "Empty"
  },
  "statusCode": "200"
}

```

- Untuk detail API, lihat [GetMethodResponse](#) di Referensi AWS CLI Perintah.

## get-method

Contoh kode berikut menunjukkan cara menggunakan `get-method`.

### AWS CLI

Untuk mendapatkan konfigurasi sumber daya metode untuk metode HTTP yang didefinisikan di bawah sumber daya REST API

Perintah:

```
aws apigateway get-method --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET
```

Output:

```
{
  "apiKeyRequired": false,
  "httpMethod": "GET",
  "methodIntegration": {
    "integrationResponses": {
      "200": {
        "responseTemplates": {
          "application/json": null
        },
        "statusCode": "200"
      }
    },
    "cacheKeyParameters": [],
    "uri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:My_Function/invocations",
    "httpMethod": "POST",
    "cacheNamespace": "y9h6rt",
    "type": "AWS"
  },
  "requestParameters": {},
  "methodResponses": {
    "200": {
      "responseModels": {
        "application/json": "Empty"
      },
      "statusCode": "200"
    }
  }
}
```

```
    },  
    "authorizationType": "NONE"  
  }  
}
```

- Untuk detail API, lihat [GetMethod](#) di Referensi AWS CLI Perintah.

## get-model-template

Contoh kode berikut menunjukkan cara menggunakan `get-model-template`.

### AWS CLI

Untuk mendapatkan template pemetaan untuk model yang ditentukan di bawah REST API

Perintah:

```
aws apigateway get-model-template --rest-api-id 1234123412 --model-name Empty
```

Output:

```
{  
  "value": "#set($inputRoot = $input.path('$'))\n{ }"  
}
```

- Untuk detail API, lihat [GetModelTemplate](#) di Referensi AWS CLI Perintah.

## get-model

Contoh kode berikut menunjukkan cara menggunakan `get-model`.

### AWS CLI

Untuk mendapatkan konfigurasi untuk model yang ditentukan di bawah REST API

Perintah:

```
aws apigateway get-model --rest-api-id 1234123412 --model-name Empty
```

Output:

```
{
```

```

    "contentType": "application/json",
    "description": "This is a default empty schema model",
    "name": "Empty",
    "id": "etd5w5",
    "schema": "{\n  \"schema\": \"http://json-schema.org/draft-04/schema#\",
  \"title\": \"Empty Schema\",
  \"type\": \"object\"
}"
}

```

- Untuk detail API, lihat [GetModel](#) di Referensi AWS CLI Perintah.

## get-models

Contoh kode berikut menunjukkan cara menggunakan `get-models`.

### AWS CLI

Untuk mendapatkan daftar model untuk REST API

Perintah:

```
aws apigateway get-models --rest-api-id 1234123412
```

Output:

```

{
  "items": [
    {
      "description": "This is a default error schema model",
      "schema": "{\n  \"schema\": \"http://json-schema.org/draft-04/schema#\",
  \"title\": \"Error Schema\",
  \"type\": \"object\",
  \"properties\": {
    \"message\": {
      \"type\": \"string\"
    }
  }
}",
      "contentType": "application/json",
      "id": "7tpbze",
      "name": "Error"
    },
    {
      "description": "This is a default empty schema model",
      "schema": "{\n  \"schema\": \"http://json-schema.org/draft-04/schema#\",
  \"title\": \"Empty Schema\",
  \"type\": \"object\"
}",
      "contentType": "application/json",
      "id": "etd5w5",
      "name": "Empty"
    }
  ]
}

```

```
]
}
```

- Untuk detail API, lihat [GetModels](#) di Referensi AWS CLI Perintah.

## get-resource

Contoh kode berikut menunjukkan cara menggunakan `get-resource`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber daya

Perintah:

```
aws apigateway get-resource --rest-api-id 1234123412 --resource-id zwo0y3
```

Output:

```
{
  "path": "/path",
  "pathPart": "path",
  "id": "zwo0y3",
  "parentId": "uyokt6ij2g"
}
```

- Untuk detail API, lihat [GetResource](#) di Referensi AWS CLI Perintah.

## get-resources

Contoh kode berikut menunjukkan cara menggunakan `get-resources`.

### AWS CLI

Untuk mendapatkan daftar sumber daya untuk REST API

Perintah:

```
aws apigateway get-resources --rest-api-id 1234123412
```

Output:

```
{
  "items": [
    {
      "path": "/resource/subresource",
      "resourceMethods": {
        "POST": {}
      },
      "id": "024ace",
      "pathPart": "subresource",
      "parentId": "ai5b02"
    }
  ]
}
```

- Untuk detail API, lihat [GetResources](#) di Referensi AWS CLI Perintah.

## get-rest-api

Contoh kode berikut menunjukkan cara menggunakan `get-rest-api`.

### AWS CLI

Untuk mendapatkan informasi tentang API

Perintah:

```
aws apigateway get-rest-api --rest-api-id 1234123412
```

Output:

```
{
  "name": "myAPI",
  "id": "o1y243m4f5",
  "createdDate": 1453416433
}
```

- Untuk detail API, lihat [GetRestApi](#) di Referensi AWS CLI Perintah.

## get-rest-apis

Contoh kode berikut menunjukkan cara menggunakan `get-rest-apis`.

## AWS CLI

Untuk mendapatkan daftar REST API

Perintah:

```
aws apigateway get-rest-apis
```

Output:

```
{
  "items": [
    {
      "createdDate": 1438884790,
      "id": "12s44z21rb",
      "name": "My First API"
    }
  ]
}
```

- Untuk detail API, lihat [GetRestApis](#) di Referensi AWS CLI Perintah.

## get-sdk

Contoh kode berikut menunjukkan cara menggunakan `get-sdk`.

## AWS CLI

Untuk mendapatkan Android SDK untuk tahap REST API

Perintah:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-type android
--parameters
groupId='com.mycompany',invokerPackage='com.mycompany.clientsdk',artifactId='Mycompany-
client',artifactVersion='1.0.0' /path/to/android_sdk.zip
```

Output:

```
{
  "contentType": "application/octet-stream",
  "contentDisposition": "attachment; filename=\"android_2016-02-22_23-52Z.zip\""
}
```



```
}
```

Untuk mendapatkan SDK IOS untuk tahap REST API

Perintah:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-type
objectivec --parameters classPrefix='myprefix' /path/to/iOS_sdk.zip
```

Output:

```
{
  "contentType": "application/octet-stream",
  "contentDisposition": "attachment; filename=\"objectivec_2016-02-22_23-52Z.zip\"
}"
```

Untuk mendapatkan Javascript SDK untuk tahap REST API

Perintah:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-type
javascript /path/to/javascript_sdk.zip
```

Output:

```
{
  "contentType": "application/octet-stream",
  "contentDisposition": "attachment; filename=\"javascript_2016-02-22_23-52Z.zip\"
}"
```

- Untuk detail API, lihat [GetSdk](#) di Referensi AWS CLI Perintah.

## get-stage

Contoh kode berikut menunjukkan cara menggunakan `get-stage`.

### AWS CLI

Untuk mendapatkan informasi tentang tahap API

**Perintah:**

```
aws apigateway get-stage --rest-api-id 1234123412 --stage-name dev
```

**Output:**

```
{
  "stageName": "dev",
  "cacheClusterSize": "0.5",
  "cacheClusterEnabled": false,
  "cacheClusterStatus": "NOT_AVAILABLE",
  "deploymentId": "rbh1fj",
  "lastUpdatedDate": 1466802961,
  "createdDate": 1460682074,
  "methodSettings": {
    "*/*": {
      "cacheTtlInSeconds": 300,
      "loggingLevel": "INFO",
      "dataTraceEnabled": false,
      "metricsEnabled": true,
      "unauthorizedCacheControlHeaderStrategy":
"SUCCEED_WITH_RESPONSE_HEADER",
      "throttlingRateLimit": 500.0,
      "cacheDataEncrypted": false,
      "cachingEnabled": false,
      "throttlingBurstLimit": 1000,
      "requireAuthorizationForCacheControl": true
    },
    "~1resource/GET": {
      "cacheTtlInSeconds": 300,
      "loggingLevel": "INFO",
      "dataTraceEnabled": false,
      "metricsEnabled": true,
      "unauthorizedCacheControlHeaderStrategy":
"SUCCEED_WITH_RESPONSE_HEADER",
      "throttlingRateLimit": 500.0,
      "cacheDataEncrypted": false,
      "cachingEnabled": false,
      "throttlingBurstLimit": 1000,
      "requireAuthorizationForCacheControl": true
    }
  }
}
```

- Untuk detail API, lihat [GetStage](#) di Referensi AWS CLI Perintah.

## get-stages

Contoh kode berikut menunjukkan cara menggunakan `get-stages`.

### AWS CLI

Untuk mendapatkan daftar tahapan untuk REST API

Perintah:

```
aws apigateway get-stages --rest-api-id 1234123412
```

Output:

```
{
  "item": [
    {
      "stageName": "dev",
      "cacheClusterSize": "0.5",
      "cacheClusterEnabled": true,
      "cacheClusterStatus": "AVAILABLE",
      "deploymentId": "123h64",
      "lastUpdatedDate": 1456185138,
      "createdDate": 1453589092,
      "methodSettings": {
        "~1resource~1subresource/POST": {
          "cacheTtlInSeconds": 300,
          "loggingLevel": "INFO",
          "dataTraceEnabled": true,
          "metricsEnabled": true,
          "throttlingRateLimit": 500.0,
          "cacheDataEncrypted": false,
          "cachingEnabled": false,
          "throttlingBurstLimit": 1000
        }
      }
    }
  ]
}
```

- Untuk detail API, lihat [GetStages](#) di Referensi AWS CLI Perintah.

## **get-usage-plan-key**

Contoh kode berikut menunjukkan cara menggunakan `get-usage-plan-key`.

### AWS CLI

Untuk mendapatkan detail kunci API yang terkait dengan Paket Penggunaan

Perintah:

```
aws apigateway get-usage-plan-key --usage-plan-id a1b2c3 --key-id
1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu
```

- Untuk detail API, lihat [GetUsagePlanKey](#) di Referensi AWS CLI Perintah.

## **get-usage-plan-keys**

Contoh kode berikut menunjukkan cara menggunakan `get-usage-plan-keys`.

### AWS CLI

Untuk mendapatkan daftar kunci API yang terkait dengan Paket Penggunaan

Perintah:

```
aws apigateway get-usage-plan-keys --usage-plan-id a1b2c3
```

- Untuk detail API, lihat [GetUsagePlanKeys](#) di Referensi AWS CLI Perintah.

## **get-usage-plan**

Contoh kode berikut menunjukkan cara menggunakan `get-usage-plan`.

### AWS CLI

Untuk mendapatkan detail Paket Penggunaan

Perintah:

```
aws apigateway get-usage-plan --usage-plan-id a1b2c3
```

- Untuk detail API, lihat [GetUsagePlan](#) di Referensi AWS CLI Perintah.

## **get-usage-plans**

Contoh kode berikut menunjukkan cara menggunakan `get-usage-plans`.

### AWS CLI

Untuk mendapatkan detail dari semua Paket Penggunaan

Perintah:

```
aws apigateway get-usage-plans
```

- Untuk detail API, lihat [GetUsagePlans](#) di Referensi AWS CLI Perintah.

## **get-usage**

Contoh kode berikut menunjukkan cara menggunakan `get-usage`.

### AWS CLI

Untuk mendapatkan detail penggunaan untuk Paket Penggunaan

Perintah:

```
aws apigateway get-usage --usage-plan-id a1b2c3 --start-date "2016-08-16" --end-date "2016-08-17"
```

- Untuk detail API, lihat [GetUsage](#) di Referensi AWS CLI Perintah.

## **import-rest-api**

Contoh kode berikut menunjukkan cara menggunakan `import-rest-api`.

### AWS CLI

Untuk mengimpor template Swagger dan membuat API

Perintah:

```
aws apigateway import-rest-api --body 'file:///path/to/API_Swagger_template.json'
```

- Untuk detail API, lihat [ImportRestApi](#) di Referensi AWS CLI Perintah.

## put-integration-response

Contoh kode berikut menunjukkan cara menggunakan `put-integration-response`.

### AWS CLI

Untuk membuat respons integrasi sebagai respons default dengan template pemetaan yang ditentukan

Perintah:

```
aws apigateway put-integration-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 200 --selection-pattern "" --response-templates '{"application/json": "{$\json\: \"template\"}'}
```

Untuk membuat respons integrasi dengan regex 400 dan nilai header yang ditentukan secara statis

Perintah:

```
aws apigateway put-integration-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 400 --selection-pattern 400 --response-parameters '{"method.response.header.custom-header": ""}'
```

- Untuk detail API, lihat [PutIntegrationResponse](#) di Referensi AWS CLI Perintah.

## put-integration

Contoh kode berikut menunjukkan cara menggunakan `put-integration`.

### AWS CLI

Untuk membuat permintaan integrasi MOCK

**Perintah:**

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --type MOCK --request-templates '{ "application/json": "{\"statusCode\": 200}" }'
```

Untuk membuat permintaan integrasi HTTP

**Perintah:**

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --type HTTP --integration-http-method GET --uri 'https://domain.tld/path'
```

Untuk membuat permintaan AWS integrasi dengan titik akhir Fungsi Lambda

**Perintah:**

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --type AWS --integration-http-method POST --uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:function_name/invocations'
```

- Untuk detail API, lihat [PutIntegration](#) di Referensi AWS CLI Perintah.

## put-method-response

Contoh kode berikut menunjukkan cara menggunakan `put-method-response`.

### AWS CLI

Untuk membuat respons metode di bawah kode status yang ditentukan dengan header respons metode kustom

**Perintah:**

```
aws apigateway put-method-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 400 --response-parameters "method.response.header.custom-header=false"
```

- Untuk detail API, lihat [PutMethodResponse](#) di Referensi AWS CLI Perintah.

## put-method

Contoh kode berikut menunjukkan cara menggunakan `put-method`.

### AWS CLI

Untuk membuat metode untuk sumber daya di API tanpa otorisasi, tanpa kunci API, dan header permintaan metode khusus

Perintah:

```
aws apigateway put-method --rest-api-id 1234123412 --resource-id a1b2c3 --http-method PUT --authorization-type "NONE" --no-api-key-required --request-parameters "method.request.header.custom-header=false"
```

- Untuk detail API, lihat [PutMethod](#) di Referensi AWS CLI Perintah.

## put-rest-api

Contoh kode berikut menunjukkan cara menggunakan `put-rest-api`.

### AWS CLI

Untuk menimpa API yang ada menggunakan template Swagger

Perintah:

```
aws apigateway put-rest-api --rest-api-id 1234123412 --mode overwrite --body 'fileb:///path/to/API_Swagger_template.json'
```

Untuk menggabungkan template Swagger ke dalam API yang ada

Perintah:

```
aws apigateway put-rest-api --rest-api-id 1234123412 --mode merge --body 'fileb:///path/to/API_Swagger_template.json'
```

- Untuk detail API, lihat [PutRestApi](#) di Referensi AWS CLI Perintah.

## test-invoke-authorizer

Contoh kode berikut menunjukkan cara menggunakan `test-invoke-authorizer`.



## AWS CLI

Untuk menguji, panggil permintaan ke Custom Authorizer termasuk header dan nilai yang diperlukan

Perintah:

```
aws apigateway test-invoke-authorizer --rest-api-id 1234123412 --authorizer-id 5yid1t --headers Authorization='Value'
```

- Untuk detail API, lihat [TestInvokeAuthorizer](#) di Referensi AWS CLI Perintah.

## test-invoke-method

Contoh kode berikut menunjukkan cara menggunakan `test-invoke-method`.

### AWS CLI

Untuk menguji, panggil sumber daya root di API dengan membuat permintaan GET

Perintah:

```
aws apigateway test-invoke-method --rest-api-id 1234123412 --resource-id av15sg8fw8 --http-method GET --path-with-query-string '/'
```

Untuk menguji, panggil sub-sumber daya di API dengan membuat permintaan GET dengan nilai parameter jalur yang ditentukan

Perintah:

```
aws apigateway test-invoke-method --rest-api-id 1234123412 --resource-id 3gapai --http-method GET --path-with-query-string '/pets/1'
```

- Untuk detail API, lihat [TestInvokeMethod](#) di Referensi AWS CLI Perintah.

## update-account

Contoh kode berikut menunjukkan cara menggunakan `update-account`.

### AWS CLI

Untuk mengubah ARN Peran IAM untuk pencatatan ke Log CloudWatch

**Perintah:**

```
aws apigateway update-account --patch-operations op='replace',path='/cloudwatchRoleArn',value='arn:aws:iam::123412341234:role/APIGatewayToCloudWatchLogs'
```

**Output:**

```
{
  "cloudwatchRoleArn": "arn:aws:iam::123412341234:role/APIGatewayToCloudWatchLogs",
  "throttleSettings": {
    "rateLimit": 1000.0,
    "burstLimit": 2000
  }
}
```

- Untuk detail API, lihat [UpdateAccount](#) di Referensi AWS CLI Perintah.

**update-api-key**

Contoh kode berikut menunjukkan cara menggunakan `update-api-key`.

**AWS CLI**

Untuk mengubah nama Kunci API

**Perintah:**

```
aws apigateway update-api-key --api-key sNvjQDMReA1eEQPNAW8r37XsU2rDD7fc7m2SiMnu --patch-operations op='replace',path='/name',value='newName'
```

**Output:**

```
{
  "description": "currentDescription",
  "enabled": true,
  "stageKeys": [
    "41t2j324r5/dev"
  ],
  "lastUpdatedDate": 1470086052,
  "createdDate": 1445460347,
```

```
"id": "sNvjQDMReA1vEQPNzW8r3dXsU2rrD7fcjm2SiMnu",
"name": "newName"
}
```

Untuk menonaktifkan Kunci API

Perintah:

```
aws apigateway update-api-key --api-key sNvjQDMReA1eEQPNAW8r37XsU2rDD7fc7m2SiMnu --
patch-operations op='replace',path='/enabled',value='false'
```

Output:

```
{
  "description": "currentDescription",
  "enabled": false,
  "stageKeys": [
    "41t2j324r5/dev"
  ],
  "lastUpdatedDate": 1470086052,
  "createdDate": 1445460347,
  "id": "sNvjQDMReA1vEQPNzW8r3dXsU2rrD7fcjm2SiMnu",
  "name": "newName"
}
```

- Untuk detail API, lihat [UpdateApiKey](#) di Referensi AWS CLI Perintah.

## update-authorizer

Contoh kode berikut menunjukkan cara menggunakan `update-authorizer`.

### AWS CLI

Untuk mengubah nama Custom Authorizer

Perintah:

```
aws apigateway update-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3 --
patch-operations op='replace',path='/name',value='testAuthorizer'
```

Output:

```
{
  "authType": "custom",
  "name": "testAuthorizer",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthorizer/invocations",
  "authorizerResultTtlInSeconds": 300,
  "identitySource": "method.request.header.Authorization",
  "type": "TOKEN",
  "id": "gfi4n3"
}
```

Untuk mengubah Fungsi Lambda yang dipanggil oleh Custom Authorizer

Perintah:

```
aws apigateway update-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3 --
patch-operations op='replace',path='/authorizerUri',value='arn:aws:apigateway:us-
west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-
west-2:123412341234:function:newAuthorizer/invocations'
```

Output:

```
{
  "authType": "custom",
  "name": "testAuthorizer",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:newAuthorizer/invocations",
  "authorizerResultTtlInSeconds": 300,
  "identitySource": "method.request.header.Authorization",
  "type": "TOKEN",
  "id": "gfi4n3"
}
```

- Untuk detail API, lihat [UpdateAuthorizer](#) di Referensi AWS CLI Perintah.

## update-base-path-mapping

Contoh kode berikut menunjukkan cara menggunakan `update-base-path-mapping`.

AWS CLI

Untuk mengubah jalur dasar untuk nama domain kustom

**Perintah:**

```
aws apigateway update-base-path-mapping --domain-name api.domain.tld --base-path prod --patch-operations op='replace',path='/basePath',value='v1'
```

**Output:**

```
{
  "basePath": "v1",
  "restApiId": "1234123412",
  "stage": "api"
}
```

- Untuk detail API, lihat [UpdateBasePathMapping](#) di Referensi AWS CLI Perintah.

**update-client-certificate**

Contoh kode berikut menunjukkan cara menggunakan `update-client-certificate`.

**AWS CLI**

Untuk memperbarui deskripsi sertifikat klien

**Perintah:**

```
aws apigateway update-client-certificate --client-certificate-id a1b2c3 --patch-operations op='replace',path='/description',value='My new description'
```

- Untuk detail API, lihat [UpdateClientCertificate](#) di Referensi AWS CLI Perintah.

**update-deployment**

Contoh kode berikut menunjukkan cara menggunakan `update-deployment`.

**AWS CLI**

Untuk mengubah deskripsi penerapan

**Perintah:**

```
aws apigateway update-deployment --rest-api-id 1234123412 --deployment-id ztt4m2 --
patch-operations op='replace',path='/description',value='newDescription'
```

Output:

```
{
  "description": "newDescription",
  "id": "ztt4m2",
  "createdDate": 1455218022
}
```

- Untuk detail API, lihat [UpdateDeployment](#) di Referensi AWS CLI Perintah.

## update-domain-name

Contoh kode berikut menunjukkan cara menggunakan `update-domain-name`.

AWS CLI

Untuk mengubah nama sertifikat untuk nama domain kustom

`update-domain-name` Contoh berikut mengubah nama sertifikat untuk domain kustom.

```
aws apigateway update-domain-name \
  --domain-name api.domain.tld \
  --patch-operations op='replace',path='/certificateArn',value='arn:aws:acm:us-
west-2:111122223333:certificate/CERTEXAMPLE123EXAMPLE'
```

Output:

```
{
  "domainName": "api.domain.tld",
  "distributionDomainName": "d123456789012.cloudfront.net",
  "certificateArn": "arn:aws:acm:us-west-2:111122223333:certificate/
CERTEXAMPLE123EXAMPLE",
  "certificateUploadDate": 1462565487
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Nama Domain Kustom untuk API di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateDomainName](#) di Referensi AWS CLI Perintah.

## update-integration-response

Contoh kode berikut menunjukkan cara menggunakan `update-integration-response`.

### AWS CLI

Untuk mengubah header respons integrasi untuk memiliki pemetaan statis `*`

Perintah:

```
aws apigateway update-integration-response --rest-api-id 1234123412 --
resource-id 3gapai --http-method GET --status-code 200 --patch-operations
  op='replace',path='/responseParameters/method.response.header.Access-Control-Allow-
  Origin',value='''''*''''
```

Output:

```
{
  "statusCode": "200",
  "responseParameters": {
    "method.response.header.Access-Control-Allow-Origin": "''''*"
  }
}
```

Untuk menghapus header respons integrasi

Perintah:

```
aws apigateway update-integration-response --rest-api-id 1234123412 --resource-id
3gapai --http-method GET --status-code 200 --patch-operations op='remove',path='/
responseParameters/method.response.header.Access-Control-Allow-Origin'
```

- Untuk detail API, lihat [UpdateIntegrationResponse](#) di Referensi AWS CLI Perintah.

## update-integration

Contoh kode berikut menunjukkan cara menggunakan `update-integration`.

### AWS CLI

Untuk menambahkan Template Pemetaan `'Content-Type: application/json'` yang dikonfigurasi dengan Input Passthrough

**Perintah:**

```
aws apigateway update-integration \  
  --rest-api-id a1b2c3d4e5 \  
  --resource-id a1b2c3 \  
  --http-method POST \  
  --patch-operations "op='add',path='/requestTemplates/application~1json'"
```

Untuk memperbarui (mengganti) Template Pemetaan 'Content-Type: application/json' yang dikonfigurasi dengan templat khusus

**Perintah:**

```
aws apigateway update-integration \  
  --rest-api-id a1b2c3d4e5 \  
  --resource-id a1b2c3 \  
  --http-method POST \  
  --patch-operations "op='replace',path='/requestTemplates/  
application~1json',value='{\"example\": \"json\"}'"
```

Untuk memperbarui (mengganti) template khusus yang terkait dengan 'Content-Type: application/json' dengan Input Passthrough

**Perintah:**

```
aws apigateway update-integration \  
  --rest-api-id a1b2c3d4e5 \  
  --resource-id a1b2c3 \  
  --http-method POST \  
  --patch-operations "op='replace',path='requestTemplates/application~1json'"
```

Untuk menghapus Template Pemetaan 'Content-Type: application/json'

**Perintah:**

```
aws apigateway update-integration \  
  --rest-api-id a1b2c3d4e5 \  
  --resource-id a1b2c3 \  
  --http-method POST \  
  --patch-operations "op='remove',path='/requestTemplates/application~1json'"
```



- Untuk detail API, lihat [UpdateIntegration](#) di Referensi AWS CLI Perintah.

## update-method-response

Contoh kode berikut menunjukkan cara menggunakan `update-method-response`.

### AWS CLI

Untuk membuat header respons metode baru untuk respons 200 dalam metode dan mendefinisikannya sebagai tidak diperlukan (default)

Perintah:

```
aws apigateway update-method-response --rest-api-id 1234123412 --resource-id
a1b2c3 --http-method GET --status-code 200 --patch-operations op="add",path="/
responseParameters/method.response.header.custom-header",value="false"
```

Untuk menghapus model respons untuk respons 200 dalam suatu metode

Perintah:

```
aws apigateway update-method-response --rest-api-id 1234123412 --resource-id
a1b2c3 --http-method GET --status-code 200 --patch-operations op="remove",path="/
responseModels/application~1json"
```

- Untuk detail API, lihat [UpdateMethodResponse](#) di Referensi AWS CLI Perintah.

## update-method

Contoh kode berikut menunjukkan cara menggunakan `update-method`.

### AWS CLI

Contoh 1: Untuk memodifikasi metode agar memerlukan kunci API

`update-method` Contoh berikut memodifikasi metode untuk memerlukan kunci API.

```
aws apigateway update-method \  
  --rest-api-id 1234123412 \  
  --resource-id a1b2c3 \  
  --http-method GET \  
  --patch-operations op="add",path="/method/requestParameters/method.requestParameters.apiKeyRequired",value="true"
```

```
--patch-operations op="replace",path="/apiKeyRequired",value="true"
```

### Output:

```
{
  "httpMethod": "GET",
  "authorizationType": "NONE",
  "apiKeyRequired": true,
  "methodResponses": {
    "200": {
      "statusCode": "200",
      "responseModels": {}
    }
  },
  "methodIntegration": {
    "type": "AWS",
    "httpMethod": "POST",
    "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
    "passthroughBehavior": "WHEN_NO_MATCH",
    "contentHandling": "CONVERT_TO_TEXT",
    "timeoutInMillis": 29000,
    "cacheNamespace": "h7i8j9",
    "cacheKeyParameters": [],
    "integrationResponses": {
      "200": {
        "statusCode": "200",
        "responseTemplates": {}
      }
    }
  }
}
```

### Contoh 2: Untuk memodifikasi metode untuk meminta otorisasi IAM

update-methodContoh berikut memodifikasi metode untuk memerlukan otorisasi IAM.

```
aws apigateway update-method \
  --rest-api-id 1234123412 \
  --resource-id a1b2c3 \
  --http-method GET \
  --patch-operations op="replace",path="/authorizationType",value="AWS_IAM"
```

**Output:**

```
{
  "httpMethod": "GET",
  "authorizationType": "AWS_IAM",
  "apiKeyRequired": false,
  "methodResponses": {
    "200": {
      "statusCode": "200",
      "responseModels": {}
    }
  },
  "methodIntegration": {
    "type": "AWS",
    "httpMethod": "POST",
    "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
    "passthroughBehavior": "WHEN_NO_MATCH",
    "contentHandling": "CONVERT_TO_TEXT",
    "timeoutInMillis": 29000,
    "cacheNamespace": "h7i8j9",
    "cacheKeyParameters": [],
    "integrationResponses": {
      "200": {
        "statusCode": "200",
        "responseTemplates": {}
      }
    }
  }
}
```

**Contoh 3: Untuk memodifikasi metode untuk meminta otorisasi Lambda**

update-method Contoh berikut memodifikasi metode untuk otorisasi Lambda yang diperlukan.

```
aws apigateway update-method --rest-api-id 1234123412 \
  --resource-id a1b2c3 \
  --http-method GET \
  --patch-operations op="replace",path="/authorizationType",value="CUSTOM"
op="replace",path="/authorizerId",value="e4f5g6"
```

**Output:**

```
{
  "httpMethod": "GET",
  "authorizationType": "CUSTOM",
  "authorizerId" : "e4f5g6",
  "apiKeyRequired": false,
  "methodResponses": {
    "200": {
      "statusCode": "200",
      "responseModels": {}
    }
  },
  "methodIntegration": {
    "type": "AWS",
    "httpMethod": "POST",
    "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
    "passthroughBehavior": "WHEN_NO_MATCH",
    "contentHandling": "CONVERT_TO_TEXT",
    "timeoutInMillis": 29000,
    "cacheNamespace": "h7i8j9",
    "cacheKeyParameters": [],
    "integrationResponses": {
      "200": {
        "statusCode": "200",
        "responseTemplates": {}
      }
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat, mengonfigurasi, dan menguji paket penggunaan menggunakan API Gateway CLI dan REST API](#) serta [Mengontrol dan mengelola akses ke REST API di API Gateway di Panduan](#) Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateMethod](#) di Referensi AWS CLI Perintah.

## update-model

Contoh kode berikut menunjukkan cara menggunakan `update-model`.

### AWS CLI

Untuk mengubah deskripsi model dalam API

**Perintah:**

```
aws apigateway update-model --rest-api-id 1234123412 --model-name 'Empty' --patch-operations op=replace,path=/description,value='New Description'
```

Untuk mengubah skema model dalam API

**Perintah:**

```
aws apigateway update-model --rest-api-id 1234123412 --model-name 'Empty' --patch-operations op=replace,path=/schema,value='{ \"$schema\": \"http://json-schema.org/draft-04/schema#\", \"title\": \"Empty Schema\", \"type\": \"object\" }''
```

- Untuk detail API, lihat [UpdateModel](#) di Referensi AWS CLI Perintah.

**update-resource**

Contoh kode berikut menunjukkan cara menggunakan `update-resource`.

**AWS CLI**

Untuk memindahkan sumber daya dan menempatkannya di bawah sumber daya induk yang berbeda di API

**Perintah:**

```
aws apigateway update-resource --rest-api-id 1234123412 --resource-id 1a2b3c --patch-operations op=replace,path=/parentId,value='3c2b1a'
```

**Output:**

```
{
  "path": "/resource",
  "pathPart": "resource",
  "id": "1a2b3c",
  "parentId": "3c2b1a"
}
```

Untuk mengganti nama sumber daya (`pathPart`) di API

**Perintah:**

```
aws apigateway update-resource --rest-api-id 1234123412 --resource-id 1a2b3c --
patch-operations op=replace,path=/pathPart,value=newresourceName
```

Output:

```
{
  "path": "/newresourceName",
  "pathPart": "newresourceName",
  "id": "1a2b3c",
  "parentId": "3c2b1a"
}
```

- Untuk detail API, lihat [UpdateResource](#) di Referensi AWS CLI Perintah.

## update-rest-api

Contoh kode berikut menunjukkan cara menggunakan `update-rest-api`.

### AWS CLI

Untuk mengubah nama API

Perintah:

```
aws apigateway update-rest-api --rest-api-id 1234123412 --patch-operations
op=replace,path=/name,value='New Name'
```

Untuk mengubah deskripsi API

Perintah:

```
aws apigateway update-rest-api --rest-api-id 1234123412 --patch-operations
op=replace,path=/description,value='New Description'
```

- Untuk detail API, lihat [UpdateRestApi](#) di Referensi AWS CLI Perintah.

## update-stage

Contoh kode berikut menunjukkan cara menggunakan `update-stage`.

## AWS CLI

Contoh 1: Untuk mengganti pengaturan panggung untuk sumber daya dan metode

`update-stage` Contoh berikut mengesampingkan pengaturan panggung dan mematikan pencatatan permintaan/respons penuh untuk sumber daya dan metode tertentu.

```
aws apigateway update-stage \  
  --rest-api-id 1234123412 \  
  --stage-name 'dev' \  
  --patch-operations op=replace,path=~1resourceName/GET/logging/  
dataTrace,value=false
```

Output:

```
{  
  "deploymentId": "5ubd17",  
  "stageName": "dev",  
  "cacheClusterEnabled": false,  
  "cacheClusterStatus": "NOT_AVAILABLE",  
  "methodSettings": {  
    "~1resourceName/GET": {  
      "metricsEnabled": false,  
      "dataTraceEnabled": false,  
      "throttlingBurstLimit": 5000,  
      "throttlingRateLimit": 10000.0,  
      "cachingEnabled": false,  
      "cacheTtlInSeconds": 300,  
      "cacheDataEncrypted": false,  
      "requireAuthorizationForCacheControl": true,  
      "unauthorizedCacheControlHeaderStrategy": "SUCCEED_WITH_RESPONSE_HEADER"  
    }  
  },  
  "tracingEnabled": false,  
  "createdDate": "2022-07-18T10:11:18-07:00",  
  "lastUpdatedDate": "2022-07-18T10:19:04-07:00"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan tahapan untuk REST API](#) di Panduan Pengembang Amazon API Gateway.

Contoh 2: Untuk memperbarui pengaturan tahap untuk semua sumber daya dan metode tahap API

`update-stage` Contoh berikut mengaktifkan pencatatan permintaan/respons penuh untuk semua sumber daya dan metode tahap API.

```
aws apigateway update-stage \  
  --rest-api-id 1234123412 \  
  --stage-name 'dev' \  
  --patch-operations 'op=replace,path=/*/*/logging/dataTrace,value=true'
```

Output:

```
{  
  "deploymentId": "5ubd17",  
  "stageName": "dev",  
  "cacheClusterEnabled": false,  
  "cacheClusterStatus": "NOT_AVAILABLE",  
  "methodSettings": {  
    "/*/*": {  
      "metricsEnabled": false,  
      "dataTraceEnabled": true,  
      "throttlingBurstLimit": 5000,  
      "throttlingRateLimit": 10000.0,  
      "cachingEnabled": false,  
      "cacheTtlInSeconds": 300,  
      "cacheDataEncrypted": false,  
      "requireAuthorizationForCacheControl": true,  
      "unauthorizedCacheControlHeaderStrategy": "SUCCEED_WITH_RESPONSE_HEADER"  
    }  
  },  
  "tracingEnabled": false,  
  "createdDate": "2022-07-18T10:11:18-07:00",  
  "lastUpdatedDate": "2022-07-18T10:31:04-07:00"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan tahapan untuk REST API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateStage](#) di Referensi AWS CLI Perintah.

## update-usage-plan

Contoh kode berikut menunjukkan cara menggunakan `update-usage-plan`.



## AWS CLI

Untuk mengubah periode yang ditentukan dalam Rencana Penggunaan

Perintah:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
  op="replace",path="/quota/period",value="MONTH"
```

Untuk mengubah batas kuota yang ditentukan dalam Paket Penggunaan

Perintah:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
  op="replace",path="/quota/limit",value="500"
```

Untuk mengubah batas laju throttle yang ditentukan dalam Paket Penggunaan

Perintah:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
  op="replace",path="/throttle/rateLimit",value="10"
```

Untuk mengubah batas throttle burst yang ditentukan dalam Paket Penggunaan

Perintah:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
  op="replace",path="/throttle/burstLimit",value="20"
```

- Untuk detail API, lihat [UpdateUsagePlan](#) di Referensi AWS CLI Perintah.

## update-usage

Contoh kode berikut menunjukkan cara menggunakan `update-usage`.

## AWS CLI

Untuk memodifikasi sementara kuota pada kunci API untuk periode saat ini yang ditentukan dalam Paket Penggunaan

**Perintah:**

```
aws apigateway update-usage --usage-plan-id a1b2c3 --key-id
1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu --patch-operations op="replace",path="/
remaining",value="50"
```

- Untuk detail API, lihat [UpdateUsage](#) di Referensi AWS CLI Perintah.

## API Gateway HTTP dan contoh WebSocket API menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan HTTP dan WebSocket API API Gateway with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-api-mapping**

Contoh kode berikut menunjukkan cara menggunakan `create-api-mapping`.

### AWS CLI

Untuk membuat pemetaan API untuk API

`create-api-mapping` Contoh berikut memetakan test tahap API ke `/myApi` jalur nama domain `regional.example.com` kustom.

```
aws apigatewayv2 create-api-mapping \  
  --domain-name regional.example.com \  
  --api-mapping-key myApi \  
  --api-id a1b2c3d4 \  
  --stage test
```

Output:

```
{  
  "ApiId": "a1b2c3d4",  
  "ApiMappingId": "0qzs2sy7bh",  
  "ApiMappingKey": "myApi"  
  "Stage": "test"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [CreateApiMapping](#) di Referensi AWS CLI Perintah.

## create-api

Contoh kode berikut menunjukkan cara menggunakan `create-api`.

### AWS CLI

Untuk membuat API HTTP

`create-api` Contoh berikut membuat API HTTP dengan menggunakan quick create. Anda dapat menggunakan quick create untuk membuat API dengan integrasi AWS Lambda atau HTTP, rute tangkapan semua default, dan tahap default yang dikonfigurasi untuk menerapkan perubahan secara otomatis. Perintah berikut menggunakan quick create untuk membuat HTTP API yang terintegrasi dengan fungsi Lambda.

```
aws apigatewayv2 create-api \  
  --name my-http-api \  
  --protocol-type HTTP \  
  --target arn:aws:lambda:us-west-2:123456789012:function:my-lambda-function
```

Output:

```
{
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
  "ApiId": "a1b2c3d4",
  "ApiKeySelectionExpression": "$request.header.x-api-key",
  "CreateDate": "2020-04-08T19:05:45+00:00",
  "Name": "my-http-api",
  "ProtocolType": "HTTP",
  "RouteSelectionExpression": "$request.method $request.path"
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan API HTTP di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

Untuk membuat WebSocket API

`create-api`Contoh berikut membuat WebSocket API dengan nama yang ditentukan.

```
aws apigatewayv2 create-api \
  --name "myWebSocketApi" \
  --protocol-type WEBSOCKET \
  --route-selection-expression '$request.body.action'
```

Output:

```
{
  "ApiKeySelectionExpression": "$request.header.x-api-key",
  "Name": "myWebSocketApi",
  "CreateDate": "2018-11-15T06:23:51Z",
  "ProtocolType": "WEBSOCKET",
  "RouteSelectionExpression": "'$request.body.action'",
  "ApiId": "aabbccdde"
}
```

Untuk informasi selengkapnya, lihat [Membuat WebSocket API di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [CreateApi](#) di Referensi AWS CLI Perintah.

## **create-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `create-authorizer`.

## AWS CLI

Untuk membuat otorisasi JWT untuk API HTTP

`create-authorizer` Contoh berikut membuat otorisasi JWT yang menggunakan Amazon Cognito sebagai penyedia identitas.

```
aws apigatewayv2 create-authorizer \  
  --name my-jwt-authorizer \  
  --api-id a1b2c3d4 \  
  --authorizer-type JWT \  
  --identity-source '$request.header.Authorization' \  
  --jwt-configuration Audience=123456abc,Issuer=https://cognito-idp.us-  
west-2.amazonaws.com/us-west-2_abc123
```

Output:

```
{  
  "AuthorizerId": "a1b2c3",  
  "AuthorizerType": "JWT",  
  "IdentitySource": [  
    "$request.header.Authorization"  
  ],  
  "JwtConfiguration": {  
    "Audience": [  
      "123456abc"  
    ],  
    "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"  
  },  
  "Name": "my-jwt-authorizer"  
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses ke API HTTP dengan otorisasi JWT di Panduan Pengembang](#) Amazon API Gateway.

- Untuk detail API, lihat [CreateAuthorizer](#) di Referensi AWS CLI Perintah.

## create-deployment

Contoh kode berikut menunjukkan cara menggunakan `create-deployment`.

## AWS CLI

Untuk membuat penerapan untuk API

`create-deployment` Contoh berikut membuat penerapan untuk API dan mengaitkan penerapan tersebut dengan dev tahap API.

```
aws apigatewayv2 create-deployment \  
  --api-id a1b2c3d4 \  
  --stage-name dev
```

Output:

```
{  
  "AutoDeployed": false,  
  "CreateDate": "2020-04-06T23:38:08Z",  
  "DeploymentId": "531z91",  
  "DeploymentStatus": "DEPLOYED"  
}
```

Untuk informasi selengkapnya, lihat [penerapan API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [CreateDeployment](#) di Referensi AWS CLI Perintah.

## `create-domain-name`

Contoh kode berikut menunjukkan cara menggunakan `create-domain-name`.

## AWS CLI

Untuk membuat nama domain kustom

`create-domain-name` Contoh berikut membuat nama domain kustom regional untuk API.

```
aws apigatewayv2 create-domain-name \  
  --domain-name regional.example.com \  
  --domain-name-configurations CertificateArn=arn:aws:acm:us-  
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678
```

Output:

```
{
  "ApiMappingSelectionExpression": "$request.basepath",
  "DomainName": "regional.example.com",
  "DomainNameConfigurations": [
    {
      "ApiGatewayDomainName": "d-id.execute-api.us-west-2.amazonaws.com",
      "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
      "EndpointType": "REGIONAL",
      "HostedZoneId": "123456789111",
      "SecurityPolicy": "TLS_1_2",
      "DomainNameStatus": "AVAILABLE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [CreateDomainName](#) di Referensi AWS CLI Perintah.

## create-integration

Contoh kode berikut menunjukkan cara menggunakan `create-integration`.

### AWS CLI

Untuk membuat integrasi WebSocket API

`create-integration` Contoh berikut membuat integrasi tiruan untuk WebSocket API.

```
aws apigatewayv2 create-integration \
  --api-id aabbccdde \
  --passthrough-behavior WHEN_NO_MATCH \
  --timeout-in-millis 29000 \
  --connection-type INTERNET \
  --integration-type MOCK
```

Output:

```
{
  "ConnectionType": "INTERNET",
```

```
"IntegrationId": "0abcdef",
"IntegrationResponseSelectionExpression": "${integration.response.statuscode}",
"IntegrationType": "MOCK",
"PassthroughBehavior": "WHEN_NO_MATCH",
"PayloadFormatVersion": "1.0",
"TimeoutInMillis": 29000
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan permintaan integrasi WebSocket API di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

Untuk membuat integrasi HTTP API

`create-integration` berikut membuat integrasi AWS Lambda untuk HTTP API.

```
aws apigatewayv2 create-integration \
  --api-id a1b2c3d4 \
  --integration-type AWS_PROXY \
  --integration-uri arn:aws:lambda:us-west-2:123456789012:function:my-function \
  --payload-format-version 2.0
```

Output:

```
{
  "ConnectionType": "INTERNET",
  "IntegrationId": "0abcdef",
  "IntegrationMethod": "POST",
  "IntegrationType": "AWS_PROXY",
  "IntegrationUri": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
  "PayloadFormatVersion": "2.0",
  "TimeoutInMillis": 30000
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi integrasi untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [CreateIntegration](#) di Referensi AWS CLI Perintah.

## **create-route**

Contoh kode berikut menunjukkan cara menggunakan `create-route`.



## AWS CLI

Untuk membuat rute `$default` untuk API WebSocket atau HTTP

`create-route` Contoh berikut membuat `$default` rute untuk API WebSocket atau HTTP.

```
aws apigatewayv2 create-route \  
  --api-id aabbccdde \\  
  --route-key '$default'
```

Output:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteKey": "$default",  
  "RouteId": "1122334"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan rute untuk WebSocket API](#) di Panduan Pengembang Amazon API Gateway

Untuk membuat rute untuk HTTP API

`create-route` Contoh berikut membuat rute bernama `signup` yang menerima permintaan POST.

```
aws apigatewayv2 create-route \  
  --api-id aabbccdde \\  
  --route-key 'POST /signup'
```

Output:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteKey": "POST /signup",  
  "RouteId": "1122334"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan rute untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway

- Untuk detail API, lihat [CreateRoute](#) di Referensi AWS CLI Perintah.

## create-stage

Contoh kode berikut menunjukkan cara menggunakan `create-stage`.

### AWS CLI

Untuk membuat panggung

`create-stage` Contoh berikut membuat tahap bernama `dev` untuk API.

```
aws apigatewayv2 create-stage \  
  --api-id a1b2c3d4 \  
  --stage-name dev
```

Output:

```
{  
  "CreateDate": "2020-04-06T23:23:46Z",  
  "DefaultRouteSettings": {  
    "DetailedMetricsEnabled": false  
  },  
  "LastUpdatedDate": "2020-04-06T23:23:46Z",  
  "RouteSettings": {},  
  "StageName": "dev",  
  "StageVariables": {},  
  "Tags": {}  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tahapan untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [CreateStaged](#) di Referensi AWS CLI Perintah.

## create-vpc-link

Contoh kode berikut menunjukkan cara menggunakan `create-vpc-link`.

### AWS CLI

Untuk membuat link VPC untuk HTTP API

`create-vpc-link` Contoh berikut membuat link VPC untuk HTTP API.

```
aws apigatewayv2 create-vpc-link \  
  --name MyVpcLink \  
  --subnet-ids subnet-aaaa subnet-bbbb \  
  --security-group-ids sg1234 sg5678
```

Output:

```
{  
  "CreateDate": "2020-04-07T00:11:46Z",  
  "Name": "MyVpcLink",  
  "SecurityGroupIds": [  
    "sg1234",  
    "sg5678"  
  ],  
  "SubnetIds": [  
    "subnet-aaaa",  
    "subnet-bbbb"  
  ],  
  "Tags": {},  
  "VpcLinkId": "abcd123",  
  "VpcLinkStatus": "PENDING",  
  "VpcLinkStatusMessage": "VPC link is provisioning ENIs",  
  "VpcLinkVersion": "V2"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tautan VPC untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [CreateVpcLink](#) di Referensi AWS CLI Perintah.

## `delete-access-log-settings`

Contoh kode berikut menunjukkan cara menggunakan `delete-access-log-settings`.

AWS CLI

Untuk menonaktifkan pencatatan akses untuk API

`delete-access-log-settings` Contoh berikut menghapus pengaturan log akses untuk `$default` tahap API. Untuk menonaktifkan pencatatan akses untuk suatu tahap, hapus pengaturan log aksesnya.

```
aws apigatewayv2 delete-access-log-settings \  
  --api-id a1b2c3d4 \  
  --stage-name '$default'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi logging untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteAccessLogSettings](#) di Referensi AWS CLI Perintah.

## **delete-api-mapping**

Contoh kode berikut menunjukkan cara menggunakan `delete-api-mapping`.

### AWS CLI

Untuk menghapus pemetaan API

`delete-api-mapping` Contoh berikut menghapus pemetaan API untuk nama domain `api.example.com` kustom.

```
aws apigatewayv2 delete-api-mapping \  
  --api-mapping-id a1b2c3 \  
  --domain-name api.example.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteApiMapping](#) di Referensi AWS CLI Perintah.

## **delete-api**

Contoh kode berikut menunjukkan cara menggunakan `delete-api`.

## AWS CLI

Untuk menghapus API

`delete-api` Contoh berikut menghapus API.

```
aws apigatewayv2 delete-api \  
  --api-id a1b2c3d4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan API HTTP](#) dan [Bekerja dengan WebSocket API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteApi](#) di Referensi AWS CLI Perintah.

## `delete-authorizer`

Contoh kode berikut menunjukkan cara menggunakan `delete-authorizer`.

## AWS CLI

Untuk menghapus otorisasi

`delete-authorizer` Contoh berikut menghapus otorisasi.

```
aws apigatewayv2 delete-authorizer \  
  --api-id a1b2c3d4 \  
  --authorizer-id a1b2c3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengontrol akses ke API HTTP dengan otorisasi JWT di Panduan Pengembang](#) Amazon API Gateway.

- Untuk detail API, lihat [DeleteAuthorizer](#) di Referensi AWS CLI Perintah.

## `delete-cors-configuration`

Contoh kode berikut menunjukkan cara menggunakan `delete-cors-configuration`.

## AWS CLI

Untuk menghapus konfigurasi CORS untuk HTTP API

`delete-cors-configuration` Contoh berikut menonaktifkan CORS untuk HTTP API dengan menghapus konfigurasi CORS-nya.

```
aws apigatewayv2 delete-cors-configuration \  
  --api-id a1b2c3d4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi CORS untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [DeleteCorsConfiguration](#) di Referensi AWS CLI Perintah.

## `delete-deployment`

Contoh kode berikut menunjukkan cara menggunakan `delete-deployment`.

## AWS CLI

Untuk menghapus penerapan

`delete-deployment` Contoh berikut menghapus penerapan API.

```
aws apigatewayv2 delete-deployment \  
  --api-id a1b2c3d4 \  
  --deployment-id a1b2c3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [penerapan API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteDeployment](#) di Referensi AWS CLI Perintah.

## `delete-domain-name`

Contoh kode berikut menunjukkan cara menggunakan `delete-domain-name`.

## AWS CLI

Untuk menghapus nama domain kustom

`delete-domain-name` Contoh berikut menghapus nama domain kustom.

```
aws apigatewayv2 delete-domain-name \  
  --domain-name api.example.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteDomainName](#) di Referensi AWS CLI Perintah.

## **delete-integration**

Contoh kode berikut menunjukkan cara menggunakan `delete-integration`.

## AWS CLI

Untuk menghapus integrasi

`delete-integration` Contoh berikut menghapus integrasi API.

```
aws apigatewayv2 delete-integration \  
  --api-id a1b2c3d4 \  
  --integration-id a1b2c3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi integrasi untuk API HTTP](#) dan [Menyiapkan integrasi WebSocket API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteIntegration](#) di Referensi AWS CLI Perintah.

## **delete-route-settings**

Contoh kode berikut menunjukkan cara menggunakan `delete-route-settings`.

## AWS CLI

Untuk menghapus pengaturan rute

`delete-route-settings` Contoh berikut menghapus pengaturan rute untuk rute yang ditentukan.

```
aws apigatewayv2 delete-route-settings \  
  --api-id a1b2c3d4 \  
  --stage-name dev \  
  --route-key 'GET /pets'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan rute untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteRouteSettings](#) di Referensi AWS CLI Perintah.

## **delete-route**

Contoh kode berikut menunjukkan cara menggunakan `delete-route`.

## AWS CLI

Untuk menghapus rute

`delete-route` Contoh berikut menghapus rute API.

```
aws apigatewayv2 delete-route \  
  --api-id a1b2c3d4 \  
  --route-id a1b2c3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan rute untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteRoute](#) di Referensi AWS CLI Perintah.

## **delete-stage**

Contoh kode berikut menunjukkan cara menggunakan `delete-stage`.



## AWS CLI

Untuk menghapus panggung

`delete-stage` Contoh berikut menghapus test tahap API.

```
aws apigatewayv2 delete-stage \  
  --api-id a1b2c3d4 \  
  --stage-name test
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan tahapan untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteStage](#) di Referensi AWS CLI Perintah.

## `delete-vpc-link`

Contoh kode berikut menunjukkan cara menggunakan `delete-vpc-link`.

## AWS CLI

Untuk menghapus tautan VPC untuk API HTTP

`delete-vpc-link` Contoh berikut menghapus link VPC.

```
aws apigatewayv2 delete-vpc-link \  
  --vpc-link-id abcd123
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan tautan VPC untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [DeleteVpcLink](#) di Referensi AWS CLI Perintah.

## `export-api`

Contoh kode berikut menunjukkan cara menggunakan `export-api`.

## AWS CLI

Untuk mengekspor definisi OpenAPI dari HTTP API

`export-api` Contoh berikut mengekspor definisi OpenAPI 3.0 dari tahap API bernama ke file YAMAL `prod` bernama `stage-definition.yaml`. File definisi yang diekspor menyertakan ekstensi API Gateway secara default.

```
aws apigatewayv2 export-api \  
  --api-id a1b2c3d4 \  
  --output-type YAML \  
  --specification OAS30 \  
  --stage-name prod \  
  stage-definition.yaml
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengekspor API HTTP dari API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [ExportApi](#) di Referensi AWS CLI Perintah.

## `get-api-mapping`

Contoh kode berikut menunjukkan cara menggunakan `get-api-mapping`.

## AWS CLI

Untuk mendapatkan informasi tentang pemetaan API untuk nama domain kustom

`get-api-mapping` Contoh berikut menampilkan informasi tentang pemetaan API untuk nama domain `api.example.com` kustom.

```
aws apigatewayv2 get-api-mapping \  
  --api-mapping-id a1b2c3 \  
  --domain-name api.example.com
```

Output:

```
{  
  "ApiId": "a1b2c3d4",
```

```
"ApiMappingId": "a1b2c3d5",  
"ApiMappingKey": "myTestApi"  
"Stage": "test"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetApiMapping](#) di Referensi AWS CLI Perintah.

## get-api-mappings

Contoh kode berikut menunjukkan cara menggunakan `get-api-mappings`.

### AWS CLI

Untuk mendapatkan pemetaan API untuk nama domain khusus

`get-api-mappings` Contoh berikut menampilkan daftar semua pemetaan API untuk nama domain `api.example.com` kustom.

```
aws apigatewayv2 get-api-mappings \  
  --domain-name api.example.com
```

Output:

```
{  
  "Items": [  
    {  
      "ApiId": "a1b2c3d4",  
      "ApiMappingId": "a1b2c3d5",  
      "ApiMappingKey": "myTestApi"  
      "Stage": "test"  
    },  
    {  
      "ApiId": "a5b6c7d8",  
      "ApiMappingId": "a1b2c3d6",  
      "ApiMappingKey": "myDevApi"  
      "Stage": "dev"  
    },  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetApiMappings](#) di Referensi AWS CLI Perintah.

## get-api

Contoh kode berikut menunjukkan cara menggunakan `get-api`.

### AWS CLI

Untuk mengambil informasi tentang API

`get-api` Contoh berikut menampilkan informasi tentang API.

```
aws apigatewayv2 get-api \  
  --api-id a1b2c3d4
```

Output:

```
{  
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",  
  "ApiId": "a1b2c3d4",  
  "ApiKeySelectionExpression": "$request.header.x-api-key",  
  "CreateDate": "2020-03-28T00:32:37Z",  
  "Name": "my-api",  
  "ProtocolType": "HTTP",  
  "RouteSelectionExpression": "$request.method $request.path",  
  "Tags": {  
    "department": "finance"  
  }  
}
```

- Untuk detail API, lihat [GetApi](#) di Referensi AWS CLI Perintah.

## get-apis

Contoh kode berikut menunjukkan cara menggunakan `get-apis`.

### AWS CLI

Untuk mengambil daftar API

`get-apis`Contoh berikut mencantumkan semua API untuk pengguna saat ini.

```
aws apigatewayv2 get-apis
```

Output:

```
{
  "Items": [
    {
      "ApiEndpoint": "wss://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
      "ApiId": "a1b2c3d4",
      "ApiKeySelectionExpression": "$request.header.x-api-key",
      "CreateDate": "2020-04-07T20:21:59Z",
      "Name": "my-websocket-api",
      "ProtocolType": "WEBSOCKET",
      "RouteSelectionExpression": "$request.body.message",
      "Tags": {}
    },
    {
      "ApiEndpoint": "https://a1b2c3d5.execute-api.us-west-2.amazonaws.com",
      "ApiId": "a1b2c3d5",
      "ApiKeySelectionExpression": "$request.header.x-api-key",
      "CreateDate": "2020-04-07T20:23:50Z",
      "Name": "my-http-api",
      "ProtocolType": "HTTP",
      "RouteSelectionExpression": "$request.method $request.path",
      "Tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan API HTTP](#) dan [Bekerja dengan WebSocket API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetApis](#) di Referensi AWS CLI Perintah.

## **get-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `get-authorizer`.

### AWS CLI

Untuk mengambil informasi tentang otorisasi

`get-authorizer` Contoh berikut menampilkan informasi tentang authorizer.

```
aws apigatewayv2 get-authorizer \  
  --api-id a1b2c3d4 \  
  --authorizer-id a1b2c3
```

Output:

```
{  
  "AuthorizerId": "a1b2c3",  
  "AuthorizerType": "JWT",  
  "IdentitySource": [  
    "$request.header.Authorization"  
  ],  
  "JwtConfiguration": {  
    "Audience": [  
      "123456abc"  
    ],  
    "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"  
  },  
  "Name": "my-jwt-authorizer"  
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses ke API HTTP dengan otorisasi JWT di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [GetAuthorizer](#) di Referensi AWS CLI Perintah.

## **get-authorizers**

Contoh kode berikut menunjukkan cara menggunakan `get-authorizers`.

AWS CLI

Untuk mengambil daftar otorisasi untuk API

`get-authorizers` Contoh berikut menampilkan daftar semua otorisasi untuk API.

```
aws apigatewayv2 get-authorizers \  
  --api-id a1b2c3d4
```

Output:

```
{
  "Items": [
    {
      "AuthorizerId": "a1b2c3",
      "AuthorizerType": "JWT",
      "IdentitySource": [
        "$request.header.Authorization"
      ],
      "JwtConfiguration": {
        "Audience": [
          "123456abc"
        ],
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_abc123"
      },
      "Name": "my-jwt-authorizer"
    },
    {
      "AuthorizerId": "a1b2c4",
      "AuthorizerType": "JWT",
      "IdentitySource": [
        "$request.header.Authorization"
      ],
      "JwtConfiguration": {
        "Audience": [
          "6789abcde"
        ],
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_abc234"
      },
      "Name": "new-jwt-authorizer"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses ke API HTTP dengan otorisasi JWT di Panduan Pengembang](#) Amazon API Gateway.

- Untuk detail API, lihat [GetAuthorizers](#) di Referensi AWS CLI Perintah.

## get-deployment

Contoh kode berikut menunjukkan cara menggunakan `get-deployment`.

## AWS CLI

Untuk mengambil informasi tentang penerapan

`get-deployment` Contoh berikut menampilkan informasi tentang penyebaran.

```
aws apigatewayv2 get-deployment \  
  --api-id a1b2c3d4 \  
  --deployment-id abcdef
```

Output:

```
{  
  "AutoDeployed": true,  
  "CreatedDate": "2020-04-07T23:58:40Z",  
  "DeploymentId": "abcdef",  
  "DeploymentStatus": "DEPLOYED",  
  "Description": "Automatic deployment triggered by changes to the Api  
  configuration"  
}
```

Untuk informasi selengkapnya, lihat [penerapan API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetDeployment](#) di Referensi AWS CLI Perintah.

## `get-deployments`

Contoh kode berikut menunjukkan cara menggunakan `get-deployments`.

### AWS CLI

Untuk mengambil daftar penerapan

`get-deployments` Contoh berikut menampilkan daftar semua penerapan API.

```
aws apigatewayv2 get-deployments \  
  --api-id a1b2c3d4
```

Output:



```
{
  "Items": [
    {
      "AutoDeployed": true,
      "CreatedDate": "2020-04-07T23:58:40Z",
      "DeploymentId": "abcdef",
      "DeploymentStatus": "DEPLOYED",
      "Description": "Automatic deployment triggered by changes to the Api
configuration"
    },
    {
      "AutoDeployed": true,
      "CreatedDate": "2020-04-06T00:33:00Z",
      "DeploymentId": "bcdefg",
      "DeploymentStatus": "DEPLOYED",
      "Description": "Automatic deployment triggered by changes to the Api
configuration"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [penerapan API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetDeployments](#) di Referensi AWS CLI Perintah.

## get-domain-name

Contoh kode berikut menunjukkan cara menggunakan `get-domain-name`.

### AWS CLI

Untuk mengambil informasi tentang nama domain kustom

`get-domain-name` Contoh berikut menampilkan informasi tentang nama domain kustom.

```
aws apigatewayv2 get-domain-name \
  --domain-name api.example.com
```

Output:

```
{
```

```

"ApiMappingSelectionExpression": "$request.basepath",
"DomainName": "api.example.com",
"DomainNameConfigurations": [
  {
    "ApiGatewayDomainName": "d-1234.execute-api.us-west-2.amazonaws.com",
    "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
    "EndpointType": "REGIONAL",
    "HostedZoneId": "123456789111",
    "SecurityPolicy": "TLS_1_2",
    "DomainNameStatus": "AVAILABLE"
  }
],
"Tags": {}
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetDomainName](#) di Referensi AWS CLI Perintah.

## get-domain-names

Contoh kode berikut menunjukkan cara menggunakan `get-domain-names`.

### AWS CLI

Untuk mengambil daftar nama domain kustom

`get-domain-names` Contoh berikut menampilkan daftar semua nama domain kustom untuk pengguna saat ini.

```
aws apigatewayv2 get-domain-names
```

Output:

```

{
  "Items": [
    {
      "ApiMappingSelectionExpression": "$request.basepath",
      "DomainName": "api.example.com",
      "DomainNameConfigurations": [

```

```

        {
            "ApiGatewayDomainName": "d-1234.execute-api.us-
west-2.amazonaws.com",
            "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
            "EndpointType": "REGIONAL",
            "HostedZoneId": "123456789111",
            "SecurityPolicy": "TLS_1_2",
            "DomainNameStatus": "AVAILABLE"
        }
    ],
    {
        "ApiMappingSelectionExpression": "$request.basepath",
        "DomainName": "newApi.example.com",
        "DomainNameConfigurations": [
            {
                "ApiGatewayDomainName": "d-5678.execute-api.us-
west-2.amazonaws.com",
                "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
                "EndpointType": "REGIONAL",
                "HostedZoneId": "123456789222",
                "SecurityPolicy": "TLS_1_2",
                "DomainNameStatus": "AVAILABLE"
            }
        ]
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetDomainNames](#) di Referensi AWS CLI Perintah.

## get-integration

Contoh kode berikut menunjukkan cara menggunakan `get-integration`.

### AWS CLI

Untuk mengambil informasi tentang integrasi

`get-integration` Contoh berikut menampilkan informasi tentang integrasi.

```
aws apigatewayv2 get-integration \  
  --api-id a1b2c3d4 \  
  --integration-id a1b2c3
```

Output:

```
{  
  "ApiGatewayManaged": true,  
  "ConnectionType": "INTERNET",  
  "IntegrationId": "a1b2c3",  
  "IntegrationMethod": "POST",  
  "IntegrationType": "AWS_PROXY",  
  "IntegrationUri": "arn:aws:lambda:us-west-2:12356789012:function:hello12",  
  "PayloadFormatVersion": "2.0",  
  "TimeoutInMillis": 30000  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi integrasi untuk API HTTP](#) dan [Menyiapkan integrasi WebSocket API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetIntegration](#) di Referensi AWS CLI Perintah.

## `get-integrations`

Contoh kode berikut menunjukkan cara menggunakan `get-integrations`.

AWS CLI

Untuk mengambil daftar integrasi

`get-integrations` Contoh berikut menampilkan daftar semua integrasi API.

```
aws apigatewayv2 get-integrations \  
  --api-id a1b2c3d4
```

Output:

```
{  
  "Items": [  
    {  
      "IntegrationId": "a1b2c3",  
      "IntegrationMethod": "POST",  
      "IntegrationType": "AWS_PROXY",  
      "IntegrationUri": "arn:aws:lambda:us-west-2:12356789012:function:hello12",  
      "PayloadFormatVersion": "2.0",  
      "TimeoutInMillis": 30000  
    }  
  ]  
}
```

```

    {
      "ApiGatewayManaged": true,
      "ConnectionType": "INTERNET",
      "IntegrationId": "a1b2c3",
      "IntegrationMethod": "POST",
      "IntegrationType": "AWS_PROXY",
      "IntegrationUri": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
      "PayloadFormatVersion": "2.0",
      "TimeoutInMillis": 30000
    },
    {
      "ConnectionType": "INTERNET",
      "IntegrationId": "a1b2c4",
      "IntegrationMethod": "ANY",
      "IntegrationType": "HTTP_PROXY",
      "IntegrationUri": "https://www.example.com",
      "PayloadFormatVersion": "1.0",
      "TimeoutInMillis": 30000
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Mengonfigurasi integrasi untuk API HTTP](#) dan [Menyiapkan integrasi WebSocket API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetIntegrations](#) di Referensi AWS CLI Perintah.

## get-route

Contoh kode berikut menunjukkan cara menggunakan `get-route`.

### AWS CLI

Untuk mengambil informasi tentang rute

`get-route` Contoh berikut menampilkan informasi tentang rute.

```

aws apigatewayv2 get-route \
  --api-id a1b2c3d4 \
  --route-id 72jz1wk

```

Output:

```
{
  "ApiKeyRequired": false,
  "AuthorizationType": "NONE",
  "RouteId": "72jz1wk",
  "RouteKey": "ANY /pets",
  "Target": "integrations/a1b2c3"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan rute untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway

- Untuk detail API, lihat [GetRouted](#) di Referensi AWS CLI Perintah.

## get-routes

Contoh kode berikut menunjukkan cara menggunakan `get-routes`.

### AWS CLI

Untuk mengambil daftar rute

`get-routes` Contoh berikut menampilkan daftar semua rute API.

```
aws apigatewayv2 get-routes \
  --api-id a1b2c3d4
```

Output:

```
{
  "Items": [
    {
      "ApiKeyRequired": false,
      "AuthorizationType": "NONE",
      "RouteId": "72jz1wk",
      "RouteKey": "ANY /admin",
      "Target": "integrations/a1b2c3"
    },
    {
      "ApiGatewayManaged": true,
      "ApiKeyRequired": false,
      "AuthorizationType": "NONE",
```

```
        "RouteId": "go65gqi",
        "RouteKey": "$default",
        "Target": "integrations/a1b2c4"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan rute untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway

- Untuk detail API, lihat [GetRoutes](#) di Referensi AWS CLI Perintah.

## get-stage

Contoh kode berikut menunjukkan cara menggunakan `get-stage`.

### AWS CLI

Untuk mengambil informasi tentang suatu panggung

`get-stage` Contoh berikut menampilkan informasi tentang `prod` tahap API.

```
aws apigatewayv2 get-stage \
  --api-id a1b2c3d4 \
  --stage-name prod
```

Output:

```
{
  "CreateDate": "2020-04-08T00:36:05Z",
  "DefaultRouteSettings": {
    "DetailedMetricsEnabled": false
  },
  "DeploymentId": "x1zwyv",
  "LastUpdatedDate": "2020-04-08T00:36:13Z",
  "RouteSettings": {},
  "StageName": "prod",
  "StageVariables": {
    "function": "my-prod-function"
  },
  "Tags": {}
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tahapan untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetStage](#) di Referensi AWS CLI Perintah.

## get-stages

Contoh kode berikut menunjukkan cara menggunakan `get-stages`.

### AWS CLI

Untuk mengambil daftar tahapan

`get-stages` Contoh berikut mencantumkan semua tahapan API.

```
aws apigatewayv2 get-stages \  
  --api-id a1b2c3d4
```

Output:

```
{  
  "Items": [  
    {  
      "ApiGatewayManaged": true,  
      "AutoDeploy": true,  
      "CreateDate": "2020-04-08T00:08:44Z",  
      "DefaultRouteSettings": {  
        "DetailedMetricsEnabled": false  
      },  
      "DeploymentId": "dty748",  
      "LastDeploymentStatusMessage": "Successfully deployed stage with  
deployment ID 'dty748'",  
      "LastUpdatedDate": "2020-04-08T00:09:49Z",  
      "RouteSettings": {},  
      "StageName": "$default",  
      "StageVariables": {},  
      "Tags": {}  
    },  
    {  
      "AutoDeploy": true,  
      "CreateDate": "2020-04-08T00:35:06Z",  
      "DefaultRouteSettings": {  
        "DetailedMetricsEnabled": false  
      }  
    }  
  ]  
}
```



```

    },
    "LastUpdatedDate": "2020-04-08T00:35:48Z",
    "RouteSettings": {},
    "StageName": "dev",
    "StageVariables": {
      "function": "my-dev-function"
    },
    "Tags": {}
  },
  {
    "CreatedDate": "2020-04-08T00:36:05Z",
    "DefaultRouteSettings": {
      "DetailedMetricsEnabled": false
    },
    "DeploymentId": "x1zwyv",
    "LastUpdatedDate": "2020-04-08T00:36:13Z",
    "RouteSettings": {},
    "StageName": "prod",
    "StageVariables": {
      "function": "my-prod-function"
    },
    "Tags": {}
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan tahapan untuk API HTTP](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetStages](#) di Referensi AWS CLI Perintah.

## get-tags

Contoh kode berikut menunjukkan cara menggunakan `get-tags`.

### AWS CLI

Untuk mengambil daftar tag untuk sumber daya

`get-tags` Contoh berikut mencantumkan semua tag API.

```
aws apigatewayv2 get-tags \
  --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4
```

**Output:**

```
{
  "Tags": {
    "owner": "dev-team",
    "environment": "prod"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai resource API Gateway Anda](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetTags](#) di Referensi AWS CLI Perintah.

**get-vpc-link**

Contoh kode berikut menunjukkan cara menggunakan `get-vpc-link`.

**AWS CLI**

Untuk mengambil informasi tentang tautan VPC

`get-vpc-link` Contoh berikut menampilkan informasi tentang link VPC.

```
aws apigatewayv2 get-vpc-link \
  --vpc-link-id abcd123
```

**Output:**

```
{
  "CreateDate": "2020-04-07T00:27:47Z",
  "Name": "MyVpcLink",
  "SecurityGroupIds": [
    "sg1234",
    "sg5678"
  ],
  "SubnetIds": [
    "subnet-aaaa",
    "subnet-bbbb"
  ],
  "Tags": {},
  "VpcLinkId": "abcd123",
  "VpcLinkStatus": "AVAILABLE",
```

```
"VpcLinkStatusMessage": "VPC link is ready to route traffic",
"VpcLinkVersion": "V2"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tautan VPC untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [GetVpcLink](#) di Referensi AWS CLI Perintah.

## get-vpc-links

Contoh kode berikut menunjukkan cara menggunakan `get-vpc-links`.

### AWS CLI

Untuk mengambil daftar tautan VPC

`get-vpc-links` Contoh berikut menampilkan daftar semua link VPC untuk pengguna saat ini.

```
aws apigatewayv2 get-vpc-links
```

Output:

```
{
  "Items": [
    {
      "CreateDate": "2020-04-07T00:27:47Z",
      "Name": "MyVpcLink",
      "SecurityGroupIds": [
        "sg1234",
        "sg5678"
      ],
      "SubnetIds": [
        "subnet-aaaa",
        "subnet-bbbb"
      ],
      "Tags": {},
      "VpcLinkId": "abcd123",
      "VpcLinkStatus": "AVAILABLE",
      "VpcLinkStatusMessage": "VPC link is ready to route traffic",
      "VpcLinkVersion": "V2"
    }
  ]
}
```

```

    "CreateDate": "2020-04-07T00:27:47Z",
    "Name": "MyOtherVpcLink",
    "SecurityGroupIds": [
      "sg1234",
      "sg5678"
    ],
    "SubnetIds": [
      "subnet-aaaa",
      "subnet-bbbb"
    ],
    "Tags": {},
    "VpcLinkId": "abcd456",
    "VpcLinkStatus": "AVAILABLE",
    "VpcLinkStatusMessage": "VPC link is ready to route traffic",
    "VpcLinkVersion": "V2"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan tautan VPC untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [GetVpcLinks](#) di Referensi AWS CLI Perintah.

## import-api

Contoh kode berikut menunjukkan cara menggunakan `import-api`.

### AWS CLI

Untuk mengimpor API HTTP

`import-api` Contoh berikut membuat API HTTP dari file definisi OpenAPI 3.0 bernama `api-definition.yaml`

```
aws apigatewayv2 import-api \
  --body file://api-definition.yaml
```

Isi dari `api-definition.yaml`:

```
openapi: 3.0.1
info:
  title: My Lambda API
```

```

    version: v1.0
  paths:
    /hello:
      x-amazon-apigateway-any-method:
        x-amazon-apigateway-integration:
          payloadFormatVersion: 2.0
          type: aws_proxy
          httpMethod: POST
          uri: arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123456789012:function:hello/invocations
          connectionType: INTERNET

```

### Output:

```

{
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
  "ApiId": "a1b2c3d4",
  "ApiKeySelectionExpression": "$request.header.x-api-key",
  "CreateDate": "2020-04-08T17:19:38+00:00",
  "Name": "My Lambda API",
  "ProtocolType": "HTTP",
  "RouteSelectionExpression": "$request.method $request.path",
  "Tags": {},
  "Version": "v1.0"
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan definisi OpenAPI untuk API HTTP di Panduan](#) Pengembang Amazon API Gateway.

- Untuk detail API, lihat [ImportApi](#) di Referensi AWS CLI Perintah.

## reimport-api

Contoh kode berikut menunjukkan cara menggunakan `reimport-api`.

### AWS CLI

Untuk mengimpor ulang API HTTP

`reimport-api` Contoh berikut memperbarui API HTTP yang ada untuk menggunakan definisi OpenAPI 3.0 yang ditentukan dalam `api-definition.yaml`

```
aws apigatewayv2 reimport-api \
```

```
--body file://api-definition.yaml \  
--api-id a1b2c3d4
```

Isi dari `api-definition.yaml`:

```
openapi: 3.0.1  
info:  
  title: My Lambda API  
  version: v1.0  
paths:  
  /hello:  
    x-amazon-apigateway-any-method:  
      x-amazon-apigateway-integration:  
        payloadFormatVersion: 2.0  
        type: aws_proxy  
        httpMethod: POST  
        uri: arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/  
arn:aws:lambda:us-west-2:12356789012:function:hello/invocations  
        connectionType: INTERNET
```

Output:

```
{  
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",  
  "ApiId": "a1b2c3d4",  
  "ApiKeySelectionExpression": "$request.header.x-api-key",  
  "CreateDate": "2020-04-08T17:19:38+00:00",  
  "Name": "My Lambda API",  
  "ProtocolType": "HTTP",  
  "RouteSelectionExpression": "$request.method $request.path",  
  "Tags": {},  
  "Version": "v1.0"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan definisi OpenAPI untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [ReimportApi](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan tag dengan nama kunci `Department` dan nilai `Accounting` ke API yang ditentukan.

```
aws apigatewayv2 tag-resource \  
  --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4 \  
  --tags Department=Accounting
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai resource API Gateway Anda](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dengan nama kunci `Project` dan `Owner` dari API yang ditentukan.

```
aws apigatewayv2 untag-resource \  
  --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4 \  
  --tag-keys Project Owner
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai resource API Gateway Anda](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-api-mapping**

Contoh kode berikut menunjukkan cara menggunakan `update-api-mapping`.

## AWS CLI

Untuk memperbarui pemetaan API

`update-api-mapping` Contoh berikut mengubah pemetaan API untuk nama domain kustom. Akibatnya, URL dasar yang menggunakan nama domain khusus untuk API dan tahap yang ditentukan menjadi `https://api.example.com/dev`.

```
aws apigatewayv2 update-api-mapping \  
  --api-id a1b2c3d4 \  
  --stage dev \  
  --domain-name api.example.com \  
  --api-mapping-id 0qzs2sy7bh \  
  --api-mapping-key dev
```

Output:

```
{  
  "ApiId": "a1b2c3d4",  
  "ApiMappingId": "0qzs2sy7bh",  
  "ApiMappingKey": "dev"  
  "Stage": "dev"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateApiMapping](#) di Referensi AWS CLI Perintah.

## **update-api**

Contoh kode berikut menunjukkan cara menggunakan `update-api`.

## AWS CLI

Untuk mengaktifkan CORS untuk HTTP API

`update-api` Contoh berikut memperbarui konfigurasi CORS API yang ditentukan untuk mengizinkan permintaan dari `https://www.example.com`.

```
aws apigatewayv2 update-api \  
  --api-id a1b2c3d4 \  
  --stage dev \  
  --cors-configurations [CORS Configurations]
```



```
--api-id a1b2c3d4 \  
--cors-configuration AllowOrigins=https://www.example.com
```

### Output:

```
{  
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",  
  "ApiId": "a1b2c3d4",  
  "ApiKeySelectionExpression": "$request.header.x-api-key",  
  "CorsConfiguration": {  
    "AllowCredentials": false,  
    "AllowHeaders": [  
      "header1",  
      "header2"  
    ],  
    "AllowMethods": [  
      "GET",  
      "OPTIONS"  
    ],  
    "AllowOrigins": [  
      "https://www.example.com"  
    ]  
  },  
  "CreateDate": "2020-04-08T18:39:37+00:00",  
  "Name": "my-http-api",  
  "ProtocolType": "HTTP",  
  "RouteSelectionExpression": "$request.method $request.path",  
  "Tags": {},  
  "Version": "v1.0"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi CORS untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [UpdateApi](#) di Referensi AWS CLI Perintah.

## update-authorizer

Contoh kode berikut menunjukkan cara menggunakan `update-authorizer`.

### AWS CLI

Untuk memperbarui otorisasi

`update-authorizer` Contoh berikut mengubah sumber identitas otorisasi JWT menjadi header bernama. `Authorization`

```
aws apigatewayv2 update-authorizer \  
  --api-id a1b2c3d4 \  
  --authorizer-id a1b2c3 \  
  --identity-source '$request.header.Authorization'
```

Output:

```
{  
  "AuthorizerId": "a1b2c3",  
  "AuthorizerType": "JWT",  
  "IdentitySource": [  
    "$request.header.Authorization"  
  ],  
  "JwtConfiguration": {  
    "Audience": [  
      "123456abc"  
    ],  
    "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"  
  },  
  "Name": "my-jwt-authorizer"  
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses ke API HTTP dengan otorisasi JWT di Panduan Pengembang](#) Amazon API Gateway.

- Untuk detail API, lihat [UpdateAuthorizer](#) di Referensi AWS CLI Perintah.

## update-deployment

Contoh kode berikut menunjukkan cara menggunakan `update-deployment`.

AWS CLI

Untuk mengubah deskripsi penerapan

`update-deployment` Contoh berikut memperbarui deskripsi penerapan.

```
aws apigatewayv2 update-deployment \  
  --api-id a1b2c3d4 \  
  --description 'New description'
```

```
--deployment-id abcdef \  
--description 'Manual deployment to fix integration test failures.'
```

### Output:

```
{  
  "AutoDeployed": false,  
  "CreateDate": "2020-02-05T16:21:48+00:00",  
  "DeploymentId": "abcdef",  
  "DeploymentStatus": "DEPLOYED",  
  "Description": "Manual deployment to fix integration test failures."  
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan API HTTP di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateDeployment](#) di Referensi AWS CLI Perintah.

## update-domain-name

Contoh kode berikut menunjukkan cara menggunakan `update-domain-name`.

### AWS CLI

Untuk memperbarui nama domain kustom

`update-domain-name` Contoh berikut menentukan sertifikat ACM baru untuk nama domain `api.example.com` kustom.

```
aws apigatewayv2 update-domain-name \  
  --domain-name api.example.com \  
  --domain-name-configurations CertificateArn=arn:aws:acm:us-  
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678
```

### Output:

```
{  
  "ApiMappingSelectionExpression": "$request.basepath",  
  "DomainName": "regional.example.com",  
  "DomainNameConfigurations": [  
    {  
      "ApiGatewayDomainName": "d-id.execute-api.us-west-2.amazonaws.com",
```

```

    "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
    "EndpointType": "REGIONAL",
    "HostedZoneId": "123456789111",
    "SecurityPolicy": "TLS_1_2",
    "DomainNameStatus": "AVAILABLE"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan nama domain kustom regional di API Gateway](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateDomainName](#) di Referensi AWS CLI Perintah.

## update-integration

Contoh kode berikut menunjukkan cara menggunakan `update-integration`.

### AWS CLI

Untuk memperbarui integrasi Lambda

`update-integration` Contoh berikut memperbarui integrasi AWS Lambda yang ada untuk menggunakan fungsi Lambda yang ditentukan.

```

aws apigatewayv2 update-integration \
  --api-id a1b2c3d4 \
  --integration-id a1b2c3 \
  --integration-uri arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123456789012:function:my-new-function/invocations

```

Output:

```

{
  "ConnectionType": "INTERNET",
  "IntegrationId": "a1b2c3",
  "IntegrationMethod": "POST",
  "IntegrationType": "AWS_PROXY",
  "IntegrationUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/
functions/arn:aws:lambda:us-west-2:123456789012:function:my-new-function/
invocations",

```

```
"PayloadFormatVersion": "2.0",  
"TimeoutInMillis": 5000  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi integrasi untuk API HTTP](#) dan [Menyiapkan integrasi WebSocket API](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateIntegration](#) di Referensi AWS CLI Perintah.

## update-route

Contoh kode berikut menunjukkan cara menggunakan `update-route`.

### AWS CLI

Contoh 1: Untuk memperbarui integrasi rute

`update-route` Contoh berikut memperbarui integrasi rute tertentu.

```
aws apigatewayv2 update-route \  
  --api-id a1b2c3d4 \  
  --route-id a1b2c3 \  
  --target integrations/a1b2c6
```

Output:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteId": "a1b2c3",  
  "RouteKey": "ANY /pets",  
  "Target": "integrations/a1b2c6"  
}
```

Contoh 2: Untuk menambahkan otorisasi ke rute

`update-route` Contoh berikut memperbarui rute yang ditentukan untuk menggunakan otorisasi JWT.

```
aws apigatewayv2 update-route \  
  --api-id a1b2c3d4 \  
  --route-id a1b2c3 \  
  --target integrations/a1b2c6
```

```
--api-id a1b2c3d4 \  
--route-id a1b2c3 \  
--authorization-type JWT \  
--authorizer-id a1b2c5 \  
--authorization-scopes user.id user.email
```

### Output:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationScopes": [  
    "user.id",  
    "user.email"  
  ],  
  "AuthorizationType": "JWT",  
  "AuthorizerId": "a1b2c5",  
  "OperationName": "GET HTTP",  
  "RequestParameters": {},  
  "RouteId": "a1b2c3",  
  "RouteKey": "GET /pets",  
  "Target": "integrations/a1b2c6"  
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses ke API HTTP dengan otorisasi JWT di Panduan Pengembang](#) Amazon API Gateway.

- Untuk detail API, lihat [UpdateRoute](#) di Referensi AWS CLI Perintah.

## update-stage

Contoh kode berikut menunjukkan cara menggunakan `update-stage`.

### AWS CLI

Untuk mengonfigurasi pelambatan khusus

`update-stage` Contoh berikut mengonfigurasi pelambatan kustom untuk tahap dan rute API yang ditentukan.

```
aws apigatewayv2 update-stage \  
  --api-id a1b2c3d4 \  
  --stage-name dev \  
  --
```

```
--route-settings '{"GET /pets":
{"ThrottlingBurstLimit":100,"ThrottlingRateLimit":2000}}'
```

### Output:

```
{
  "CreateDate": "2020-04-05T16:21:16+00:00",
  "DefaultRouteSettings": {
    "DetailedMetricsEnabled": false
  },
  "DeploymentId": "shktxb",
  "LastUpdatedDate": "2020-04-08T22:23:17+00:00",
  "RouteSettings": {
    "GET /pets": {
      "ThrottlingBurstLimit": 100,
      "ThrottlingRateLimit": 2000.0
    }
  },
  "StageName": "dev",
  "StageVariables": {},
  "Tags": {}
}
```

Untuk informasi selengkapnya, lihat [Melindungi API HTTP Anda](#) di Panduan Pengembang Amazon API Gateway.

- Untuk detail API, lihat [UpdateStage](#) di Referensi AWS CLI Perintah.

## update-vpc-link

Contoh kode berikut menunjukkan cara menggunakan `update-vpc-link`.

### AWS CLI

Untuk memperbarui tautan VPC

`update-vpc-link` Contoh berikut memperbarui nama tautan VPC. Setelah membuat tautan VPC, Anda tidak dapat mengubah grup keamanan atau subnetnya.

```
aws apigatewayv2 update-vpc-link \
  --vpc-link-id abcd123 \
  --name MyUpdatedVpcLink
```

## Output:

```
{
  "CreateDate": "2020-04-07T00:27:47Z",
  "Name": "MyUpdatedVpcLink",
  "SecurityGroupIds": [
    "sg1234",
    "sg5678"
  ],
  "SubnetIds": [
    "subnet-aaaa",
    "subnet-bbbb"
  ],
  "Tags": {},
  "VpcLinkId": "abcd123",
  "VpcLinkStatus": "AVAILABLE",
  "VpcLinkStatusMessage": "VPC link is ready to route traffic",
  "VpcLinkVersion": "V2"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tautan VPC untuk API HTTP di Panduan Pengembang Amazon API Gateway](#).

- Untuk detail API, lihat [UpdateVpcLink](#) di Referensi AWS CLI Perintah.

## Contoh API Gateway Management API menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan API Gateway Management API.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)



## Tindakan

### **delete-connection**

Contoh kode berikut menunjukkan cara menggunakan `delete-connection`.

#### AWS CLI

Untuk menghapus WebSocket koneksi

`delete-connection` Contoh berikut memutus klien dari WebSocket API yang ditentukan.

```
aws apigatewaymanagementapi delete-connection \  
  --connection-id L0SM9c0FvHcCIhw= \  
  --endpoint-url https://aabbccddee.execute-api.us-west-2.amazonaws.com/prod
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan perintah @connections di layanan backend Anda di Panduan](#) Pengembang Amazon API Gateway.

- Untuk detail API, lihat [DeleteConnection](#) di Referensi AWS CLI Perintah.

### **get-connection**

Contoh kode berikut menunjukkan cara menggunakan `get-connection`.

#### AWS CLI

Untuk mendapatkan informasi tentang WebSocket koneksi

`get-connection` Contoh berikut menjelaskan koneksi ke WebSocket API yang ditentukan.

```
aws apigatewaymanagementapi get-connection \  
  --connection-id L0SM9c0FvHcCIhw= \  
  --endpoint-url https://aabbccddee.execute-api.us-west-2.amazonaws.com/prod
```

Output:

```
{  
  "ConnectedAt": "2020-04-30T20:10:33.236Z",  
  "Identity": {
```

```
    "SourceIp": "192.0.2.1"  
  },  
  "LastActiveAt": "2020-04-30T20:10:42.997Z"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan perintah @connections di layanan backend Anda di Panduan](#) Pengembang Amazon API Gateway.

- Untuk detail API, lihat [GetConnection](#) di Referensi AWS CLI Perintah.

## post-to-connection

Contoh kode berikut menunjukkan cara menggunakan `post-to-connection`.

### AWS CLI

Untuk mengirim data ke WebSocket koneksi

`post-to-connection` Contoh berikut mengirimkan pesan ke klien yang terhubung ke WebSocket API yang ditentukan.

```
aws apigatewaymanagementapi post-to-connection \  
  --connection-id L0SM9c0FvHcCIhw= \  
  --data "Hello from API Gateway!" \  
  --endpoint-url https://aabbccdde.execute-api.us-west-2.amazonaws.com/prod
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan perintah @connections di layanan backend Anda di Panduan](#) Pengembang Amazon API Gateway.

- Untuk detail API, lihat [PostToConnection](#) di Referensi AWS CLI Perintah.

## Contoh App Mesh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan App Mesh AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-mesh**

Contoh kode berikut menunjukkan cara menggunakan `create-mesh`.

AWS CLI

Contoh 1: Untuk membuat mesh layanan baru

`create-mesh` Contoh berikut membuat mesh layanan.

```
aws appmesh create-mesh \  
  --mesh-name app1
```

Output:

```
{  
  "mesh": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",  
      "createdAt": 1563809909.282,  
      "lastUpdatedAt": 1563809909.282,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "spec": {},  
    "status": {  
      "status": "ACTIVE"  
    }  
  }  
}
```

Contoh 2: Untuk membuat mesh layanan baru dengan beberapa tag

`create-mesh` Contoh berikut membuat mesh layanan dengan beberapa tag.

```
aws appmesh create-mesh \
  --mesh-name app2 \
  --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Output:

```
{
  "mesh":{
    "meshName":"app2",
    "metadata":{
      "arn":"arn:aws:appmesh:us-east-1:123456789012:mesh/app2",
      "createdAt":1563822121.877,
      "lastUpdatedAt":1563822121.877,
      "uid":"a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version":1
    },
    "spec":{},
    "status":{
      "status":"ACTIVE"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Service Meshes](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [CreateMesh](#) di Referensi AWS CLI Perintah.

## **create-route**

Contoh kode berikut menunjukkan cara menggunakan `create-route`.

### AWS CLI

Untuk membuat rute gRPC baru

`create-route` Contoh berikut menggunakan file input JSON untuk membuat rute gRPC. Lalu lintas GRPC yang memiliki metadata yang dimulai dengan 123 dirutekan ke node virtual bernama ServiceBGRPC. Jika ada kegagalan gRPC, HTTP, atau TCP tertentu saat mencoba

berkomunikasi dengan target rute, rute dicoba ulang tiga kali. Ada penundaan 15 detik antara setiap upaya coba lagi.

```
aws appmesh create-route \  
  --cli-input-json file://create-route-grpc.json
```

Isi dari `create-route-grpc.json`:

```
{  
  "meshName" : "apps",  
  "routeName" : "grpcRoute",  
  "spec" : {  
    "grpcRoute" : {  
      "action" : {  
        "weightedTargets" : [  
          {  
            "virtualNode" : "serviceBgrpc",  
            "weight" : 100  
          }  
        ]  
      },  
      "match" : {  
        "metadata" : [  
          {  
            "invert" : false,  
            "match" : {  
              "prefix" : "123"  
            },  
            "name" : "myMetadata"  
          }  
        ],  
        "methodName" : "GetColor",  
        "serviceName" : "com.amazonaws.services.ColorService"  
      },  
      "retryPolicy" : {  
        "grpcRetryEvents" : [ "deadline-exceeded" ],  
        "httpRetryEvents" : [ "server-error", "gateway-error" ],  
        "maxRetries" : 3,  
        "perRetryTimeout" : {  
          "unit" : "s",  
          "value" : 15  
        },  
        "tcpRetryEvents" : [ "connection-error" ]  
      }  
    }  
  }  
}
```

```
    }
  },
  "priority" : 100
},
"virtualRouterName" : "serviceBgrpc"
}
```

### Output:

```
{
  "route": {
    "meshName": "apps",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBgrpc/route/grpcRoute",
      "createdAt": 1572010806.008,
      "lastUpdatedAt": 1572010806.008,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "grpcRoute",
    "spec": {
      "grpcRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "serviceBgrpc",
              "weight": 100
            }
          ]
        },
        "match": {
          "metadata": [
            {
              "invert": false,
              "match": {
                "prefix": "123"
              },
              "name": "mymetadata"
            }
          ],
          "methodName": "GetColor",
          "serviceName": "com.amazonaws.services.ColorService"
        }
      }
    }
  }
}
```

```
    },
    "retryPolicy": {
      "grpcRetryEvents": [
        "deadline-exceeded"
      ],
      "httpRetryEvents": [
        "server-error",
        "gateway-error"
      ],
      "maxRetries": 3,
      "perRetryTimeout": {
        "unit": "s",
        "value": 15
      },
      "tcpRetryEvents": [
        "connection-error"
      ]
    },
    "priority": 100
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualRouterName": "serviceBgrpc"
}
}
```

Untuk membuat rute HTTP atau HTTP/2 baru

`create-route` Contoh berikut menggunakan file input JSON untuk membuat rute HTTP/2. Untuk membuat rute HTTP, ganti `Http2Route` dengan `HttpRoute` di bawah spesifikasi. Semua lalu lintas HTTP/2 yang ditujukan ke awalan URL apa pun yang memiliki nilai header yang dimulai dengan 123 dirutekan ke node virtual bernama `ServiceBHTTP2`. Jika ada kegagalan HTTP atau TCP tertentu ketika mencoba berkomunikasi dengan target rute, rute dicoba ulang tiga kali. Ada penundaan 15 detik antara setiap upaya coba lagi.

```
aws appmesh create-route \  
  --cli-input-json file://create-route-http2.json
```

Isi dari `create-route-http2.json`:

```
{
  "meshName": "apps",
  "routeName": "http2Route",
  "spec": {
    "http2Route": {
      "action": {
        "weightedTargets": [
          {
            "virtualNode": "serviceBhttp2",
            "weight": 100
          }
        ]
      },
      "match": {
        "headers": [
          {
            "invert": false,
            "match": {
              "prefix": "123"
            },
            "name": "clientRequestId"
          }
        ],
        "method": "POST",
        "prefix": "/",
        "scheme": "http"
      },
      "retryPolicy": {
        "httpRetryEvents": [
          "server-error",
          "gateway-error"
        ],
        "maxRetries": 3,
        "perRetryTimeout": {
          "unit": "s",
          "value": 15
        },
        "tcpRetryEvents": [
          "connection-error"
        ]
      }
    },
    "priority": 200
  }
}
```



```
  },  
  "virtualRouterName": "serviceBhttp2"  
}
```

### Output:

```
{  
  "route": {  
    "meshName": "apps",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/  
serviceBhttp2/route/http2Route",  
      "createdAt": 1572011008.352,  
      "lastUpdatedAt": 1572011008.352,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "routeName": "http2Route",  
    "spec": {  
      "http2Route": {  
        "action": {  
          "weightedTargets": [  
            {  
              "virtualNode": "serviceBhttp2",  
              "weight": 100  
            }  
          ]  
        },  
        "match": {  
          "headers": [  
            {  
              "invert": false,  
              "match": {  
                "prefix": "123"  
              },  
              "name": "clientRequestId"  
            }  
          ],  
          "method": "POST",  
          "prefix": "/",  
          "scheme": "http"  
        },  
        "retryPolicy": {
```

```

        "httpRetryEvents": [
            "server-error",
            "gateway-error"
        ],
        "maxRetries": 3,
        "perRetryTimeout": {
            "unit": "s",
            "value": 15
        },
        "tcpRetryEvents": [
            "connection-error"
        ]
    }
},
"priority": 200
},
"status": {
    "status": "ACTIVE"
},
"virtualRouterName": "serviceBhttp2"
}
}

```

Untuk membuat rute TCP baru

`create-route` Contoh berikut menggunakan file input JSON untuk membuat rute TCP. 75 persen lalu lintas dirutekan ke node virtual bernama `ServiceBTCP`, dan 25 persen lalu lintas dirutekan ke node virtual bernama `ServiceBV2TCP`. Menentukan bobot yang berbeda untuk target yang berbeda adalah cara yang efektif untuk melakukan penerapan versi baru aplikasi. Anda dapat menyesuaikan bobot sehingga pada akhirnya, 100 persen dari semua lalu lintas diarahkan ke target yang memiliki versi baru aplikasi.

```

aws appmesh create-route \
  --cli-input-json file://create-route-tcp.json

```

Isi dari `create-route-tcp.json`:

```

{
  "meshName": "apps",
  "routeName": "tcpRoute",
  "spec": {
    "priority": 300,

```

```

    "tcpRoute": {
      "action": {
        "weightedTargets": [
          {
            "virtualNode": "serviceBtcp",
            "weight": 75
          },
          {
            "virtualNode": "serviceBv2tcp",
            "weight": 25
          }
        ]
      }
    },
    "virtualRouterName": "serviceBtcp"
  }
}

```

### Output:

```

{
  "route": {
    "meshName": "apps",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/serviceBtcp/route/tcpRoute",
      "createdAt": 1572011436.26,
      "lastUpdatedAt": 1572011436.26,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "tcpRoute",
    "spec": {
      "priority": 300,
      "tcpRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "serviceBtcp",
              "weight": 75
            },
            {
              "virtualNode": "serviceBv2tcp",

```

```

        "weight": 25
      }
    ]
  }
},
"status": {
  "status": "ACTIVE"
},
"virtualRouterName": "serviceBtcp"
}
}

```

Untuk informasi selengkapnya, lihat [Rute](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [CreateRoute](#) di Referensi AWS CLI Perintah.

## create-virtual-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-virtual-gateway`.

### AWS CLI

Untuk membuat gateway virtual baru

`create-virtual-gateway` Contoh berikut menggunakan file input JSON untuk membuat gateway virtual dengan pendengar untuk HTTP menggunakan port 9080.

```

aws appmesh create-virtual-gateway \
  --mesh-name meshName \
  --virtual-gateway-name virtualGatewayName \
  --cli-input-json file://create-virtual-gateway.json

```

Isi dari `create-virtual-gateway.json`:

```

{
  "spec": {
    "listeners": [
      {
        "portMapping": {
          "port": 9080,
          "protocol": "http"
        }
      }
    ]
  }
}

```

```

    }
  }
]
}
}

```

### Output:

```

{
  "virtualGateway": {
    "meshName": "meshName",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/meshName/
virtualGateway/virtualGatewayName",
      "createdAt": "2022-04-06T10:42:42.015000-05:00",
      "lastUpdatedAt": "2022-04-06T10:42:42.015000-05:00",
      "meshOwner": "123456789012",
      "resourceOwner": "123456789012",
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "spec": {
      "listeners": [
        {
          "portMapping": {
            "port": 9080,
            "protocol": "http"
          }
        }
      ]
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualGatewayName": "virtualGatewayName"
  }
}

```

Untuk informasi selengkapnya, lihat [Gateway Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [CreateVirtualGateway](#) di Referensi AWS CLI Perintah.

## create-virtual-node

Contoh kode berikut menunjukkan cara menggunakan `create-virtual-node`.

### AWS CLI

Contoh 1: Untuk membuat node virtual baru yang menggunakan DNS untuk penemuan

`create-virtual-node` Contoh berikut menggunakan file input JSON untuk membuat node virtual yang menggunakan DNS untuk penemuan layanan.

```
aws appmesh create-virtual-node \  
  --cli-input-json file://create-virtual-node-dns.json
```

Isi dari `create-virtual-node-dns.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "portMapping": {  
          "port": 80,  
          "protocol": "http"  
        }  
      }  
    ],  
    "serviceDiscovery": {  
      "dns": {  
        "hostname": "serviceBv1.svc.cluster.local"  
      }  
    }  
  },  
  "virtualNodeName": "vnServiceBv1"  
}
```

Output:

```
{  
  "virtualNode": {  
    "meshName": "app1",  
    "metadata": {
```

```

    "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv1",
    "createdAt": 1563810019.874,
    "lastUpdatedAt": 1563810019.874,
    "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "version": 1
  },
  "spec": {
    "listeners": [
      {
        "portMapping": {
          "port": 80,
          "protocol": "http"
        }
      }
    ],
    "serviceDiscovery": {
      "dns": {
        "hostname": "serviceBv1.svc.cluster.local"
      }
    }
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualNodeName": "vnServiceBv1"
}
}

```

Contoh 2: Untuk membuat node virtual baru yang menggunakan AWS Cloud Map untuk penemuan

`create-virtual-node` Contoh berikut menggunakan file input JSON untuk membuat node virtual yang menggunakan AWS Cloud Map untuk penemuan layanan.

```

aws appmesh create-virtual-node \
  --cli-input-json file://create-virtual-node-cloud-map.json

```

Isi dari `create-virtual-node-cloud-map.json`:

```

{
  "meshName": "app1",

```

```
"spec": {
  "backends": [
    {
      "virtualService": {
        "virtualServiceName": "serviceA.svc.cluster.local"
      }
    }
  ],
  "listeners": [
    {
      "portMapping": {
        "port": 80,
        "protocol": "http"
      }
    }
  ],
  "serviceDiscovery": {
    "awsCloudMap": {
      "attributes": [
        {
          "key": "Environment",
          "value": "Testing"
        }
      ],
      "namespaceName": "namespace1",
      "serviceName": "serviceA"
    }
  ],
  "virtualNodeName": "vnServiceA"
}
```

### Output:

```
{
  "virtualNode": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceA",
      "createdAt": 1563810859.465,
      "lastUpdatedAt": 1563810859.465,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    }
  }
}
```



```
    "version": 1
  },
  "spec": {
    "backends": [
      {
        "virtualService": {
          "virtualServiceName": "serviceA.svc.cluster.local"
        }
      }
    ],
    "listeners": [
      {
        "portMapping": {
          "port": 80,
          "protocol": "http"
        }
      }
    ],
    "serviceDiscovery": {
      "awsCloudMap": {
        "attributes": [
          {
            "key": "Environment",
            "value": "Testing"
          }
        ],
        "namespaceName": "namespace1",
        "serviceName": "serviceA"
      }
    }
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualNodeName": "vnServiceA"
}
```

Untuk informasi selengkapnya, lihat [Node Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [CreateVirtualNode](#) di Referensi AWS CLI Perintah.

## create-virtual-router

Contoh kode berikut menunjukkan cara menggunakan `create-virtual-router`.

### AWS CLI

Untuk membuat router virtual baru

`create-virtual-router` Contoh berikut menggunakan file input JSON untuk membuat router virtual dengan pendengar untuk HTTP menggunakan port 80.

```
aws appmesh create-virtual-router \  
  --cli-input-json file://create-virtual-router.json
```

Isi dari `create-virtual-router.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "portMapping": {  
          "port": 80,  
          "protocol": "http"  
        }  
      }  
    ]  
  },  
  "virtualRouterName": "vrServiceB"  
}
```

Output:

```
{  
  "virtualRouter": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB",  
      "createdAt": 1563810546.59,  
      "lastUpdatedAt": 1563810546.59,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    }  
  }  
}
```

```

    "version": 1
  },
  "spec": {
    "listeners": [
      {
        "portMapping": {
          "port": 80,
          "protocol": "http"
        }
      }
    ]
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualRouterName": "vrServiceB"
}
}

```

Untuk informasi selengkapnya, lihat [Router Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [CreateVirtualRouter](#) di Referensi AWS CLI Perintah.

## create-virtual-service

Contoh kode berikut menunjukkan cara menggunakan `create-virtual-service`.

### AWS CLI

Contoh 1: Untuk membuat layanan virtual baru dengan penyedia node virtual

`create-virtual-service` Contoh berikut menggunakan file input JSON untuk membuat layanan virtual dengan penyedia node virtual.

```

aws appmesh create-virtual-service \
  --cli-input-json file://create-virtual-service-virtual-node.json

```

Isi dari `create-virtual-service-virtual-node.json`:

```

{
  "meshName": "app1",
  "spec": {
    "provider": {

```

```

        "virtualNode": {
            "virtualNodeName": "vnServiceA"
        }
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
}

```

### Output:

```

{
  "virtualService": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/serviceA.svc.cluster.local",
      "createdAt": 1563810859.474,
      "lastUpdatedAt": 1563810967.179,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    },
    "spec": {
      "provider": {
        "virtualNode": {
          "virtualNodeName": "vnServiceA"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
  }
}

```

Untuk informasi selengkapnya, lihat [Node Virtual](#) di Panduan Pengguna AWS App Mesh.

Contoh 2: Untuk membuat layanan virtual baru dengan penyedia router virtual

`create-virtual-service` Contoh berikut menggunakan file input JSON untuk membuat layanan virtual dengan penyedia router virtual.

```
aws appmesh create-virtual-service \
```

```
--cli-input-json file://create-virtual-service-virtual-router.json
```

Isi dari create-virtual-service-virtual-router.json:

```
{
  "meshName": "app1",
  "spec": {
    "provider": {
      "virtualRouter": {
        "virtualRouterName": "vrServiceB"
      }
    }
  },
  "virtualServiceName": "serviceB.svc.cluster.local"
}
```

Output:

```
{
  "virtualService": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
      "createdAt": 1563908363.999,
      "lastUpdatedAt": 1563908363.999,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "spec": {
      "provider": {
        "virtualRouter": {
          "virtualRouterName": "vrServiceB"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceB.svc.cluster.local"
  }
}
```

Untuk informasi selengkapnya, lihat Layanan Virtual < [https://docs.aws.amazon.com/app-mesh/latest/userguide/virtual\\_services.html](https://docs.aws.amazon.com/app-mesh/latest/userguide/virtual_services.html) > di Panduan Pengguna AWS App Mesh

- Untuk detail API, lihat [CreateVirtualService](#) di Referensi AWS CLI Perintah.

## delete-mesh

Contoh kode berikut menunjukkan cara menggunakan `delete-mesh`.

### AWS CLI

Untuk menghapus mesh layanan

`delete-mesh` Contoh berikut menghapus mesh layanan yang ditentukan.

```
aws appmesh delete-mesh \  
  --mesh-name app1
```

Output:

```
{  
  "mesh": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",  
      "createdAt": 1563809909.282,  
      "lastUpdatedAt": 1563824981.248,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 2  
    },  
    "spec": {  
      "egressFilter": {  
        "type": "ALLOW_ALL"  
      }  
    },  
    "status": {  
      "status": "DELETED"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Service Meshes](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DeleteMesh](#) di Referensi AWS CLI Perintah.

## delete-route

Contoh kode berikut menunjukkan cara menggunakan `delete-route`.

### AWS CLI

Untuk menghapus rute

`delete-route` Contoh berikut menghapus rute yang ditentukan.

```
aws appmesh delete-route \  
  --mesh-name app1 \  
  --virtual-router-name vrServiceB \  
  --route-name toVnServiceB-weighted
```

Output:

```
{  
  "route": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB/route/toVnServiceB-weighted",  
      "createdAt": 1563811384.015,  
      "lastUpdatedAt": 1563823915.936,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 3  
    },  
    "routeName": "toVnServiceB-weighted",  
    "spec": {  
      "httpRoute": {  
        "action": {  
          "weightedTargets": [  
            {  
              "virtualNode": "vnServiceBv1",  
              "weight": 80  
            },  
            {  
              "virtualNode": "vnServiceBv2",  
              "weight": 20  
            }  
          ]  
        }  
      }  
    }  
  }  
}
```

```

        }
      ],
      },
      "match": {
        "prefix": "/"
      }
    }
  },
  "status": {
    "status": "DELETED"
  },
  "virtualRouterName": "vrServiceB"
}
}

```

Untuk informasi selengkapnya, lihat [Route](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DeleteRoute](#) di Referensi AWS CLI Perintah.

## delete-virtual-node

Contoh kode berikut menunjukkan cara menggunakan `delete-virtual-node`.

### AWS CLI

Untuk menghapus node virtual

`delete-virtual-node` Contoh berikut menghapus node virtual yang ditentukan.

```

aws appmesh delete-virtual-node \
  --mesh-name app1 \
  --virtual-node-name vnServiceBv2

```

Output:

```

{
  "virtualNode": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceBv2",
      "createdAt": 1563810117.297,
      "lastUpdatedAt": 1563824700.678,
    }
  }
}

```



```

        "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "version": 2
    },
    "spec": {
        "backends": [],
        "listeners": [
            {
                "portMapping": {
                    "port": 80,
                    "protocol": "http"
                }
            }
        ],
        "serviceDiscovery": {
            "dns": {
                "hostname": "serviceBv2.svc.cluster.local"
            }
        }
    },
    "status": {
        "status": "DELETED"
    },
    "virtualNodeName": "vnServiceBv2"
}
}

```

Untuk informasi selengkapnya, lihat [Node Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DeleteVirtualNode](#) di Referensi AWS CLI Perintah.

## delete-virtual-router

Contoh kode berikut menunjukkan cara menggunakan `delete-virtual-router`.

### AWS CLI

Untuk menghapus router virtual

`delete-virtual-router` Contoh berikut menghapus router virtual yang ditentukan.

```

aws appmesh delete-virtual-router \
  --mesh-name app1 \
  --virtual-router-name vrServiceB

```

**Output:**

```
{
  "virtualRouter": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/vrServiceB",
      "createdAt": 1563810546.59,
      "lastUpdatedAt": 1563824253.467,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 3
    },
    "spec": {
      "listeners": [
        {
          "portMapping": {
            "port": 80,
            "protocol": "http"
          }
        }
      ]
    },
    "status": {
      "status": "DELETED"
    },
    "virtualRouterName": "vrServiceB"
  }
}
```

Untuk informasi selengkapnya, lihat [Router Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DeleteVirtualRouter](#) di Referensi AWS CLI Perintah.

**delete-virtual-service**

Contoh kode berikut menunjukkan cara menggunakan `delete-virtual-service`.

**AWS CLI**

Untuk menghapus layanan virtual

`delete-virtual-service` Contoh berikut menghapus layanan virtual yang ditentukan.

```
aws appmesh delete-virtual-service \  
  --mesh-name app1 \  
  --virtual-service-name serviceB.svc.cluster.local
```

Output:

```
{  
  "virtualService": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/  
serviceB.svc.cluster.local",  
      "createdAt": 1563908363.999,  
      "lastUpdatedAt": 1563913940.866,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 3  
    },  
    "spec": {},  
    "status": {  
      "status": "DELETED"  
    },  
    "virtualServiceName": "serviceB.svc.cluster.local"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Layanan Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DeleteVirtualService](#) di Referensi AWS CLI Perintah.

## describe-mesh

Contoh kode berikut menunjukkan cara menggunakan `describe-mesh`.

### AWS CLI

Untuk menggambarkan mesh layanan

`describe-mesh` Contoh berikut mengembalikan rincian tentang mesh layanan tertentu.

```
aws appmesh describe-mesh \  
  --mesh-name app1
```

**Output:**

```
{
  "mesh": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
      "createdAt": 1563809909.282,
      "lastUpdatedAt": 1563809909.282,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "spec": {},
    "status": {
      "status": "ACTIVE"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Service Meshes](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DescribeMesh](#) di Referensi AWS CLI Perintah.

**describe-route**

Contoh kode berikut menunjukkan cara menggunakan `describe-route`.

**AWS CLI**

Untuk menggambarkan rute

`describe-route` Contoh berikut mengembalikan rincian tentang rute yang ditentukan.

```
aws appmesh describe-route \
  --mesh-name app1 \
  --virtual-router-name vrServiceB \
  --route-name toVnServiceB-weighted
```

**Output:**

```
{
  "route": {
```

```

    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB/route/toVnServiceB-weighted",
      "createdAt": 1563811384.015,
      "lastUpdatedAt": 1563811384.015,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "toVnServiceB-weighted",
    "spec": {
      "httpRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "vnServiceBv1",
              "weight": 90
            },
            {
              "virtualNode": "vnServiceBv2",
              "weight": 10
            }
          ]
        },
        "match": {
          "prefix": "/"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualRouterName": "vrServiceB"
  }
}

```

Untuk informasi selengkapnya, lihat [Route](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DescribeRoute](#) di Referensi AWS CLI Perintah.

## describe-virtual-node

Contoh kode berikut menunjukkan cara menggunakan `describe-virtual-node`.

## AWS CLI

Untuk menggambarkan node virtual

`describe-virtual-node` Contoh berikut mengembalikan rincian tentang node virtual tertentu.

```
aws appmesh describe-virtual-node \  
  --mesh-name app1 \  
  --virtual-node-name vnServiceBv1
```

Output:

```
{  
  "virtualNode": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/  
vnServiceBv1",  
      "createdAt": 1563810019.874,  
      "lastUpdatedAt": 1563810019.874,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "spec": {  
      "backends": [],  
      "listeners": [  
        {  
          "portMapping": {  
            "port": 80,  
            "protocol": "http"  
          }  
        }  
      ],  
      "serviceDiscovery": {  
        "dns": {  
          "hostname": "serviceBv1.svc.cluster.local"  
        }  
      }  
    },  
    "status": {  
      "status": "ACTIVE"  
    },  
    "virtualNodeName": "vnServiceBv1"  
  }  
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Node Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DescribeVirtualNode](#) di Referensi AWS CLI Perintah.

## describe-virtual-router

Contoh kode berikut menunjukkan cara menggunakan `describe-virtual-router`.

### AWS CLI

Untuk menggambarkan router virtual

`describe-virtual-router` Contoh berikut mengembalikan rincian tentang router virtual yang ditentukan.

```
aws appmesh describe-virtual-router \  
  --mesh-name app1 \  
  --virtual-router-name vrServiceB
```

### Output:

```
{  
  "virtualRouter": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB",  
      "createdAt": 1563810546.59,  
      "lastUpdatedAt": 1563810546.59,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "spec": {  
      "listeners": [  
        {  
          "portMapping": {  
            "port": 80,  
            "protocol": "http"  
          }  
        }  
      ]  
    }  
  }  
}
```

```

        }
      ]
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualRouterName": "vrServiceB"
  }
}

```

Untuk informasi selengkapnya, lihat [Router Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DescribeVirtualRouter](#) di Referensi AWS CLI Perintah.

## describe-virtual-service

Contoh kode berikut menunjukkan cara menggunakan `describe-virtual-service`.

### AWS CLI

Untuk menggambarkan layanan virtual

`describe-virtual-service` Contoh berikut mengembalikan rincian tentang layanan virtual yang ditentukan.

```

aws appmesh describe-virtual-service \
  --mesh-name app1 \
  --virtual-service-name serviceB.svc.cluster.local

```

Output:

```

{
  "virtualService": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/serviceB.svc.cluster.local",
      "createdAt": 1563908363.999,
      "lastUpdatedAt": 1563908363.999,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    }
  },

```



```
    "spec": {
      "provider": {
        "virtualRouter": {
          "virtualRouterName": "vrServiceB"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceB.svc.cluster.local"
  }
}
```

Untuk informasi selengkapnya, lihat [Layanan Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [DescribeVirtualService](#) di Referensi AWS CLI Perintah.

## list-meshes

Contoh kode berikut menunjukkan cara menggunakan `list-meshes`.

### AWS CLI

Untuk membuat daftar jerat layanan

`list-meshes` Contoh berikut mencantumkan semua jerat layanan di AWS Wilayah saat ini.

```
aws appmesh list-meshes
```

Output:

```
{
  "meshes": [
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
      "meshName": "app1"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Service Meshes](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [ListMeshes](#) di Referensi AWS CLI Perintah.

## list-routes

Contoh kode berikut menunjukkan cara menggunakan `list-routes`.

### AWS CLI

Untuk daftar rute

`list-routes` Contoh berikut mencantumkan semua rute untuk router virtual yang ditentukan.

```
aws appmesh list-routes \  
  --mesh-name app1 \  
  --virtual-router-name vrServiceB
```

Output:

```
{  
  "routes": [  
    {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB/route/toVnServiceB",  
      "meshName": "app1",  
      "routeName": "toVnServiceB-weighted",  
      "virtualRouterName": "vrServiceB"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Rute](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [ListRoutes](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan semua tag yang ditetapkan ke sumber daya yang ditentukan.

```
aws appmesh list-tags-for-resource \  
  --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1
```

Output:

```
{  
  "tags": [  
    {  
      "key": "key1",  
      "value": "value1"  
    },  
    {  
      "key": "key2",  
      "value": "value2"  
    },  
    {  
      "key": "key3",  
      "value": "value3"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## `list-virtual-nodes`

Contoh kode berikut menunjukkan cara menggunakan `list-virtual-nodes`.

### AWS CLI

Untuk daftar node virtual

`list-virtual-nodes` Contoh berikut mencantumkan semua node virtual di mesh layanan yang ditentukan.

```
aws appmesh list-virtual-nodes \  
  --mesh-name app1
```

**Output:**

```
{
  "virtualNodes": [
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceBv1",
      "meshName": "app1",
      "virtualNodeName": "vnServiceBv1"
    },
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceBv2",
      "meshName": "app1",
      "virtualNodeName": "vnServiceBv2"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Node Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [ListVirtualNodes](#) di Referensi AWS CLI Perintah.

**list-virtual-routers**

Contoh kode berikut menunjukkan cara menggunakan `list-virtual-routers`.

**AWS CLI**

Untuk daftar router virtual

`list-virtual-routers` Contoh berikut mencantumkan semua router virtual di mesh layanan yang ditentukan.

```
aws appmesh list-virtual-routers \
  --mesh-name app1
```

**Output:**

```
{
  "virtualRouters": [
    {
```

```

        "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
        "meshName": "app1",
        "virtualRouterName": "vrServiceB"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Router Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [ListVirtualRouters](#) di Referensi AWS CLI Perintah.

## list-virtual-services

Contoh kode berikut menunjukkan cara menggunakan `list-virtual-services`.

### AWS CLI

Untuk daftar layanan virtual

`list-virtual-services` Contoh berikut mencantumkan semua layanan virtual di mesh layanan yang ditentukan.

```

aws appmesh list-virtual-services \
  --mesh-name app1

```

Output:

```

{
  "virtualServices": [
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
      "meshName": "app1",
      "virtualServiceName": "serviceA.svc.cluster.local"
    },
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
      "meshName": "app1",
      "virtualServiceName": "serviceB.svc.cluster.local"
    }
  ]
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Layanan Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [ListVirtualServices](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan tag `key1` dengan nilai `value1` ke sumber daya yang ditentukan.

```
aws appmesh tag-resource \
  --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1 \
  --tags key=key1,value=value1
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag sumber daya

`untag-resource` Contoh berikut menghapus tag dengan kunci `key1` dari sumber daya yang ditentukan.

```
aws appmesh untag-resource \
  --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1 \
  --tag-keys key1
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-mesh

Contoh kode berikut menunjukkan cara menggunakan `update-mesh`.

### AWS CLI

Untuk memperbarui mesh layanan

`update-mesh` Contoh berikut menggunakan file input JSON untuk memperbarui mesh layanan untuk memungkinkan semua lalu lintas keluar eksternal diteruskan melalui proxy Envoy yang tidak tersentuh.

```
aws appmesh update-mesh \  
  --cli-input-json file://update-mesh.json
```

Isi dari `update-mesh.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "egressFilter": {  
      "type": "ALLOW_ALL"  
    }  
  }  
}
```

Output:

```
{  
  "mesh": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",  
      "createdAt": 1563809909.282,  
      "lastUpdatedAt": 1563812829.687,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 2  
    },  
    "spec": {
```

```
        "egressFilter": {
            "type": "ALLOW_ALL"
        },
        "status": {
            "status": "ACTIVE"
        }
    }
}
```

Untuk informasi selengkapnya, lihat [Service Meshes](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [UpdateMesh](#) di Referensi AWS CLI Perintah.

## update-route

Contoh kode berikut menunjukkan cara menggunakan `update-route`.

### AWS CLI

Untuk memperbarui rute

`update-route` Contoh berikut menggunakan file input JSON untuk memperbarui bobot untuk rute.

```
aws appmesh update-route \  
  --cli-input-json file://update-route-weighted.json
```

Isi dari `update-route-weighted.json`:

```
{  
  "meshName": "app1",  
  "routeName": "toVnServiceB-weighted",  
  "spec": {  
    "httpRoute": {  
      "action": {  
        "weightedTargets": [  
          {  
            "virtualNode": "vnServiceBv1",  
            "weight": 80  
          },  
          {  

```



```

        "virtualNode": "vnServiceBv2",
        "weight": 20
      }
    ]
  },
  "match": {
    "prefix": "/"
  }
}
},
"virtualRouterName": "vrServiceB"
}

```

### Output:

```

{
  "route": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/vrServiceB/route/toVnServiceB-weighted",
      "createdAt": 1563811384.015,
      "lastUpdatedAt": 1563819600.022,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    },
    "routeName": "toVnServiceB-weighted",
    "spec": {
      "httpRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "vnServiceBv1",
              "weight": 80
            },
            {
              "virtualNode": "vnServiceBv2",
              "weight": 20
            }
          ]
        },
        "match": {
          "prefix": "/"
        }
      }
    }
  }
}

```

```
    }
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualRouterName": "vrServiceB"
}
}
```

Untuk informasi selengkapnya, lihat [Route](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [UpdateRoute](#) di Referensi AWS CLI Perintah.

## update-virtual-node

Contoh kode berikut menunjukkan cara menggunakan `update-virtual-node`.

### AWS CLI

Untuk memperbarui node virtual

`update-virtual-node` Contoh berikut menggunakan file input JSON untuk menambahkan pemeriksaan kesehatan ke node virtual.

```
aws appmesh update-virtual-node \  
  --cli-input-json file://update-virtual-node.json
```

Isi dari `update-virtual-node.json`:

```
{  
  "clientToken": "500",  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "healthCheck": {  
          "healthyThreshold": 5,  
          "intervalMillis": 10000,  
          "path": "/",  
          "port": 80,  
          "protocol": "http",
```

```

        "timeoutMillis": 3000,
        "unhealthyThreshold": 3
    },
    "portMapping": {
        "port": 80,
        "protocol": "http"
    }
},
"serviceDiscovery": {
    "dns": {
        "hostname": "serviceBv1.svc.cluster.local"
    }
}
},
"virtualNodeName": "vnServiceBv1"
}

```

### Output:

```

{
  "virtualNode": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceBv1",
      "createdAt": 1563810019.874,
      "lastUpdatedAt": 1563819234.825,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    },
    "spec": {
      "listeners": [
        {
          "healthCheck": {
            "healthyThreshold": 5,
            "intervalMillis": 10000,
            "path": "/",
            "port": 80,
            "protocol": "http",
            "timeoutMillis": 3000,
            "unhealthyThreshold": 3
          }
        }
      ]
    }
  }
}

```

```

        "portMapping": {
            "port": 80,
            "protocol": "http"
        }
    ],
    "serviceDiscovery": {
        "dns": {
            "hostname": "serviceBv1.svc.cluster.local"
        }
    },
    "status": {
        "status": "ACTIVE"
    },
    "virtualNodeName": "vnServiceBv1"
}
}

```

Untuk informasi selengkapnya, lihat [Node Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [UpdateVirtualNode](#) di Referensi AWS CLI Perintah.

## update-virtual-router

Contoh kode berikut menunjukkan cara menggunakan `update-virtual-router`.

### AWS CLI

Untuk memperbarui router virtual

`update-virtual-router` Contoh berikut menggunakan file input JSON untuk memperbarui port pendengar router virtual.

```
aws appmesh update-virtual-router \
  --cli-input-json file://update-virtual-router.json
```

Isi dari `update-virtual-router.json`:

```
{
  "meshName": "app1",
  "spec": {
```

```
    "listeners": [  
      {  
        "portMapping": {  
          "port": 8080,  
          "protocol": "http"  
        }  
      }  
    ],  
    "virtualRouterName": "vrServiceB"  
  }  
}
```

### Output:

```
{  
  "virtualRouter": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB",  
      "createdAt": 1563810546.59,  
      "lastUpdatedAt": 1563819431.352,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 2  
    },  
    "spec": {  
      "listeners": [  
        {  
          "portMapping": {  
            "port": 8080,  
            "protocol": "http"  
          }  
        }  
      ]  
    },  
    "status": {  
      "status": "ACTIVE"  
    },  
    "virtualRouterName": "vrServiceB"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Router Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [UpdateVirtualRouter](#) di Referensi AWS CLI Perintah.

## update-virtual-service

Contoh kode berikut menunjukkan cara menggunakan `update-virtual-service`.

### AWS CLI

Untuk memperbarui layanan virtual

`update-virtual-service` Contoh berikut menggunakan file input JSON untuk memperbarui layanan virtual untuk menggunakan penyedia router virtual.

```
aws appmesh update-virtual-service \  
  --cli-input-json file://update-virtual-service.json
```

Isi dari `update-virtual-service.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "provider": {  
      "virtualRouter": {  
        "virtualRouterName": "vrServiceA"  
      }  
    }  
  },  
  "virtualServiceName": "serviceA.svc.cluster.local"  
}
```

Output:

```
{  
  "virtualService": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/  
serviceA.svc.cluster.local",  
      "createdAt": 1563810859.474,  
      "lastUpdatedAt": 1563820257.411,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 3  
    }  
  }  
}
```

```
    },
    "spec": {
      "provider": {
        "virtualRouter": {
          "virtualRouterName": "vrServiceA"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
  }
}
```

Untuk informasi selengkapnya, lihat [Layanan Virtual](#) di Panduan Pengguna AWS App Mesh.

- Untuk detail API, lihat [UpdateVirtualService](#) di Referensi AWS CLI Perintah.

## Contoh App Runner menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum menggunakan AWS Command Line Interface With App Runner.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **associate-custom-domain**

Contoh kode berikut menunjukkan cara menggunakan `associate-custom-domain`.

## AWS CLI

Untuk mengaitkan nama domain dan subdomain `www` dengan layanan

`associate-custom-domain` Contoh berikut mengaitkan nama domain kustom yang Anda kontrol dengan layanan App Runner. Nama domain adalah domain `rootexample.com`, termasuk subdomain kasus khusus `www.example.com`

```
aws apprunner associate-custom-domain \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "DomainName": "example.com",  
  "EnableWWWSubdomain": true  
}
```

Output:

```
{  
  "CustomDomain": {  
    "CertificateValidationRecords": [  
      {  
        "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",  
        "Status": "PENDING_VALIDATION",  
        "Type": "CNAME",  
        "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-  
validations.aws."  
      },  
      {  
        "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",  
        "Status": "PENDING_VALIDATION",  
        "Type": "CNAME",  
        "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-  
validations.aws."  
      }  
    ],  
    "DomainName": "example.com",  
    "EnableWWWSubdomain": true,  
  }  
}
```



```

    "Status": "CREATING"
  },
  "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}

```

- Untuk detail API, lihat [AssociateCustomDomain](#) di Referensi AWS CLI Perintah.

## create-auto-scaling-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-auto-scaling-configuration`.

### AWS CLI

Untuk membuat konfigurasi penskalaan otomatis ketersediaan tinggi

`create-auto-scaling-configuration` Contoh berikut membuat konfigurasi penskalaan otomatis yang dioptimalkan untuk ketersediaan tinggi dengan menyetel `MinSize` ke 5. Dengan konfigurasi ini, App Runner mencoba menyebarkan instans layanan Anda ke sebagian besar Availability Zone, hingga lima, tergantung pada AWS Wilayah.

Panggilan mengembalikan `AutoScalingConfiguration` objek dengan pengaturan lain yang disetel ke defaultnya. Dalam contoh, ini adalah panggilan pertama untuk membuat konfigurasi bernama `high-availability`. Revisi diatur ke 1, dan ini adalah revisi terbaru.

```

aws apprunner create-auto-scaling-configuration \
  --cli-input-json file://input.json

```

Isi dari `input.json`:

```

{
  "AutoScalingConfigurationName": "high-availability",
  "MinSize": 5
}

```

Output:

```

{
  "AutoScalingConfiguration": {

```

```
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
    "AutoScalingConfigurationName": "high-availability",
    "AutoScalingConfigurationRevision": 1,
    "CreatedAt": "2020-11-03T00:29:17Z",
    "Latest": true,
    "Status": "ACTIVE",
    "MaxConcurrency": 100,
    "MaxSize": 50,
    "MinSize": 5
  }
}
```

- Untuk detail API, lihat [CreateAutoScalingConfiguration](#) di Referensi AWS CLI Perintah.

## create-connection

Contoh kode berikut menunjukkan cara menggunakan `create-connection`.

### AWS CLI

Untuk membuat GitHub koneksi

`create-connection` Contoh berikut membuat koneksi ke repositori GitHub kode pribadi. Status koneksi setelah panggilan berhasil adalah `PENDING_HANDSHAKE`. Ini karena jabat tangan otentikasi dengan penyedia masih belum terjadi. Selesaikan jabat tangan menggunakan konsol App Runner.

```
aws apprunner create-connection \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ConnectionName": "my-github-connection",
  "ProviderType": "GITHUB"
}
```

Output:

```
{
  "Connection": {
    "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-connection",
    "ConnectionName": "my-github-connection",
    "Status": "PENDING_HANDSHAKE",
    "CreatedAt": "2020-11-03T00:32:51Z",
    "ProviderType": "GITHUB"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola koneksi App Runner](#) di Panduan Pengembang AWS App Runner.

- Untuk detail API, lihat [CreateConnection](#) di Referensi AWS CLI Perintah.

## create-service

Contoh kode berikut menunjukkan cara menggunakan `create-service`.

### AWS CLI

Contoh 1: Untuk membuat layanan repositori kode sumber

`create-service` Contoh berikut membuat layanan App Runner berdasarkan repositori kode sumber Python.

```
aws apprunner create-service \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ServiceName": "python-app",
  "SourceConfiguration": {
    "AuthenticationConfiguration": {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-connection/e7656250f67242d7819feade6800f59e"
    },
    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "RepositoryUrl": "https://github.com/my-account/python-hello",

```

```

    "SourceCodeVersion": {
      "Type": "BRANCH",
      "Value": "main"
    },
    "CodeConfiguration": {
      "ConfigurationSource": "API",
      "CodeConfigurationValues": {
        "Runtime": "PYTHON_3",
        "BuildCommand": "pip install -r requirements.txt",
        "StartCommand": "python server.py",
        "Port": "8080",
        "RuntimeEnvironmentVariables": [
          {
            "NAME": "Jane"
          }
        ]
      }
    }
  },
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}

```

### Output:

```

{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-20T19:05:25Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      }
    }
  }
}

```

```

    },
    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "CodeConfiguration": {
        "CodeConfigurationValues": {
          "BuildCommand": "pip install -r requirements.txt",
          "Port": "8080",
          "Runtime": "PYTHON_3",
          "RuntimeEnvironmentVariables": [
            {
              "NAME": "Jane"
            }
          ],
          "StartCommand": "python server.py"
        },
        "ConfigurationSource": "Api"
      },
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      }
    }
  },
  "Status": "OPERATION_IN_PROGRESS",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}
}

```

Contoh 2: Untuk membuat layanan repositori kode sumber

`create-service` Contoh berikut membuat layanan App Runner berdasarkan repositori kode sumber Python.

```
aws apprunner create-service \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
```

```

"ServiceName": "python-app",
"SourceConfiguration": {
  "AuthenticationConfiguration": {
    "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-connection/e7656250f67242d7819feade6800f59e"
  },
  "AutoDeploymentsEnabled": true,
  "CodeRepository": {
    "RepositoryUrl": "https://github.com/my-account/python-hello",
    "SourceCodeVersion": {
      "Type": "BRANCH",
      "Value": "main"
    },
  },
  "CodeConfiguration": {
    "ConfigurationSource": "API",
    "CodeConfigurationValues": {
      "Runtime": "PYTHON_3",
      "BuildCommand": "pip install -r requirements.txt",
      "StartCommand": "python server.py",
      "Port": "8080",
      "RuntimeEnvironmentVariables": [
        {
          "NAME": "Jane"
        }
      ]
    }
  }
},
"InstanceConfiguration": {
  "CPU": "1 vCPU",
  "Memory": "3 GB"
}
}

```

### Output:

```

{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-20T19:05:25Z",
  }
}

```

```
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      },
      "AutoDeploymentsEnabled": true,
      "CodeRepository": {
        "CodeConfiguration": {
          "CodeConfigurationValues": {
            "BuildCommand": "pip install -r requirements.txt",
            "Port": "8080",
            "Runtime": "PYTHON_3",
            "RuntimeEnvironmentVariables": [
              {
                "NAME": "Jane"
              }
            ],
            "StartCommand": "python server.py"
          },
          "ConfigurationSource": "Api"
        },
        "RepositoryUrl": "https://github.com/my-account/python-hello",
        "SourceCodeVersion": {
          "Type": "BRANCH",
          "Value": "main"
        }
      }
    },
    "Status": "OPERATION_IN_PROGRESS",
    "InstanceConfiguration": {
      "CPU": "1 vCPU",
      "Memory": "3 GB"
    }
  }
}
```

Contoh 3: Untuk membuat layanan repositori gambar sumber

`create-service` Contoh berikut membuat layanan App Runner berdasarkan gambar yang disimpan di Elastic Container Registry (ECR).

```
aws apprunner create-service \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceName": "golang-container-app",  
  "SourceConfiguration": {  
    "AuthenticationConfiguration": {  
      "AccessRoleArn": "arn:aws:iam::123456789012:role/my-ecr-role"  
    },  
    "AutoDeploymentsEnabled": true,  
    "ImageRepository": {  
      "ImageIdentifier": "123456789012.dkr.ecr.us-east-1.amazonaws.com/golang-  
app:latest",  
      "ImageConfiguration": {  
        "Port": "8080",  
        "RuntimeEnvironmentVariables": [  
          {  
            "NAME": "Jane"  
          }  
        ]  
      },  
      "ImageRepositoryType": "ECR"  
    }  
  },  
  "InstanceConfiguration": {  
    "CPU": "1 vCPU",  
    "Memory": "3 GB"  
  }  
}
```

Output:

```
{  
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
  "Service": {  
    "CreatedAt": "2020-11-06T23:15:30Z",  
    "UpdatedAt": "2020-11-06T23:15:30Z",
```



```

    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/golang-
container-app/51728f8a20ce46d39b25398a6c8e9d1a",
    "ServiceId": "51728f8a20ce46d39b25398a6c8e9d1a",
    "ServiceName": "golang-container-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "AccessRoleArn": "arn:aws:iam::123456789012:role/my-ecr-role"
      },
      "AutoDeploymentsEnabled": true,
      "ImageRepository": {
        "ImageIdentifier": "123456789012.dkr.ecr.us-east-1.amazonaws.com/
golang-app:latest",
        "ImageConfiguration": {
          "Port": "8080",
          "RuntimeEnvironmentVariables": [
            {
              "NAME": "Jane"
            }
          ]
        },
        "ImageRepositoryType": "ECR"
      }
    },
    "Status": "OPERATION_IN_PROGRESS",
    "InstanceConfiguration": {
      "CPU": "1 vCPU",
      "Memory": "3 GB"
    }
  }
}

```

- Untuk detail API, lihat [CreateService](#) di Referensi AWS CLI Perintah.

## delete-auto-scaling-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-auto-scaling-configuration`.

### AWS CLI

Contoh 1: Untuk menghapus revisi aktif terbaru dari konfigurasi penskalaan otomatis

`delete-auto-scaling-configuration` Contoh berikut menghapus revisi aktif terbaru dari konfigurasi penskalaan otomatis App Runner. Untuk menghapus revisi aktif terbaru, tentukan Nama Sumber Daya Amazon (ARN) yang diakhiri dengan nama konfigurasi, tanpa komponen revisi.

Dalam contoh, ada dua revisi sebelum tindakan ini. Oleh karena itu, revisi 2 (terbaru) dihapus. Namun, sekarang ditampilkan `"Latest": false`, karena, setelah dihapus, itu bukan revisi aktif terbaru lagi.

```
aws apprunner delete-auto-scaling-configuration \  
--cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-availability"  
}
```

Output:

```
{  
  "AutoScalingConfiguration": {  
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-availability/2/  
e76562f50d78042e819fead0f59672e6",  
    "AutoScalingConfigurationName": "high-availability",  
    "AutoScalingConfigurationRevision": 2,  
    "CreatedAt": "2021-02-25T17:42:59Z",  
    "DeletedAt": "2021-03-02T08:07:06Z",  
    "Latest": false,  
    "Status": "INACTIVE",  
    "MaxConcurrency": 30,  
    "MaxSize": 90,  
    "MinSize": 5  
  }  
}
```

Contoh 2: Untuk menghapus revisi tertentu dari konfigurasi penskalaan otomatis

`delete-auto-scaling-configuration` Contoh berikut menghapus revisi spesifik konfigurasi penskalaan otomatis App Runner. Untuk menghapus revisi tertentu, tentukan ARN yang menyertakan nomor revisi.

Dalam contoh, beberapa revisi ada sebelum tindakan ini. Tindakan menghapus 1 revisi.

```
aws apprunner delete-auto-scaling-configuration \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-availability/1"  
}
```

Output:

```
{  
  "AutoScalingConfiguration": {  
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-  
availability/1/2f50e7656d7819fead0f59672e68042e",  
    "AutoScalingConfigurationName": "high-availability",  
    "AutoScalingConfigurationRevision": 1,  
    "CreatedAt": "2020-11-03T00:29:17Z",  
    "DeletedAt": "2021-03-02T08:07:06Z",  
    "Latest": false,  
    "Status": "INACTIVE",  
    "MaxConcurrency": 100,  
    "MaxSize": 50,  
    "MinSize": 5  
  }  
}
```

- Untuk detail API, lihat [DeleteAutoScalingConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-connection**

Contoh kode berikut menunjukkan cara menggunakan `delete-connection`.

## AWS CLI

Untuk menghapus koneksi

`delete-connection` Contoh berikut menghapus koneksi App Runner. Status koneksi setelah panggilan berhasil adalah `DELETED`. Ini karena koneksi tidak lagi tersedia.

```
aws apprunner delete-connection \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-  
connection"  
}
```

Output:

```
{  
  "Connection": {  
    "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-  
github-connection",  
    "ConnectionName": "my-github-connection",  
    "Status": "DELETED",  
    "CreatedAt": "2020-11-03T00:32:51Z",  
    "ProviderType": "GITHUB"  
  }  
}
```

- Untuk detail API, lihat [DeleteConnection](#) di Referensi AWS CLI Perintah.

## `delete-service`

Contoh kode berikut menunjukkan cara menggunakan `delete-service`.

## AWS CLI

Untuk menghapus layanan

`delete-service` Contoh berikut menghapus layanan App Runner.

```
aws apprunner delete-service \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Output:

```
{  
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
  "Service": {  
    "CreatedAt": "2020-11-20T19:05:25Z",  
    "UpdatedAt": "2020-11-20T19:05:25Z",  
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceName": "python-app",  
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",  
    "SourceConfiguration": {  
      "AuthenticationConfiguration": {  
        "ConnectionArn": "arn:aws:apprunner:us-  
east-1:123456789012:connection/my-github-connection/  
e7656250f67242d7819feade6800f59e"  
      },  
      "AutoDeploymentsEnabled": true,  
      "CodeRepository": {  
        "CodeConfiguration": {  
          "CodeConfigurationValues": {  
            "BuildCommand": "pip install -r requirements.txt",  
            "Port": "8080",  
            "Runtime": "PYTHON_3",  
            "RuntimeEnvironmentVariables": [  
              {  
                "NAME": "Jane"  
              }  
            ],  
            "StartCommand": "python server.py"  
          }  
        }  
      }  
    }  
  }  
}
```

```

        "ConfigurationSource": "Api"
      },
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      }
    }
  },
  "Status": "OPERATION_IN_PROGRESS",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}

```

- Untuk detail API, lihat [DeleteService](#) di Referensi AWS CLI Perintah.

## describe-auto-scaling-configuration

Contoh kode berikut menunjukkan cara menggunakan `describe-auto-scaling-configuration`.

### AWS CLI

Contoh 1: Untuk menjelaskan revisi aktif terbaru dari konfigurasi penskalaan otomatis

`describe-auto-scaling-configuration` Contoh berikut mendapatkan deskripsi revisi aktif terbaru dari konfigurasi penskalaan otomatis App Runner. Untuk menjelaskan revisi aktif terbaru, tentukan ARN yang diakhiri dengan nama konfigurasi, tanpa komponen revisi.

Dalam contoh tersebut, ada dua revisi. Oleh karena itu, revisi 2 (terbaru) dijelaskan. Objek yang dihasilkan menunjukkan `"Latest": true`.

```
aws apprunner describe-auto-scaling-configuration \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
```

```
"AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability"
}
```

### Output:

```
{
  "AutoScalingConfiguration": {
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/2/
e76562f50d78042e819fead0f59672e6",
    "AutoScalingConfigurationName": "high-availability",
    "AutoScalingConfigurationRevision": 2,
    "CreatedAt": "2021-02-25T17:42:59Z",
    "Latest": true,
    "Status": "ACTIVE",
    "MaxConcurrency": 30,
    "MaxSize": 90,
    "MinSize": 5
  }
}
```

Contoh 2: Untuk menjelaskan revisi spesifik dari konfigurasi penskalaan otomatis

`describe-auto-scaling-configuration` Contoh berikut mendapatkan deskripsi revisi spesifik konfigurasi penskalaan otomatis App Runner. Untuk menggambarkan revisi tertentu, tentukan ARN yang menyertakan nomor revisi.

Dalam contoh, beberapa revisi ada dan revisi 1 ditanyakan. Objek yang dihasilkan menunjukkan `"Latest": false`.

```
aws apprunner describe-auto-scaling-configuration \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/1"
}
```

**Output:**

```
{
  "AutoScalingConfiguration": {
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
    "AutoScalingConfigurationName": "high-availability",
    "AutoScalingConfigurationRevision": 1,
    "CreatedAt": "2020-11-03T00:29:17Z",
    "Latest": false,
    "Status": "ACTIVE",
    "MaxConcurrency": 100,
    "MaxSize": 50,
    "MinSize": 5
  }
}
```

- Untuk detail API, lihat [DescribeAutoScalingConfiguration](#) di Referensi AWS CLI Perintah.

**describe-custom-domains**

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-domains`.

**AWS CLI**

Untuk mendapatkan deskripsi nama domain kustom yang terkait dengan layanan

`describe-custom-domains` Contoh berikut mendapatkan deskripsi dan status nama domain kustom yang terkait dengan layanan App Runner.

```
aws apprunner describe-custom-domains \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
  "DomainName": "example.com",
  "EnableWWWSubdomain": true
}
```



```
}

```

## Output:

```
{
  "CustomDomains": [
    {
      "CertificateValidationRecords": [
        {
          "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",
          "Status": "PENDING_VALIDATION",
          "Type": "CNAME",
          "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-
validations.aws."
        },
        {
          "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",
          "Status": "PENDING_VALIDATION",
          "Type": "CNAME",
          "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-
validations.aws."
        }
      ],
      "DomainName": "example.com",
      "EnableWWWSubdomain": true,
      "Status": "PENDING_CERTIFICATE_DNS_VALIDATION"
    },
    {
      "CertificateValidationRecords": [
        {
          "Name": "_a94f784c70d3f507c72dc28f55db2f6b.deals.example.com",
          "Status": "SUCCESS",
          "Type": "CNAME",
          "Value": "_2db02504c1270c137383c6307b6834b0.bsgbmzkfwj.acm-
validations.aws."
        }
      ],
      "DomainName": "deals.example.com",
      "EnableWWWSubdomain": false,
      "Status": "ACTIVE"
    }
  ],
  "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",

```

```
"ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

- Untuk detail API, lihat [DescribeCustomDomains](#) di Referensi AWS CLI Perintah.

## describe-service

Contoh kode berikut menunjukkan cara menggunakan `describe-service`.

### AWS CLI

Untuk mendeskripsikan layanan

`describe-service` Contoh berikut mendapatkan deskripsi layanan App Runner.

```
aws apprunner describe-service \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-20T19:05:25Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      }
    }
  }
}
```

```

    },
    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "CodeConfiguration": {
        "CodeConfigurationValues": {
          "BuildCommand": "pip install -r requirements.txt",
          "Port": "8080",
          "Runtime": "PYTHON_3",
          "RuntimeEnvironmentVariables": [
            {
              "NAME": "Jane"
            }
          ],
          "StartCommand": "python server.py"
        },
        "ConfigurationSource": "Api"
      },
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      }
    }
  },
  "Status": "RUNNING",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}
}

```

- Untuk detail API, lihat [DescribeService](#) di Referensi AWS CLI Perintah.

## **disassociate-custom-domain**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-custom-domain`.

### AWS CLI

Untuk memisahkan nama domain dari layanan

`disassociate-custom-domain` Contoh berikut memisahkan domain `example.com` dari layanan App Runner. Panggilan juga memisahkan subdomain `www.example.com` yang dikaitkan bersama dengan domain root.

```
aws apprunner disassociate-custom-domain \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "DomainName": "example.com"  
}
```

Output:

```
{  
  "CustomDomain": {  
    "CertificateValidationRecords": [  
      {  
        "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",  
        "Status": "PENDING_VALIDATION",  
        "Type": "CNAME",  
        "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-  
validations.aws."  
      },  
      {  
        "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",  
        "Status": "PENDING_VALIDATION",  
        "Type": "CNAME",  
        "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-  
validations.aws."  
      }  
    ],  
    "DomainName": "example.com",  
    "EnableWWWSubdomain": true,  
    "Status": "DELETING"  
  },  
  "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"
```

```
}
```

- Untuk detail API, lihat [DisassociateCustomDomain](#) di Referensi AWS CLI Perintah.

## list-auto-scaling-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-auto-scaling-configurations`.

### AWS CLI

Untuk mendapatkan daftar paginasi konfigurasi penskalaan otomatis App Runner

`list-auto-scaling-configurations` Contoh berikut mencantumkan semua konfigurasi penskalaan otomatis App Runner di akun Anda. AWS Hingga lima konfigurasi penskalaan otomatis tercantum di setiap respons. `AutoScalingConfigurationName` dan `LatestOnly` tidak ditentukan. Defaultnya menyebabkan revisi terbaru dari semua konfigurasi aktif terdaftar.

Dalam contoh ini, respons mencakup dua hasil dan tidak ada yang tambahan, jadi `NextToken` tidak dikembalikan.

```
aws apprunner list-auto-scaling-configurations \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "MaxResults": 5
}
```

Output:

```
{
  "AutoScalingConfigurationSummaryList": [
    {
      "AutoScalingConfigurationArn": "arn:aws:apprunner:us-east-1:123456789012:autoscalingconfiguration/high-availability/2/e76562f50d78042e819fead0f59672e6",
      "AutoScalingConfigurationName": "high-availability",
      "AutoScalingConfigurationRevision": 2
    },
    {
```

```

        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/low-
cost/1/50d7804e7656fead0f59672e62f2e819",
        "AutoScalingConfigurationName": "low-cost",
        "AutoScalingConfigurationRevision": 1
    }
]
}

```

- Untuk detail API, lihat [ListAutoScalingConfigurations](#) di Referensi AWS CLI Perintah.

## list-connections

Contoh kode berikut menunjukkan cara menggunakan `list-connections`.

### AWS CLI

Contoh 1: Untuk daftar semua koneksi

`list-connections` Contoh berikut mencantumkan semua koneksi App Runner di AWS akun.

```
aws apprunner list-connections
```

Output:

```

{
  "ConnectionSummaryList": [
    {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-connection",
      "ConnectionName": "my-github-connection",
      "Status": "AVAILABLE",
      "CreatedAt": "2020-11-03T00:32:51Z",
      "ProviderType": "GITHUB"
    },
    {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-org-connection",
      "ConnectionName": "my-github-org-connection",
      "Status": "AVAILABLE",
      "CreatedAt": "2020-11-03T02:54:17Z",
      "ProviderType": "GITHUB"
    }
  ]
}

```

```

    }
  ]
}

```

Contoh 2: Untuk membuat daftar koneksi dengan nama

`list-connections` Contoh berikut mencantumkan koneksi dengan namanya.

```

aws apprunner list-connections \
  --cli-input-json file://input.json

```

Isi dari `input.json`:

```

{
  "ConnectionName": "my-github-org-connection"
}

```

Output:

```

{
  "ConnectionSummaryList": [
    {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-org-connection",
      "ConnectionName": "my-github-org-connection",
      "Status": "AVAILABLE",
      "CreatedAt": "2020-11-03T02:54:17Z",
      "ProviderType": "GITHUB"
    }
  ]
}

```

- Untuk detail API, lihat [ListConnections](#) di Referensi AWS CLI Perintah.

## list-operations

Contoh kode berikut menunjukkan cara menggunakan `list-operations`.

### AWS CLI

Untuk membuat daftar operasi yang terjadi pada servicee

`list-operations` Contoh berikut mencantumkan semua operasi yang terjadi pada layanan App Runner sejauh ini. Dalam contoh ini, layanan ini baru dan hanya satu operasi tipe yang `CREATE_SERVICE` telah terjadi.

```
aws apprunner list-operations \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Output:

```
{  
  "OperationSummaryList": [  
    {  
      "EndedAt": 1606156217,  
      "Id": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
      "StartedAt": 1606156014,  
      "Status": "SUCCEEDED",  
      "TargetArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
      "Type": "CREATE_SERVICE",  
      "UpdatedAt": 1606156217  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListOperations](#) di Referensi AWS CLI Perintah.

## **list-services**

Contoh kode berikut menunjukkan cara menggunakan `list-services`.

### AWS CLI

Untuk mendapatkan daftar paginasi layanan App Runner



`list-services` Contoh berikut mencantumkan semua layanan App Runner di AWS akun. Hingga dua layanan tercantum dalam setiap tanggapan. Contoh ini menunjukkan permintaan pertama. Respons mencakup dua hasil dan token yang dapat digunakan dalam permintaan berikutnya. Ketika respons berikutnya tidak menyertakan token, semua layanan telah terdaftar.

```
aws apprunner list-services \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "MaxResults": 2  
}
```

Output:

```
{  
  "NextToken":  
  "eyJDDdXN0b211ckFjY291bnRjZCI6IjI3MDIwNTQwMjg0NSIsI1N1cnZpY2VTdGF0dXNDb2R1IjoiUFJpVnk1TSU90SU"  
  "ServiceSummaryList": [  
    {  
      "CreatedAt": "2020-11-20T19:05:25Z",  
      "UpdatedAt": "2020-11-23T12:41:37Z",  
      "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-app/8fe1e10304f84fd2b0df550fe98a71fa",  
      "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",  
      "ServiceName": "python-app",  
      "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",  
      "Status": "RUNNING"  
    },  
    {  
      "CreatedAt": "2020-11-06T23:15:30Z",  
      "UpdatedAt": "2020-11-23T13:21:22Z",  
      "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/golang-container-app/ab8f94cfe29a460fb8760afd2ee87555",  
      "ServiceId": "ab8f94cfe29a460fb8760afd2ee87555",  
      "ServiceName": "golang-container-app",  
      "ServiceUrl": "e2m8rrrx33.us-east-1.awsapprunner.com",  
      "Status": "RUNNING"  
    }  
  ]  
}
```

```
}
```

- Untuk detail API, lihat [ListServices](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang terkait dengan layanan App Runner

`list-tags-for-resource` Contoh berikut mencantumkan semua tag yang terkait dengan layanan App Runner.

```
aws apprunner list-tags-for-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Department",  
      "Value": "Retail"  
    },  
    {  
      "Key": "CustomerId",  
      "Value": "56439872357912"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## pause-service

Contoh kode berikut menunjukkan cara menggunakan `pause-service`.

### AWS CLI

Untuk menjeda layanan

`pause-service` Contoh berikut menjeda layanan App Runner.

```
aws apprunner pause-service \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Output:

```
{  
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
  "Service": {  
    "CreatedAt": "2020-11-20T19:05:25Z",  
    "UpdatedAt": "2020-11-23T12:41:37Z",  
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceName": "python-app",  
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",  
    "SourceConfiguration": {  
      "AuthenticationConfiguration": {  
        "ConnectionArn": "arn:aws:apprunner:us-  
east-1:123456789012:connection/my-github-connection/  
e7656250f67242d7819feade6800f59e"  
      },  
      "AutoDeploymentsEnabled": true,  
      "CodeRepository": {  
        "CodeConfiguration": {  
          "CodeConfigurationValues": {  
            "BuildCommand": "pip install -r requirements.txt",  

```

```
        "Port": "8080",
        "Runtime": "PYTHON_3",
        "RuntimeEnvironmentVariables": [
            {
                "NAME": "Jane"
            }
        ],
        "StartCommand": "python server.py"
    },
    "ConfigurationSource": "Api"
},
"RepositoryUrl": "https://github.com/my-account/python-hello",
"SourceCodeVersion": {
    "Type": "BRANCH",
    "Value": "main"
}
},
"Status": "OPERATION_IN_PROGRESS",
"InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
}
}
}
```

- Untuk detail API, lihat [PauseService](#) di Referensi AWS CLI Perintah.

## resume-service

Contoh kode berikut menunjukkan cara menggunakan `resume-service`.

### AWS CLI

Untuk melanjutkan layanan

`resume-service` Contoh berikut melanjutkan layanan App Runner.

```
aws apprunner resume-service \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

### Output:

```
{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-23T12:41:37Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      },
      "AutoDeploymentsEnabled": true,
      "CodeRepository": {
        "CodeConfiguration": {
          "CodeConfigurationValues": {
            "BuildCommand": "pip install -r requirements.txt",
            "Port": "8080",
            "Runtime": "PYTHON_3",
            "RuntimeEnvironmentVariables": [
              {
                "NAME": "Jane"
              }
            ],
            "StartCommand": "python server.py"
          },
          "ConfigurationSource": "Api"
        },
        "RepositoryUrl": "https://github.com/my-account/python-hello",
        "SourceCodeVersion": {
          "Type": "BRANCH",
          "Value": "main"
        }
      }
    }
  }
}
```

```
    }
  },
  "Status": "OPERATION_IN_PROGRESS",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}
```

- Untuk detail API, lihat [ResumeService](#) di Referensi AWS CLI Perintah.

## start-deployment

Contoh kode berikut menunjukkan cara menggunakan `start-deployment`.

### AWS CLI

Untuk memulai penerapan manual

`start-deployment` Contoh berikut melakukan penerapan manual ke layanan App Runner.

```
aws apprunner start-deployment \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
  "OperationId": "853a7d5b-fc9f-4730-831b-fd8037ab832a"
}
```

- Untuk detail API, lihat [StartDeployment](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke layanan App Runner

`tag-resource` Contoh berikut menambahkan dua tag ke layanan App Runner.

```
aws apprunner tag-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "Tags": [  
    {  
      "Key": "Department",  
      "Value": "Retail"  
    },  
    {  
      "Key": "CustomerId",  
      "Value": "56439872357912"  
    }  
  ]  
}
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari layanan App Runner

`untag-resource` Contoh berikut menghapus dua tag dari layanan App Runner.

```
aws apprunner untag-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "TagKeys": [  
    "Department",  
    "CustomerId"  
  ]  
}
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-service

Contoh kode berikut menunjukkan cara menggunakan `update-service`.

### AWS CLI

Untuk memperbarui ukuran memori

`update-service` Contoh berikut memperbarui ukuran memori instance (unit penskalaan) layanan App Runner menjadi 2048 MiB.

Ketika panggilan berhasil, App Runner memulai proses pembaruan asinkron. `Service` Struktur yang dikembalikan oleh panggilan mencerminkan nilai memori baru yang sedang diterapkan oleh panggilan ini.

```
aws apprunner update-service \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",
```



```

    "InstanceConfiguration": {
      "Memory": "4 GB"
    }
  }
}

```

### Output:

```

{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-23T12:41:37Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      },
      "AutoDeploymentsEnabled": true,
      "CodeRepository": {
        "CodeConfiguration": {
          "CodeConfigurationValues": {
            "BuildCommand": "pip install -r requirements.txt",
            "Port": "8080",
            "Runtime": "PYTHON_3",
            "RuntimeEnvironmentVariables": [
              {
                "NAME": "Jane"
              }
            ],
            "StartCommand": "python server.py"
          },
          "ConfigurationSource": "Api"
        },
        "RepositoryUrl": "https://github.com/my-account/python-hello",
        "SourceCodeVersion": {
          "Type": "BRANCH",
          "Value": "main"
        }
      }
    }
  }
}

```

```
    }
  },
  "Status": "OPERATION_IN_PROGRESS",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "4 GB"
  }
}
```

- Untuk detail API, lihat [UpdateService](#) di Referensi AWS CLI Perintah.

## AWS AppConfig contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS AppConfig.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-application**

Contoh kode berikut menunjukkan cara menggunakan `create-application`.

AWS CLI

Untuk membuat aplikasi

**create-application** Contoh berikut membuat aplikasi di AWS AppConfig.

```
aws appconfig create-application \  
  --name "example-application" \  
  --description "An application used for creating an example."
```

Output:

```
{  
  "Description": "An application used for creating an example.",  
  "Id": "339ohji",  
  "Name": "example-application"  
}
```

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [CreateApplication](#) di Referensi AWS CLI Perintah.

## **create-configuration-profile**

Contoh kode berikut menunjukkan cara menggunakan `create-configuration-profile`.

### AWS CLI

Untuk membuat profil konfigurasi

**create-configuration-profile** Contoh berikut membuat profil konfigurasi menggunakan konfigurasi yang disimpan di Parameter Store, kemampuan Systems Manager.

```
aws appconfig create-configuration-profile \  
  --application-id "339ohji" \  
  --name "Example-Configuration-Profile" \  
  --location-uri "ssm-parameter://Example-Parameter" \  
  --retrieval-role-arn "arn:aws:iam::111122223333:role/Example-App-Config-Role"
```

Output:

```
{  
  "ApplicationId": "339ohji",  
  "Description": null,  
}
```

```
"Id": "ur8hx2f",
"LocationUri": "ssm-parameter://Example-Parameter",
"Name": "Example-Configuration-Profile",
"RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role",
"Type": null,
"Validators": null
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [CreateConfigurationProfile](#) di Referensi AWS CLI Perintah.

## create-environment

Contoh kode berikut menunjukkan cara menggunakan `create-environment`.

### AWS CLI

Untuk menciptakan lingkungan

`create-environment` Contoh berikut menciptakan AWS AppConfig lingkungan bernama `Example-Environment` menggunakan aplikasi yang Anda buat menggunakan `create-application`.

```
aws appconfig create-environment \
  --application-id "339ohji" \
  --name "Example-Environment"
```

Output:

```
{
  "ApplicationId": "339ohji",
  "Description": null,
  "Id": "54j1r29",
  "Monitors": null,
  "Name": "Example-Environment",
  "State": "ReadyForDeployment"
}
```

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat lingkungan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [CreateEnvironment](#) di Referensi AWS CLI Perintah.

## create-extension-association

Contoh kode berikut menunjukkan cara menggunakan `create-extension-association`.

### AWS CLI

Untuk membuat asosiasi ekstensi

`create-extension-association` Contoh berikut membuat asosiasi ekstensi baru di AWS AppConfig.

```
aws appconfig create-extension-association \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension \  
  --resource-identifier "arn:aws:appconfig:us-west-2:123456789012:application/  
Finance" \  
  --parameters S3bucket=FinanceConfigurationBackup
```

Output:

```
{  
  "Id": "a1b2c3d4",  
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-  
extension/1",  
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",  
  "Parameters": {  
    "S3bucket": "FinanceConfigurationBackup"  
  },  
  "ExtensionVersionNumber": 1  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [CreateExtensionAssociation](#) di Referensi AWS CLI Perintah.

## create-extension

Contoh kode berikut menunjukkan cara menggunakan `create-extension`.

## AWS CLI

Untuk membuat ekstensi

`create-extension` Contoh berikut membuat ekstensi baru di AWS AppConfig.

```
aws appconfig create-extension \  
  --region us-west-2 \  
  --name S3-backup-extension \  
  --actions  
  PRE_CREATE_HOSTED_CONFIGURATION_VERSION=[{Name=S3backup,Uri=arn:aws:lambda:us-  
west-2:123456789012:function:s3backupfunction,RoleArn=arn:aws:iam::123456789012:role/  
appconfigextensionrole}] \  
  --parameters S3bucket={Required=true}
```

Output:

```
{  
  "Id": "1A2B3C4D",  
  "Name": "S3-backup-extension",  
  "VersionNumber": 1,  
  "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1",  
  "Actions": {  
    "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [  
      {  
        "Name": "S3backup",  
        "Uri": "arn:aws:lambda:us-  
west-2:123456789012:function:s3backupfunction",  
        "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"  
      }  
    ]  
  },  
  "Parameters": {  
    "S3bucket": {  
      "Required": true  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [CreateExtension](#) di Referensi AWS CLI Perintah.

## create-hosted-configuration-version

Contoh kode berikut menunjukkan cara menggunakan `create-hosted-configuration-version`.

### AWS CLI

Untuk membuat versi konfigurasi yang dihosting

`create-hosted-configuration-version` Contoh berikut membuat konfigurasi baru di toko konfigurasi yang AWS AppConfig dihosting. Konten konfigurasi harus terlebih dahulu dikonversi ke base64.

```
aws appconfig create-hosted-configuration-version \
  --application-id "339ohji" \
  --configuration-profile-id "ur8hx2f" \
  --content
eyAiTmFtZSI6ICJFeGFtcGx1QXBwbGljYXRpb24iLCAiSWQiOiBFFeGFtcGx1SUQsICJSYW5rIjogNyB9 \
  --content-type "application/json" \
  configuration_version_output_file
```

Isi dari `configuration_version_output_file`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Output:

```
{
  "ApplicationId": "339ohji",
  "ConfigurationProfileId": "ur8hx2f",
  "VersionNumber": "1",
  "ContentType": "application/json"
}
```

Untuk informasi selengkapnya, lihat [Tentang penyimpanan konfigurasi yang AWS AppConfig dihosting](#) di Panduan Pengguna AWS AppConfig.

- Untuk detail API, lihat [CreateHostedConfigurationVersion](#) di Referensi AWS CLI Perintah.

## delete-application

Contoh kode berikut menunjukkan cara menggunakan `delete-application`.

## AWS CLI

Untuk menghapus aplikasi

`delete-application` Contoh berikut menghapus aplikasi yang ditentukan.

```
aws appconfig delete-application \  
--application-id 339ohji
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteApplication](#) di Referensi AWS CLI Perintah.

## **delete-configuration-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-configuration-profile`.

## AWS CLI

Untuk menghapus profil konfigurasi

`delete-configuration-profile` Contoh berikut menghapus profil konfigurasi yang ditentukan.

```
aws appconfig delete-configuration-profile \  
--application-id 339ohji \  
--configuration-profile-id ur8hx2f
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteConfigurationProfile](#) di Referensi AWS CLI Perintah.

## **delete-deployment-strategy**

Contoh kode berikut menunjukkan cara menggunakan `delete-deployment-strategy`.



## AWS CLI

Untuk menghapus strategi penerapan

`delete-deployment-strategy` Contoh berikut menghapus strategi penyebaran yang ditentukan.

```
aws appconfig delete-deployment-strategy \  
  --deployment-strategy-id 1225qzk
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat strategi penerapan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteDeploymentStrategy](#) di Referensi AWS CLI Perintah.

## `delete-environment`

Contoh kode berikut menunjukkan cara menggunakan `delete-environment`.

## AWS CLI

Untuk menghapus lingkungan

`delete-environment` Contoh berikut menghapus lingkungan aplikasi yang ditentukan.

```
aws appconfig delete-environment \  
  --application-id 339ohji \  
  --environment-id 54j1r29
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat lingkungan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteEnvironment](#) di Referensi AWS CLI Perintah.

## `delete-extension-association`

Contoh kode berikut menunjukkan cara menggunakan `delete-extension-association`.

## AWS CLI

Untuk menghapus asosiasi ekstensi

`delete-extension-association` Contoh berikut menghapus asosiasi ekstensi dari AWS AppConfig.

```
aws appconfig delete-extension-association \  
  --region us-west-2 \  
  --extension-association-id a1b2c3d4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteExtensionAssociation](#) di Referensi AWS CLI Perintah.

## **delete-extension**

Contoh kode berikut menunjukkan cara menggunakan `delete-extension`.

## AWS CLI

Untuk menghapus ekstensi

`delete-extension` Contoh berikut menghapus ekstensi dari AWS AppConfig.

```
aws appconfig delete-extension \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteExtension](#) di Referensi AWS CLI Perintah.

## **delete-hosted-configuration-version**

Contoh kode berikut menunjukkan cara menggunakan `delete-hosted-configuration-version`.

## AWS CLI

Untuk menghapus versi konfigurasi yang dihosting

`delete-hosted-configuration-version` Contoh berikut menghapus versi konfigurasi yang dihosting di toko konfigurasi yang AWS AppConfig dihosting.

```
aws appconfig delete-hosted-configuration-version \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f \  
  --version-number 1
```

Output:: Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [DeleteHostedConfigurationVersion](#) di Referensi AWS CLI Perintah.

## get-application

Contoh kode berikut menunjukkan cara menggunakan `get-application`.

## AWS CLI

Untuk membuat daftar detail aplikasi

`get-application` Contoh berikut mencantumkan rincian aplikasi yang ditentukan.

```
aws appconfig get-application \  
  --application-id 339ohji
```

Output:

```
{  
  "Description": "An application used for creating an example.",  
  "Id": "339ohji",  
  "Name": "example-application"  
}
```

Untuk informasi selengkapnya, lihat [Cara AWS AppConfig kerja](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetApplication](#) di Referensi AWS CLI Perintah.

## get-configuration-profile

Contoh kode berikut menunjukkan cara menggunakan `get-configuration-profile`.

### AWS CLI

Untuk mengambil detail profil konfigurasi

`get-configuration-profile` Contoh berikut mengembalikan rincian profil konfigurasi yang ditentukan.

```
aws appconfig get-configuration-profile \
  --application-id 339ohji \
  --configuration-profile-id ur8hx2f
```

Output:

```
{
  "ApplicationId": "339ohji",
  "Id": "ur8hx2f",
  "Name": "Example-Configuration-Profile",
  "LocationUri": "ssm-parameter://Example-Parameter",
  "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role"
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetConfigurationProfile](#) di Referensi AWS CLI Perintah.

## get-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-configuration`.

### AWS CLI

Untuk mengambil detail konfigurasi

`get-configuration` Contoh berikut mengembalikan rincian konfigurasi aplikasi contoh. Pada panggilan berikutnya untuk mendapatkan konfigurasi, gunakan `client-configuration-`

`version` parameter untuk hanya memperbarui konfigurasi aplikasi Anda jika versi telah berubah. Hanya memperbarui konfigurasi ketika versi telah berubah menghindari biaya berlebih yang dikeluarkan dengan memanggil `get-configuration`.

```
aws appconfig get-configuration \  
  --application "example-application" \  
  --environment "Example-Environment" \  
  --configuration "Example-Configuration-Profile" \  
  --client-id "test-id" \  
  configuration-output-file
```

Isi dari `configuration-output-file`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Output:

```
{  
  "ConfigurationVersion": "1",  
  "ContentType": "application/json"  
}
```

Untuk informasi selengkapnya, lihat [Langkah 6: Menerima konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetConfiguration](#) di Referensi AWS CLI Perintah.

## get-deployment-strategy

Contoh kode berikut menunjukkan cara menggunakan `get-deployment-strategy`.

AWS CLI

Untuk mengambil rincian strategi penyebaran

`get-deployment-strategy` Contoh berikut mencantumkan rincian strategi penyebaran yang ditentukan.

```
aws appconfig get-deployment-strategy \  
  --deployment-strategy-id 1225qzk
```

**Output:**

```
{
  "Id": "1225qzk",
  "Name": "Example-Deployment",
  "DeploymentDurationInMinutes": 15,
  "GrowthType": "LINEAR",
  "GrowthFactor": 25.0,
  "FinalBakeTimeInMinutes": 0,
  "ReplicateTo": "SSM_DOCUMENT"
}
```

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat strategi penerapan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetDeploymentStrategy](#) di Referensi AWS CLI Perintah.

**get-deployment**

Contoh kode berikut menunjukkan cara menggunakan `get-deployment`.

**AWS CLI**

Untuk mengambil detail penerapan

`get-deployment` Contoh berikut mencantumkan rincian penyebaran ke aplikasi di lingkungan dan penyebaran yang ditentukan.

```
aws appconfig get-deployment \
  --application-id 339ohji \
  --environment-id 54j1r29 \
  --deployment-number 1
```

**Output:**

```
{
  "ApplicationId": "339ohji",
  "EnvironmentId": "54j1r29",
  "DeploymentStrategyId": "1225qzk",
  "ConfigurationProfileId": "ur8hx2f",
  "DeploymentNumber": 1,
  "ConfigurationName": "Example-Configuration-Profile",
}
```

```
"ConfigurationLocationUri": "ssm-parameter://Example-Parameter",
"ConfigurationVersion": "1",
"DeploymentDurationInMinutes": 15,
"GrowthType": "LINEAR",
"GrowthFactor": 25.0,
"FinalBakeTimeInMinutes": 0,
"State": "COMPLETE",
"EventLog": [
  {
    "EventType": "DEPLOYMENT_COMPLETED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Deployment completed",
    "OccurredAt": "2021-09-17T21:59:03.888000+00:00"
  },
  {
    "EventType": "BAKE_TIME_STARTED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Deployment bake time started",
    "OccurredAt": "2021-09-17T21:58:57.722000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 100.00% of clients",
    "OccurredAt": "2021-09-17T21:55:56.816000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 75.00% of clients",
    "OccurredAt": "2021-09-17T21:52:56.567000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 50.00% of clients",
    "OccurredAt": "2021-09-17T21:49:55.737000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 25.00% of clients",
    "OccurredAt": "2021-09-17T21:46:55.187000+00:00"
  },
],
```

```
{
  "EventType": "DEPLOYMENT_STARTED",
  "TriggeredBy": "USER",
  "Description": "Deployment started",
  "OccurredAt": "2021-09-17T21:43:54.205000+00:00"
},
"PercentageComplete": 100.0,
"StartedAt": "2021-09-17T21:43:54.205000+00:00",
"CompletedAt": "2021-09-17T21:59:03.888000+00:00"
}
```

Untuk informasi selengkapnya, lihat [Langkah 5: Menerapkan konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetDeployment](#) di Referensi AWS CLI Perintah.

## get-environment

Contoh kode berikut menunjukkan cara menggunakan `get-environment`.

### AWS CLI

Untuk mengambil detail lingkungan

`get-environment` Contoh berikut mengembalikan rincian dan keadaan lingkungan tertentu.

```
aws appconfig get-environment \
  --application-id 339ohji \
  --environment-id 54j1r29
```

Output:

```
{
  "ApplicationId": "339ohji",
  "Id": "54j1r29",
  "Name": "Example-Environment",
  "State": "ReadyForDeployment"
}
```

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat lingkungan](#) di Panduan AWS AppConfig Pengguna.



- Untuk detail API, lihat [GetEnvironment](#) di Referensi AWS CLI Perintah.

## get-extension-association

Contoh kode berikut menunjukkan cara menggunakan `get-extension-association`.

### AWS CLI

Untuk mendapatkan detail asosiasi ekstensi

`get-extension-association` Contoh berikut menampilkan informasi tentang asosiasi ekstensi.

```
aws appconfig get-extension-association \
  --region us-west-2 \
  --extension-association-id a1b2c3d4
```

Output:

```
{
  "Id": "a1b2c3d4",
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-extension/1",
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
  "Parameters": {
    "S3bucket": "FinanceConfigurationBackup"
  },
  "ExtensionVersionNumber": 1
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetExtensionAssociation](#) di Referensi AWS CLI Perintah.

## get-extension

Contoh kode berikut menunjukkan cara menggunakan `get-extension`.

### AWS CLI

Untuk mendapatkan detail ekstensi

`get-extension` Contoh berikut menampilkan informasi tentang ekstensi.

```
aws appconfig get-extension \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension
```

Output:

```
{  
  "Id": "1A2B3C4D",  
  "Name": "S3-backup-extension",  
  "VersionNumber": 1,  
  "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-  
extension/1",  
  "Actions": {  
    "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [  
      {  
        "Name": "S3backup",  
        "Uri": "arn:aws:lambda:us-  
west-2:123456789012:function:S3backupfunction",  
        "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"  
      }  
    ]  
  },  
  "Parameters": {  
    "S3bucket": {  
      "Required": true  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetExtension](#) di Referensi AWS CLI Perintah.

## **get-hosted-configuration-version**

Contoh kode berikut menunjukkan cara menggunakan `get-hosted-configuration-version`.

### AWS CLI

Untuk memulihkan detail konfigurasi yang dihosting

`get-hosted-configuration-version` Contoh berikut mengambil detail konfigurasi konfigurasi yang AWS AppConfig dihosting.

```
aws appconfig get-hosted-configuration-version \
  --application-id 339ohji \
  --configuration-profile-id ur8hx2f \
  --version-number 1 \
  hosted-configuration-version-output
```

Isi dari `hosted-configuration-version-output`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Output:

```
{
  "ApplicationId": "339ohji",
  "ConfigurationProfileId": "ur8hx2f",
  "VersionNumber": "1",
  "ContentType": "application/json"
}
```

Untuk informasi selengkapnya, lihat [Tentang penyimpanan konfigurasi yang AWS AppConfig dihosting](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [GetHostedConfigurationVersion](#) di Referensi AWS CLI Perintah.

## **list-applications**

Contoh kode berikut menunjukkan cara menggunakan `list-applications`.

### AWS CLI

Untuk daftar aplikasi yang tersedia

`list-applications` Contoh berikut mencantumkan aplikasi yang tersedia di AWS akun Anda.

```
aws appconfig list-applications
```

Output:

```
{
  "Items": [
    {
      "Id": "339ohji",
      "Name": "test-application",
      "Description": "An application used for creating an example."
    },
    {
      "Id": "rwalwu7",
      "Name": "Test-Application"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListApplications](#) di Referensi AWS CLI Perintah.

## list-configuration-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-configuration-profiles`.

### AWS CLI

Untuk membuat daftar profil konfigurasi yang tersedia

`list-configuration-profiles` Contoh berikut mencantumkan profil konfigurasi yang tersedia untuk aplikasi yang ditentukan.

```
aws appconfig list-configuration-profiles \
  --application-id 339ohji
```

Output:

```
{
  "Items": [
    {
      "ApplicationId": "339ohji",
      "Id": "ur8hx2f",
      "Name": "Example-Configuration-Profile",
      "LocationUri": "ssm-parameter://Example-Parameter"
    }
  ]
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListConfigurationProfiles](#) di Referensi AWS CLI Perintah.

## list-deployment-strategies

Contoh kode berikut menunjukkan cara menggunakan `list-deployment-strategies`.

### AWS CLI

Untuk membuat daftar strategi penyebaran yang tersedia

`list-deployment-strategies` Contoh berikut mencantumkan strategi penyebaran yang tersedia di AWS akun Anda.

```
aws appconfig list-deployment-strategies
```

Output:

```
{  
  "Items": [  
    {  
      "Id": "1225qzk",  
      "Name": "Example-Deployment",  
      "DeploymentDurationInMinutes": 15,  
      "GrowthType": "LINEAR",  
      "GrowthFactor": 25.0,  
      "FinalBakeTimeInMinutes": 0,  
      "ReplicateTo": "SSM_DOCUMENT"  
    },  
    {  
      "Id": "AppConfig.AllAtOnce",  
      "Name": "AppConfig.AllAtOnce",  
      "Description": "Quick",  
      "DeploymentDurationInMinutes": 0,  
      "GrowthType": "LINEAR",  
      "GrowthFactor": 100.0,  
      "FinalBakeTimeInMinutes": 10,  
    }  
  ]  
}
```

```

        "ReplicateTo": "NONE"
    },
    {
        "Id": "AppConfig.Linear50PercentEvery30Seconds",
        "Name": "AppConfig.Linear50PercentEvery30Seconds",
        "Description": "Test/Demo",
        "DeploymentDurationInMinutes": 1,
        "GrowthType": "LINEAR",
        "GrowthFactor": 50.0,
        "FinalBakeTimeInMinutes": 1,
        "ReplicateTo": "NONE"
    },
    {
        "Id": "AppConfig.Canary10Percent20Minutes",
        "Name": "AppConfig.Canary10Percent20Minutes",
        "Description": "AWS Recommended",
        "DeploymentDurationInMinutes": 20,
        "GrowthType": "EXPONENTIAL",
        "GrowthFactor": 10.0,
        "FinalBakeTimeInMinutes": 10,
        "ReplicateTo": "NONE"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat strategi penerapan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListDeploymentStrategies](#) di Referensi AWS CLI Perintah.

## list-deployments

Contoh kode berikut menunjukkan cara menggunakan `list-deployments`.

### AWS CLI

Untuk membuat daftar penerapan yang tersedia

`list-deployments` Contoh berikut mencantumkan penerapan yang tersedia di AWS akun Anda untuk aplikasi dan lingkungan yang ditentukan.

```
aws appconfig list-deployments \
  --application-id 339ohji \
```

```
--environment-id 54j1r29
```

Output:

```
{
  "Items": [
    {
      "DeploymentNumber": 1,
      "ConfigurationName": "Example-Configuration-Profile",
      "ConfigurationVersion": "1",
      "DeploymentDurationInMinutes": 15,
      "GrowthType": "LINEAR",
      "GrowthFactor": 25.0,
      "FinalBakeTimeInMinutes": 0,
      "State": "COMPLETE",
      "PercentageComplete": 100.0,
      "StartedAt": "2021-09-17T21:43:54.205000+00:00",
      "CompletedAt": "2021-09-17T21:59:03.888000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Langkah 5: Menerapkan konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListDeployments](#) di Referensi AWS CLI Perintah.

## list-environments

Contoh kode berikut menunjukkan cara menggunakan `list-environments`.

### AWS CLI

Untuk membuat daftar lingkungan yang tersedia

`list-environments` Contoh berikut mencantumkan lingkungan yang tersedia di AWS akun Anda untuk aplikasi yang ditentukan.

```
aws appconfig list-environments \
  --application-id 339ohji
```

Output:

```
{
  "Items": [
    {
      "ApplicationId": "339ohji",
      "Id": "54j1r29",
      "Name": "Example-Environment",
      "State": "ReadyForDeployment"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat lingkungan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListEnvironments](#) di Referensi AWS CLI Perintah.

## list-extension-associations

Contoh kode berikut menunjukkan cara menggunakan `list-extension-associations`.

### AWS CLI

Untuk mencantumkan semua asosiasi AWS AppConfig ekstensi di AWS akun Anda untuk AWS Wilayah

`list-extension-associations` Contoh berikut mencantumkan semua asosiasi AWS AppConfig ekstensi untuk AWS akun saat ini di AWS Wilayah tertentu.

```
aws appconfig list-extension-associations \
  --region us-west-2
```

### Output:

```
{
  "Items": [
    {
      "Id": "a1b2c3d4",
      "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-
backup-extension/1",
      "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/
Finance"
    }
  ]
}
```



```
]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListExtensionAssociations](#) di Referensi AWS CLI Perintah.

## list-extensions

Contoh kode berikut menunjukkan cara menggunakan `list-extensions`.

### AWS CLI

Untuk mencantumkan semua AWS AppConfig ekstensi di AWS akun Anda untuk AWS Wilayah

`list-extensions` Contoh berikut mencantumkan semua AWS AppConfig ekstensi untuk AWS akun saat ini di AWS Wilayah tertentu. Perintah mengembalikan ekstensi kustom dan AWS ditulis.

```
aws appconfig list-extensions \
  --region us-west-2
```

Output:

```
{
  "Items": [
    {
      "Id": "1A2B3C4D",
      "Name": "S3-backup-extension",
      "VersionNumber": 1,
      "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1"
    },
    {
      "Id": "AWS.AppConfig.FeatureFlags",
      "Name": "AppConfig Feature Flags Helper",
      "VersionNumber": 1,
      "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.FeatureFlags/1",
      "Description": "Validates AppConfig feature flag data automatically
against a JSON schema that includes structure and constraints. Also transforms
feature flag data prior to sending to the client. This extension is automatically
associated to configuration profiles with type \"AWS.AppConfig.FeatureFlags\"."
    },
  ],
}
```

```
{
  "Id": "AWS.AppConfig.JiraIntegration",
  "Name": "AppConfig integration with Atlassian Jira",
  "VersionNumber": 1,
  "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.JiraIntegration/1",
  "Description": "Exports feature flag data from AWS AppConfig into
Jira. The lifecycle of each feature flag in AppConfig is tracked in Jira as an
individual issue. Customers can see in Jira when flags are updated, turned on or
off. Works in conjunction with the AppConfig app in the Atlassian Marketplace and
is automatically associated to configuration profiles configured within that app."
},
{
  "Id": "AWS.AppConfig.DeploymentNotificationsToEventBridge",
  "Name": "AppConfig deployment events to Amazon EventBridge",
  "VersionNumber": 1,
  "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.DeploymentNotificationsToEventBridge/1",
  "Description": "Sends events to Amazon EventBridge when a deployment
of configuration data in AppConfig is started, completed, or rolled back. Can
be associated to the following resources in AppConfig: Application, Environment,
Configuration Profile."
},
{
  "Id": "AWS.AppConfig.DeploymentNotificationsToSqs",
  "Name": "AppConfig deployment events to Amazon SQS",
  "VersionNumber": 1,
  "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.DeploymentNotificationsToSqs/1",
  "Description": "Sends messages to the configured Amazon SQS queue when
a deployment of configuration data in AppConfig is started, completed, or rolled
back. Can be associated to the following resources in AppConfig: Application,
Environment, Configuration Profile."
},
{
  "Id": "AWS.AppConfig.DeploymentNotificationsToSns",
  "Name": "AppConfig deployment events to Amazon SNS",
  "VersionNumber": 1,
  "Description": "Sends events to the configured Amazon SNS topic when
a deployment of configuration data in AppConfig is started, completed, or rolled
back. Can be associated to the following resources in AppConfig: Application,
Environment, Configuration Profile."
}
]
```

```
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListExtensions](#) di Referensi AWS CLI Perintah.

## list-hosted-configuration-versions

Contoh kode berikut menunjukkan cara menggunakan `list-hosted-configuration-versions`.

### AWS CLI

Untuk mencantumkan versi konfigurasi yang dihosting yang tersedia

`list-hosted-configuration-versions` Contoh berikut mencantumkan versi konfigurasi yang dihosting di toko konfigurasi yang AWS AppConfig dihosting untuk profil aplikasi dan konfigurasi yang ditentukan.

```
aws appconfig list-hosted-configuration-versions \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f
```

### Output:

```
{  
  "Items": [  
    {  
      "ApplicationId": "339ohji",  
      "ConfigurationProfileId": "ur8hx2f",  
      "VersionNumber": 1,  
      "ContentType": "application/json"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Tentang penyimpanan konfigurasi yang AWS AppConfig dihosting](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListHostedConfigurationVersions](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag aplikasi

`list-tags-for-resource` Contoh berikut mencantumkan tag aplikasi tertentu.

```
aws appconfig list-tags-for-resource \
  --resource-arn arn:aws:appconfig:us-east-1:682428703967:application/339ohji
```

Output:

```
{
  "Tags": {
    "group1": "1"
  }
}
```

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## start-deployment

Contoh kode berikut menunjukkan cara menggunakan `start-deployment`.

### AWS CLI

Untuk memulai penerapan konfigurasi

`start-deployment` Contoh berikut memulai penyebaran ke aplikasi menggunakan lingkungan yang ditentukan, strategi penyebaran, dan profil konfigurasi.

```
aws appconfig start-deployment \
  --application-id 339ohji \
  --environment-id 54j1r29 \
  --deployment-strategy-id 1225qzk \
```

```
--configuration-profile-id ur8hx2f \  
--configuration-version 1
```

### Output:

```
{  
  "ApplicationId": "339ohji",  
  "EnvironmentId": "54j1r29",  
  "DeploymentStrategyId": "1225qzk",  
  "ConfigurationProfileId": "ur8hx2f",  
  "DeploymentNumber": 1,  
  "ConfigurationName": "Example-Configuration-Profile",  
  "ConfigurationLocationUri": "ssm-parameter://Example-Parameter",  
  "ConfigurationVersion": "1",  
  "DeploymentDurationInMinutes": 15,  
  "GrowthType": "LINEAR",  
  "GrowthFactor": 25.0,  
  "FinalBakeTimeInMinutes": 0,  
  "State": "DEPLOYING",  
  "EventLog": [  
    {  
      "EventType": "DEPLOYMENT_STARTED",  
      "TriggeredBy": "USER",  
      "Description": "Deployment started",  
      "OccurredAt": "2021-09-17T21:43:54.205000+00:00"  
    }  
  ],  
  "PercentageComplete": 0.0,  
  "StartedAt": "2021-09-17T21:43:54.205000+00:00"  
}
```

Untuk informasi selengkapnya, lihat [Langkah 5: Menerapkan konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [StartDeployment](#) di Referensi AWS CLI Perintah.

## stop-deployment

Contoh kode berikut menunjukkan cara menggunakan `stop-deployment`.

### AWS CLI

Untuk menghentikan penerapan konfigurasi

`stop-deployment` Contoh berikut menghentikan penyebaran konfigurasi aplikasi ke lingkungan yang ditentukan.

```
aws appconfig stop-deployment \  
  --application-id 339ohji \  
  --environment-id 54j1r29 \  
  --deployment-number 2
```

Output:

```
{  
  "DeploymentNumber": 0,  
  "DeploymentDurationInMinutes": 0,  
  "GrowthFactor": 0.0,  
  "FinalBakeTimeInMinutes": 0,  
  "PercentageComplete": 0.0  
}
```

Untuk informasi selengkapnya, lihat [Langkah 5: Menerapkan konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [StopDeployment](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai aplikasi

`tag-resource` Contoh berikut menandai sumber daya aplikasi.

```
aws appconfig tag-resource \  
  --resource-arn arn:aws:appconfig:us-east-1:682428703967:application/339ohji \  
  --tags '{"group1" : "1"}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari aplikasi

`untag-resource` Contoh berikut menghapus tag `group1` dari aplikasi yang ditentukan.

```
aws appconfig untag-resource \  
  --resource-arn arn:aws:appconfig:us-east-1:111122223333:application/339ohji \  
  --tag-keys ["group1"]
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-application

Contoh kode berikut menunjukkan cara menggunakan `update-application`.

### AWS CLI

Untuk memperbarui aplikasi

`update-application` Contoh berikut memperbarui nama aplikasi yang ditentukan.

```
aws appconfig update-application \  
  --application-id 339ohji \  
  --name "Example-Application"
```

Output:

```
{  
  "Id": "339ohji",  
  "Name": "Example-Application",  
  "Description": "An application used for creating an example."
```

```
}
```

Untuk informasi selengkapnya, lihat [Langkah 1: Membuat AWS AppConfig aplikasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UpdateApplication](#) di Referensi AWS CLI Perintah.

## update-configuration-profile

Contoh kode berikut menunjukkan cara menggunakan `update-configuration-profile`.

### AWS CLI

Untuk memperbarui profil konfigurasi

`update-configuration-profile` Contoh berikut memperbarui deskripsi profil konfigurasi yang ditentukan.

```
aws appconfig update-configuration-profile \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f \  
  --description "Configuration profile used for examples."
```

Output:

```
{  
  "ApplicationId": "339ohji",  
  "Id": "ur8hx2f",  
  "Name": "Example-Configuration-Profile",  
  "Description": "Configuration profile used for examples.",  
  "LocationUri": "ssm-parameter://Example-Parameter",  
  "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role"  
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UpdateConfigurationProfile](#) di Referensi AWS CLI Perintah.

## update-deployment-strategy

Contoh kode berikut menunjukkan cara menggunakan `update-deployment-strategy`.



## AWS CLI

Untuk memperbarui strategi penerapan

`update-deployment-strategy` Contoh berikut memperbarui waktu pemanggangan akhir menjadi 20 menit dalam strategi penerapan yang ditentukan.

```
aws appconfig update-deployment-strategy \  
  --deployment-strategy-id 1225qzk \  
  --final-bake-time-in-minutes 20
```

Output:

```
{  
  "Id": "1225qzk",  
  "Name": "Example-Deployment",  
  "DeploymentDurationInMinutes": 15,  
  "GrowthType": "LINEAR",  
  "GrowthFactor": 25.0,  
  "FinalBakeTimeInMinutes": 20,  
  "ReplicateTo": "SSM_DOCUMENT"  
}
```

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat strategi penerapan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UpdateDeploymentStrategy](#) di Referensi AWS CLI Perintah.

## update-environment

Contoh kode berikut menunjukkan cara menggunakan `update-environment`.

### AWS CLI

Untuk memperbarui lingkungan

`update-environment` Contoh berikut memperbarui deskripsi lingkungan.

```
aws appconfig update-environment \  
  --application-id 339ohji \  
  --environment-id 54j1r29 \  
  --description "New description"
```

```
--description "An environment for examples."
```

Output:

```
{
  "ApplicationId": "339ohji",
  "Id": "54j1r29",
  "Name": "Example-Environment",
  "Description": "An environment for examples.",
  "State": "RolledBack"
}
```

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat lingkungan](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UpdateEnvironment](#) di Referensi AWS CLI Perintah.

## update-extension-association

Contoh kode berikut menunjukkan cara menggunakan `update-extension-association`.

### AWS CLI

Untuk memperbarui asosiasi AWS AppConfig ekstensi

`update-extension-association` Contoh berikut menambahkan nilai parameter baru ke asosiasi ekstensi di AWS AppConfig.

```
aws appconfig update-extension-association \
  --region us-west-2 \
  --extension-association-id a1b2c3d4 \
  --parameters S3bucket=FinanceMobileApp
```

Output:

```
{
  "Id": "a1b2c3d4",
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-extension/1",
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
  "Parameters": {
    "S3bucket": "FinanceMobileApp"
  }
}
```

```

    },
    "ExtensionVersionNumber": 1
  }

```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UpdateExtensionAssociation](#) di Referensi AWS CLI Perintah.

## update-extension

Contoh kode berikut menunjukkan cara menggunakan `update-extension`.

### AWS CLI

Untuk memperbarui AWS AppConfig ekstensi

`update-extension` Contoh berikut menambahkan parameter tambahan Kunci untuk ekstensi di AWS AppConfig.

```

aws appconfig update-extension \
  --region us-west-2 \
  --extension-identifier S3-backup-extension \
  --parameters S3bucket={Required=true},CampaignID={Required=false}

```

Output:

```

{
  "Id": "1A2B3C4D",
  "Name": "S3-backup-extension",
  "VersionNumber": 1,
  "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1",
  "Actions": {
    "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [
      {
        "Name": "S3backup",
        "Uri": "arn:aws:lambda:us-west-2:123456789012:function:S3backupfunction",
        "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"
      }
    ]
  },
}

```

```
"Parameters": {
  "CampaignID": {
    "Required": false
  },
  "S3bucket": {
    "Required": true
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS AppConfig ekstensi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [UpdateExtension](#) di Referensi AWS CLI Perintah.

## validate-configuration

Contoh kode berikut menunjukkan cara menggunakan `validate-configuration`.

### AWS CLI

Untuk memvalidasi konfigurasi

`validate-configuration` Contoh berikut menggunakan validator dalam profil konfigurasi untuk memvalidasi konfigurasi.

```
aws appconfig validate-configuration \
  --application-id abc1234 \
  --configuration-profile-id ur8hx2f \
  --configuration-version 1
```

Perintah tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat konfigurasi dan profil konfigurasi](#) di Panduan AWS AppConfig Pengguna.

- Untuk detail API, lihat [ValidateConfiguration](#) di Referensi AWS CLI Perintah.

## Contoh Application Auto Scaling menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Application Auto Scaling AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-scaling-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-scaling-policy`.

AWS CLI

Untuk menghapus kebijakan penskalaan

Contoh ini menghapus kebijakan penskalaan untuk aplikasi web layanan Amazon ECS yang berjalan di klaster default.

Perintah:

```
aws application-autoscaling delete-scaling-policy --policy-name web-app-cpu-1t-25 --scalable-dimension ecs:service:DesiredCount --resource-id service/default/web-app --service-namespace ecs
```

- Untuk detail API, lihat [DeleteScalingPolicy](#) di Referensi AWS CLI Perintah.

### **delete-scheduled-action**

Contoh kode berikut menunjukkan cara menggunakan `delete-scheduled-action`.

AWS CLI

Untuk menghapus tindakan terjadwal

`delete-scheduled-action` Contoh berikut menghapus tindakan terjadwal yang ditentukan dari armada Amazon AppStream 2.0 yang ditentukan:

```
aws application-autoscaling delete-scheduled-action \  
  --service-namespace appstream \  
  --scalable-dimension appstream:fleet:DesiredCapacity \  
  --resource-id fleet/sample-fleet \  
  --scheduled-action-name my-recurring-action
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Penskalaan Terjadwal](#) di Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [DeleteScheduledAction](#) di Referensi AWS CLI Perintah.

## **deregister-scalable-target**

Contoh kode berikut menunjukkan cara menggunakan `deregister-scalable-target`.

### AWS CLI

Untuk membatalkan pendaftaran target yang dapat diskalakan

Contoh ini membatalkan pendaftaran target yang dapat diskalakan untuk layanan Amazon ECS yang disebut `web-app` yang berjalan di kluster default.

Perintah:

```
aws application-autoscaling deregister-scalable-target --service-namespace ecs --  
scalable-dimension ecs:service:DesiredCount --resource-id service/default/web-app
```

Contoh ini membatalkan pendaftaran target yang dapat diskalakan untuk sumber daya kustom. `custom-resource-idFile.txt` berisi string yang mengidentifikasi ID Sumber Daya, yang, untuk sumber daya kustom, adalah jalur ke sumber daya kustom melalui titik akhir Amazon API Gateway Anda.

Perintah:

```
aws application-autoscaling deregister-scalable-target --service-namespace custom-  
resource --scalable-dimension custom-resource:ResourceType:Property --resource-id  
file://~/custom-resource-id.txt
```

Isi custom-resource-id file.txt:

```
https://example.execute-api.us-west-2.amazonaws.com/prod/
scalableTargetDimensions/1-23456789
```

- Untuk detail API, lihat [DeregisterScalableTarget](#) di Referensi AWS CLI Perintah.

## describe-scalable-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-scalable-targets`.

### AWS CLI

Untuk menggambarkan target yang dapat diskalakan

`describe-scalable-targets` Contoh berikut menjelaskan target yang dapat diskalakan untuk namespace ecs layanan.

```
aws application-autoscaling describe-scalable-targets \
  --service-namespace ecs
```

Output:

```
{
  "ScalableTargets": [
    {
      "ServiceNamespace": "ecs",
      "ScalableDimension": "ecs:service:DesiredCount",
      "ResourceId": "service/default/web-app",
      "MinCapacity": 1,
      "MaxCapacity": 10,
      "RoleARN": "arn:aws:iam::123456789012:role/
aws-service-role/ecs.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_ECSService",
      "CreationTime": 1462558906.199,
      "SuspendedState": {
        "DynamicScalingOutSuspended": false,
        "ScheduledScalingSuspended": false,
        "DynamicScalingInSuspended": false
      },
      "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [AWS layanan yang dapat Anda gunakan dengan Application Auto Scaling di Panduan Pengguna Application Auto Scaling](#).

- Untuk detail API, lihat [DescribeScalableTargets](#) di Referensi AWS CLI Perintah.

## describe-scaling-activities

Contoh kode berikut menunjukkan cara menggunakan `describe-scaling-activities`.

### AWS CLI

Contoh 1: Untuk menjelaskan aktivitas penskalaan untuk layanan Amazon ECS yang ditentukan

`describe-scaling-activities` Contoh berikut menjelaskan aktivitas penskalaan untuk layanan Amazon ECS web-app yang disebut yang berjalan di default cluster. Output menunjukkan aktivitas penskalaan yang diprakarsai oleh kebijakan penskalaan.

```

aws application-autoscaling describe-scaling-activities \
  --service-namespace ecs \
  --resource-id service/default/web-app

```

Output:

```

{
  "ScalingActivities": [
    {
      "ScalableDimension": "ecs:service:DesiredCount",
      "Description": "Setting desired count to 1.",
      "ResourceId": "service/default/web-app",
      "ActivityId": "e6c5f7d1-dbbb-4a3f-89b2-51f33e766399",
      "StartTime": 1462575838.171,
      "ServiceNamespace": "ecs",
      "EndTime": 1462575872.111,
      "Cause": "monitor alarm web-app-cpu-lt-25 in state ALARM triggered policy web-app-cpu-lt-25",
      "StatusMessage": "Successfully set desired count to 1. Change successfully fulfilled by ecs.",
    }
  ]
}

```



```

        "StatusCode": "Successful"
      }
    ]
  }

```

Untuk informasi selengkapnya, lihat [Aktivitas penskalaan untuk Application Auto Scaling](#) di Panduan Pengguna Application Auto Scaling.

Contoh 2: Untuk menggambarkan aktivitas penskalaan untuk tabel DynamoDB yang ditentukan `describe-scaling-activities` Contoh berikut menjelaskan aktivitas penskalaan untuk tabel DynamoDB yang disebut `TestTable` Output menunjukkan aktivitas penskalaan yang diprakarsai oleh dua tindakan terjadwal yang berbeda.

```

aws application-autoscaling describe-scaling-activities \
  --service-namespace dynamodb \
  --resource-id table/TestTable

```

Output:

```

{
  "ScalingActivities": [
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting write capacity units to 10.",
      "ResourceId": "table/my-table",
      "ActivityId": "4d1308c0-bbcf-4514-a673-b0220ae38547",
      "StartTime": 1561574415.086,
      "ServiceNamespace": "dynamodb",
      "EndTime": 1561574449.51,
      "Cause": "maximum capacity was set to 10",
      "StatusMessage": "Successfully set write capacity units to 10. Change
successfully fulfilled by dynamodb.",
      "StatusCode": "Successful"
    },
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting min capacity to 5 and max capacity to 10",
      "ResourceId": "table/my-table",
      "ActivityId": "f2b7847b-721d-4e01-8ef0-0c8d3bacc1c7",
      "StartTime": 1561574414.644,
      "ServiceNamespace": "dynamodb",

```

```

    "Cause": "scheduled action name my-second-scheduled-action was
triggered",
    "StatusMessage": "Successfully set min capacity to 5 and max capacity to
10",
    "StatusCode": "Successful"
  },
  {
    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting write capacity units to 15.",
    "ResourceId": "table/my-table",
    "ActivityId": "d8ea4de6-9eaa-499f-b466-2cc5e681ba8b",
    "StartTime": 1561574108.904,
    "ServiceNamespace": "dynamodb",
    "EndTime": 1561574140.255,
    "Cause": "minimum capacity was set to 15",
    "StatusMessage": "Successfully set write capacity units to 15. Change
successfully fulfilled by dynamodb.",
    "StatusCode": "Successful"
  },
  {
    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting min capacity to 15 and max capacity to 20",
    "ResourceId": "table/my-table",
    "ActivityId": "3250fd06-6940-4e8e-bb1f-d494db7554d2",
    "StartTime": 1561574108.512,
    "ServiceNamespace": "dynamodb",
    "Cause": "scheduled action name my-first-scheduled-action was
triggered",
    "StatusMessage": "Successfully set min capacity to 15 and max capacity
to 20",
    "StatusCode": "Successful"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Aktivitas penskalaan untuk Application Auto Scaling](#) di Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [DescribeScalingActivities](#) di Referensi AWS CLI Perintah.

## describe-scaling-policies

Contoh kode berikut menunjukkan cara menggunakan `describe-scaling-policies`.

## AWS CLI

Untuk menggambarkan kebijakan penskalaan

Perintah contoh ini menjelaskan kebijakan penskalaan untuk namespace layanan ecs.

Perintah:

```
aws application-autoscaling describe-scaling-policies --service-namespace ecs
```

Output:

```
{
  "ScalingPolicies": [
    {
      "PolicyName": "web-app-cpu-gt-75",
      "ScalableDimension": "ecs:service:DesiredCount",
      "ResourceId": "service/default/web-app",
      "CreationTime": 1462561899.23,
      "StepScalingPolicyConfiguration": {
        "Cooldown": 60,
        "StepAdjustments": [
          {
            "ScalingAdjustment": 200,
            "MetricIntervalLowerBound": 0.0
          }
        ],
        "AdjustmentType": "PercentChangeInCapacity"
      },
      "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/web-app-cpu-gt-75",
      "PolicyType": "StepScaling",
      "Alarms": [
        {
          "AlarmName": "web-app-cpu-gt-75",
          "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:web-app-cpu-gt-75"
        }
      ],
      "ServiceNamespace": "ecs"
    },
    {
      "PolicyName": "web-app-cpu-lt-25",
```

```

    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app",
    "CreationTime": 1462562575.099,
    "StepScalingPolicyConfiguration": {
      "Cooldown": 1,
      "StepAdjustments": [
        {
          "ScalingAdjustment": -50,
          "MetricIntervalUpperBound": 0.0
        }
      ],
      "AdjustmentType": "PercentChangeInCapacity"
    },
    "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/web-app-cpu-lt-25",
    "PolicyType": "StepScaling",
    "Alarms": [
      {
        "AlarmName": "web-app-cpu-lt-25",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:web-app-cpu-lt-25"
      }
    ],
    "ServiceNamespace": "ecs"
  }
]
}

```

- Untuk detail API, lihat [DescribeScalingPolicies](#) di Referensi AWS CLI Perintah.

## describe-scheduled-actions

Contoh kode berikut menunjukkan cara menggunakan `describe-scheduled-actions`.

### AWS CLI

Untuk menggambarkan tindakan terjadwal

`describe-scheduled-actions` Contoh berikut menampilkan rincian untuk tindakan terjadwal untuk namespace layanan tertentu:

```
aws application-autoscaling describe-scheduled-actions \
```

```
--service-namespace dynamodb
```

## Output:

```
{
  "ScheduledActions": [
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Schedule": "at(2019-05-20T18:35:00)",
      "ResourceId": "table/my-table",
      "CreationTime": 1561571888.361,
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:2d36aa3b-cdf9-4565-b290-81db519b227d:resource/
dynamodb/table/my-table:scheduledActionName/my-first-scheduled-action",
      "ScalableTargetAction": {
        "MinCapacity": 15,
        "MaxCapacity": 20
      },
      "ScheduledActionName": "my-first-scheduled-action",
      "ServiceNamespace": "dynamodb"
    },
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Schedule": "at(2019-05-20T18:40:00)",
      "ResourceId": "table/my-table",
      "CreationTime": 1561571946.021,
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:2d36aa3b-cdf9-4565-b290-81db519b227d:resource/
dynamodb/table/my-table:scheduledActionName/my-second-scheduled-action",
      "ScalableTargetAction": {
        "MinCapacity": 5,
        "MaxCapacity": 10
      },
      "ScheduledActionName": "my-second-scheduled-action",
      "ServiceNamespace": "dynamodb"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Penskalaan Terjadwal](#) di Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [DescribeScheduledActions](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag untuk target yang dapat diskalakan

`list-tags-for-resource` Contoh berikut mencantumkan nama kunci tag dan nilai yang dilampirkan ke target skalabel yang ditentukan oleh ARN nya.

```
aws application-autoscaling list-tags-for-resource \
  --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-
  target/1234abcd56ab78cd901ef1234567890ab123
```

Output:

```
{
  "Tags": {
    "environment": "production"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai dukungan untuk Application Auto Scaling](#) di Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-scaling-policy

Contoh kode berikut menunjukkan cara menggunakan `put-scaling-policy`.

### AWS CLI

Contoh 1: Untuk menerapkan kebijakan penskalaan pelacakan target dengan spesifikasi metrik yang telah ditentukan

`put-scaling-policy` Contoh berikut menerapkan kebijakan penskalaan pelacakan target dengan spesifikasi metrik yang telah ditentukan sebelumnya ke layanan Amazon ECS yang disebut `web-app` di kluster default. Kebijakan tersebut menjaga pemanfaatan CPU rata-rata layanan pada 75 persen, dengan periode cooldown scale-out dan scale-in 60 detik. Outputnya berisi ARN dan nama dari dua CloudWatch alarm yang dibuat atas nama Anda.

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \  
--scalable-dimension ecs:service:DesiredCount \  
--resource-id service/default/web-app \  
--policy-name cpu75-target-tracking-scaling-policy --policy-type  
TargetTrackingScaling \  
--target-tracking-scaling-policy-configuration file://config.json
```

Contoh ini mengasumsikan bahwa Anda memiliki file config.json di direktori saat ini dengan konten berikut:

```
{  
  "TargetValue": 75.0,  
  "PredefinedMetricSpecification": {  
    "PredefinedMetricType": "ECSServiceAverageCPUUtilization"  
  },  
  "ScaleOutCooldown": 60,  
  "ScaleInCooldown": 60  
}
```

Output:

```
{  
  "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:6d8972f3-  
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/cpu75-  
target-tracking-scaling-policy",  
  "Alarms": [  
    {  
      "AlarmARN": "arn:aws:cloudwatch:us-  
west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-  
b46e-434a-a60f-3b36d653feca",  
      "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-  
b46e-434a-a60f-3b36d653feca"  
    },  
    {  
      "AlarmARN": "arn:aws:cloudwatch:us-  
west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmLow-1b437334-  
d19b-4a63-a812-6c67aaf2910d",  
      "AlarmName": "TargetTracking-service/default/web-app-AlarmLow-1b437334-  
d19b-4a63-a812-6c67aaf2910d"  
    }  
  ]  
}
```

Contoh 2: Untuk menerapkan kebijakan penskalaan pelacakan target dengan spesifikasi metrik yang disesuaikan

`put-scaling-policy` Contoh berikut menerapkan kebijakan penskalaan pelacakan target dengan spesifikasi metrik yang disesuaikan ke layanan Amazon ECS yang disebut `web-app` di kluster default. Kebijakan tersebut menjaga pemanfaatan rata-rata layanan pada 75 persen, dengan periode cooldown scale-out dan scale-in 60 detik. Outputnya berisi ARN dan nama dari dua CloudWatch alarm yang dibuat atas nama Anda.

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \  
--scalable-dimension ecs:service:DesiredCount \  
--resource-id service/default/web-app \  
--policy-name cms75-target-tracking-scaling-policy \  
--policy-type TargetTrackingScaling \  
--target-tracking-scaling-policy-configuration file://config.json
```

Contoh ini mengasumsikan bahwa Anda memiliki file `config.json` di direktori saat ini dengan konten berikut:

```
{  
  "TargetValue":75.0,  
  "CustomizedMetricSpecification":{  
    "MetricName":"MyUtilizationMetric",  
    "Namespace":"MyNamespace",  
    "Dimensions": [  
      {  
        "Name":"MyOptionalMetricDimensionName",  
        "Value":"MyOptionalMetricDimensionValue"  
      }  
    ],  
    "Statistic":"Average",  
    "Unit":"Percent"  
  },  
  "ScaleOutCooldown": 60,  
  "ScaleInCooldown": 60  
}
```

Output:

```
{
```



```

    "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:
8784a896-b2ba-47a1-b08c-27301cc499a1:resource/ecs/service/default/web-
app:policyName/cms75-target-tracking-scaling-policy",
    "Alarms": [
      {
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-
AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0",
        "AlarmName": "TargetTracking-service/default/web-app-
AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0"
      },
      {
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-
AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4",
        "AlarmName": "TargetTracking-service/default/web-app-
AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4"
      }
    ]
  }
}

```

Contoh 3: Untuk menerapkan kebijakan penskalaan pelacakan target hanya untuk skala keluar put-scaling-policy Contoh berikut menerapkan kebijakan penskalaan pelacakan target ke layanan Amazon ECS yang dipanggil web-app di kluster default. Kebijakan ini digunakan untuk meningkatkan skala layanan ECS ketika RequestCountPerTarget metrik dari Application Load Balancer melebihi ambang batas. Outputnya berisi ARN dan nama CloudWatch alarm yang dibuat atas nama Anda.

```

aws application-autoscaling put-scaling-policy \
  --service-namespace ecs \
  --scalable-dimension ecs:service:DesiredCount \
  --resource-id service/default/web-app \
  --policy-name alb-scale-out-target-tracking-scaling-policy \
  --policy-type TargetTrackingScaling \
  --target-tracking-scaling-policy-configuration file://config.json

```

Isi dari config.json:

```

{
  "TargetValue": 1000.0,
  "PredefinedMetricSpecification": {

```

```

    "PredefinedMetricType": "ALBRequestCountPerTarget",
    "ResourceLabel": "app/EC2Co-EcsE1-1TKLTMITMM0E0/f37c06a68c1748aa/
targetgroup/EC2Co-Defau-LDNM7Q3ZH1ZN/6d4ea56ca2d6a18d"
  },
  "ScaleOutCooldown": 60,
  "ScaleInCooldown": 60,
  "DisableScaleIn": true
}

```

### Output:

```

{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:6d8972f3-
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/alb-
scale-out-target-tracking-scaling-policy",
  "Alarms": [
    {
      "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca",
      "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Kebijakan Penskalaan Pelacakan Target untuk Application Auto Scaling](#) di AWS Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [PutScalingPolicy](#) di Referensi AWS CLI Perintah.

## put-scheduled-action

Contoh kode berikut menunjukkan cara menggunakan `put-scheduled-action`.

### AWS CLI

Untuk menambahkan tindakan terjadwal ke tabel DynamoDB

Contoh ini menambahkan tindakan terjadwal ke tabel DynamoDB `TestTable` dipanggil untuk skala keluar pada jadwal berulang. Pada jadwal yang ditentukan (setiap hari pukul 12:15 WIB), jika

kapasitas saat ini di bawah nilai yang ditentukan MinCapacity, Application Auto Scaling sesuai dengan nilai yang ditentukan oleh MinCapacity

Perintah:

```
aws application-autoscaling put-scheduled-action --service-namespace dynamodb
--scheduled-action-name my-recurring-action --schedule "cron(15 12 * * ? *)" --
resource-id table/TestTable --scalable-dimension dynamodb:table:WriteCapacityUnits
--scalable-target-action MinCapacity=6
```

Untuk informasi selengkapnya, lihat Penskalaan Terjadwal di Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [PutScheduledAction](#) di Referensi AWS CLI Perintah.

## register-scalable-target

Contoh kode berikut menunjukkan cara menggunakan `register-scalable-target`.

AWS CLI

Contoh 1: Untuk mendaftarkan layanan ECS sebagai target yang dapat diskalakan

`register-scalable-target` Contoh berikut mendaftarkan layanan Amazon ECS dengan Application Auto Scaling. Ini juga menambahkan tag dengan nama kunci `environment` dan nilai `production` ke target yang dapat diskalakan.

```
aws application-autoscaling register-scalable-target \
--service-namespace ecs \
--scalable-dimension ecs:service:DesiredCount \
--resource-id service/default/web-app \
--min-capacity 1 --max-capacity 10 \
--tags environment=production
```

Output:

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Untuk contoh untuk AWS layanan lain dan sumber daya khusus, lihat topik dalam [AWS layanan yang dapat Anda gunakan dengan Application Auto Scaling](#) di Panduan Pengguna Application Auto Scaling.

Contoh 2: Untuk menanggihkan aktivitas penskalaan untuk target yang dapat diskalakan

`register-scalable-target` Contoh berikut menanggihkan aktivitas penskalaan untuk target skalabel yang ada.

```
aws application-autoscaling register-scalable-target \
  --service-namespace dynamodb \
  --scalable-dimension dynamodb:table:ReadCapacityUnits \
  --resource-id table/my-table \
  --suspended-state
DynamicScalingInSuspended=true,DynamicScalingOutSuspended=true,ScheduledScalingSuspended=true
```

Output:

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Untuk informasi selengkapnya, lihat [Menanggihkan dan melanjutkan penskalaan untuk Application Auto Scaling di Panduan Pengguna Application Auto Scaling](#).

Contoh 3: Untuk melanjutkan aktivitas penskalaan untuk target yang dapat diskalakan

`register-scalable-target` Contoh berikut melanjutkan aktivitas penskalaan untuk target skalabel yang ada.

```
aws application-autoscaling register-scalable-target \
  --service-namespace dynamodb \
  --scalable-dimension dynamodb:table:ReadCapacityUnits \
  --resource-id table/my-table \
  --suspended-state
DynamicScalingInSuspended=false,DynamicScalingOutSuspended=false,ScheduledScalingSuspended=false
```

Output:

```
{
```

```
"ScalableTargetARN": "arn:aws:application-autoscaling:us-west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Untuk informasi selengkapnya, lihat [Menangguhkan dan melanjutkan penskalaan untuk Application Auto Scaling di Panduan Pengguna Application Auto Scaling](#).

- Untuk detail API, lihat [RegisterScalableTarget](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke target yang dapat diskalakan

`tag-resource` Contoh berikut menambahkan tag dengan nama kunci `environment` dan nilai `production` ke target skalabel yang ditentukan oleh ARN nya.

```
aws application-autoscaling tag-resource \
  --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-
  target/1234abcd56ab78cd901ef1234567890ab123 \
  --tags environment=production
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai dukungan untuk Application Auto Scaling di Panduan Pengguna Application Auto Scaling](#).

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari target yang dapat diskalakan

`untag-resource` Contoh berikut menghapus pasangan tag dengan nama kunci `environment` dari target skalabel yang ditentukan oleh ARN-nya.

```
aws application-autoscaling untag-resource \
  --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-
target/1234abcd56ab78cd901ef1234567890ab123 \
  --tag-keys "environment"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai dukungan untuk Application Auto Scaling](#) di Panduan Pengguna Application Auto Scaling.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Contoh Application Discovery Service menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Application Discovery Service AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-agents**

Contoh kode berikut menunjukkan cara menggunakan `describe-agents`.

AWS CLI

Jelaskan agen dengan status `CollectionStatus` tertentu

Perintah contoh ini menjelaskan agen koleksi dengan status koleksi "STARTED" atau "STOPPED".

Perintah:

```
aws discovery describe-agents --filters
name="collectionStatus",values="STARTED","STOPPED",condition="EQUALS" --max-results
3
```

Output:

```
{
  "Snapshots": [
    {
      "version": "1.0.40.0",
      "agentType": "EC2",
      "hostName": "ip-172-31-40-234",
      "collectionStatus": "STOPPED",
      "agentNetworkInfoList": [
        {
          "macAddress": "06:b5:97:14:fc:0d",
          "ipAddress": "172.31.40.234"
        }
      ],
      "health": "UNKNOWN",
      "agentId": "i-003305c02a776e883",
      "registeredTime": "2016-12-09T19:05:06Z",
      "lastHealthPingTime": "2016-12-09T19:05:10Z"
    },
    {
      "version": "1.0.40.0",
      "agentType": "EC2",
      "hostName": "ip-172-31-39-64",
      "collectionStatus": "STARTED",
      "agentNetworkInfoList": [
        {
          "macAddress": "06:a1:0e:c7:b2:73",
          "ipAddress": "172.31.39.64"
        }
      ],
      "health": "SHUTDOWN",
      "agentId": "i-003a5e5e2b36cf8bd",
      "registeredTime": "2016-11-16T16:36:25Z",
```

```

        "lastHealthPingTime": "2016-11-16T16:47:37Z"
      }
    ]
  }

```

- Untuk detail API, lihat [DescribeAgents](#) di Referensi AWS CLI Perintah.

## describe-configurations

Contoh kode berikut menunjukkan cara menggunakan `describe-configurations`.

### AWS CLI

Jelaskan konfigurasi aset yang dipilih

Perintah contoh ini menjelaskan konfigurasi dari dua server tertentu. Tindakan mendeteksi jenis aset dari ID konfigurasi. Hanya satu jenis aset yang diizinkan per perintah.

Perintah:

```
aws discovery describe-configurations --configuration-ids "d-server-099385097ef9fbcfb" "d-server-0c4f2dd1fee22c6c1"
```

Output:

```

{
  "configurations": [
    {
      "server.performance.maxCpuUsagePct": "0.0",
      "server.performance.maxDiskReadIOPS": "0.0",
      "server.performance.avgCpuUsagePct": "0.0",
      "server.type": "EC2",
      "server.performance.maxNetworkReadsPerSecondInKB": "0.19140625",
      "server.hostName": "ip-172-31-35-152",
      "server.configurationId": "d-server-0c4f2dd1fee22c6c1",
      "server.tags.hasMoreValues": "false",
      "server.performance.minFreeRAMInKB": "1543496.0",
      "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
      "server.performance.maxDiskReadsPerSecondInKB": "0.0",
      "server.applications": "[]",
      "server.performance.numDisks": "1",
      "server.performance.numCpus": "1",
      "server.performance.numCores": "1",

```



```

        "server.performance.maxDiskWriteIOPS": "0.0",
        "server.performance.maxNetworkWritesPerSecondInKB": "0.82421875",
        "server.performance.avgDiskWritesPerSecondInKB": "0.0",
        "server.networkInterfaceInfo": "[{\\"name\\":\\"eth0\\",
\\"macAddress\\":\\"06:A7:7D:3F:54:57\\",\\"ipAddress\\":\\"172.31.35.152\\",\\"netMask\\":
\\"255.255.240.0\\"},{\\"name\\":\\"lo\\",\\"macAddress\\":\\"00:00:00:00:00:00\\",\\"ipAddress
\\":\\"127.0.0.1\\",\\"netMask\\":\\"255.0.0.0\\"},{\\"name\\":\\"eth0\\",\\"macAddress\\":
\\"06:A7:7D:3F:54:57\\",\\"ipAddress\\":\\"fe80::4a7:7dff:fe3f:5457\\",{\\"name\\":\\"lo\\",
\\"macAddress\\":\\"00:00:00:00:00:00\\",\\"ipAddress\\":\\":::1\\"}]",
        "server.performance.avgNetworkReadsPerSecondInKB":
"0.049153645833333333",
        "server.tags": "[]",
        "server.applications.hasMoreValues": "false",
        "server.timeOfCreation": "2016-10-28 23:44:00.0",
        "server.agentId": "i-4447bc1b",
        "server.performance.maxDiskWritesPerSecondInKB": "0.0",
        "server.performance.avgDiskReadIOPS": "0.0",
        "server.performance.avgFreeRAMInKB": "1547210.1333333333",
        "server.performance.avgDiskReadsPerSecondInKB": "0.0",
        "server.performance.avgDiskWriteIOPS": "0.0",
        "server.performance.numNetworkCards": "2",
        "server.hypervisor": "xen",
        "server.networkInterfaceInfo.hasMoreValues": "false",
        "server.performance.avgNetworkWritesPerSecondInKB": "0.1380859375",
        "server.osName": "Linux - Amazon Linux AMI release 2015.03",
        "server.performance.totalRAMInKB": "1694732.0",
        "server.cpuType": "x64"
    },
    {
        "server.performance.maxCpuUsagePct": "100.0",
        "server.performance.maxDiskReadIOPS": "0.0",
        "server.performance.avgCpuUsagePct": "14.733333333333338",
        "server.type": "EC2",
        "server.performance.maxNetworkReadsPerSecondInKB": "13.400390625",
        "server.hostName": "ip-172-31-42-208",
        "server.configurationId": "d-server-099385097ef9fbcbfb",
        "server.tags.hasMoreValues": "false",
        "server.performance.minFreeRAMInKB": "1531104.0",
        "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
        "server.performance.maxDiskReadsPerSecondInKB": "0.0",
        "server.applications": "[]",
        "server.performance.numDisks": "1",
        "server.performance.numCpus": "1",
        "server.performance.numCores": "1",

```

```

        "server.performance.maxDiskWriteIOPS": "1.0",
        "server.performance.maxNetworkWritesPerSecondInKB": "12.271484375",
        "server.performance.avgDiskWritesPerSecondInKB":
"0.5333333333333334",
        "server.networkInterfaceInfo": "[{"name":"eth0",
\\"macAddress\\":\\"06:4A:79:60:75:61\\",\\"ipAddress\\":\\"172.31.42.208\\",\\"netMask
\\":\\"255.255.240.0\\"}, {"name":"eth0",\\"macAddress\\":\\"06:4A:79:60:75:61\\",
\\"ipAddress\\":\\"fe80::44a:79ff:fe60:7561\\"}, {"name":"lo",\\"macAddress\\":
\\"00:00:00:00:00:00\\",\\"ipAddress\\":\\"::1\\"}, {"name":"lo",\\"macAddress\\":
\\"00:00:00:00:00:00\\",\\"ipAddress\\":\\"127.0.0.1\\",\\"netMask\\":\\"255.0.0.0\\"}]",
        "server.performance.avgNetworkReadsPerSecondInKB":
"2.8720052083333334",
        "server.tags": "[]",
        "server.applications.hasMoreValues": "false",
        "server.timeOfCreation": "2016-10-28 23:44:30.0",
        "server.agentId": "i-c142b99e",
        "server.performance.maxDiskWritesPerSecondInKB": "4.0",
        "server.performance.avgDiskReadIOPS": "0.0",
        "server.performance.avgFreeRAMInKB": "1534946.4",
        "server.performance.avgDiskReadsPerSecondInKB": "0.0",
        "server.performance.avgDiskWriteIOPS": "0.13333333333333336",
        "server.performance.numNetworkCards": "2",
        "server.hypervisor": "xen",
        "server.networkInterfaceInfo.hasMoreValues": "false",
        "server.performance.avgNetworkWritesPerSecondInKB":
"1.7977864583333332",
        "server.osName": "Linux - Amazon Linux AMI release 2015.03",
        "server.performance.totalRAMInKB": "1694732.0",
        "server.cpuType": "x64"
    }
]
}

```

Jelaskan konfigurasi aset yang dipilih

Perintah contoh ini menjelaskan konfigurasi dari dua aplikasi yang ditentukan. Tindakan mendeteksi jenis aset dari ID konfigurasi. Hanya satu jenis aset yang diizinkan per perintah.

Perintah:

```
aws discovery describe-configurations --configuration-ids "d-
application-0ac39bc0e4fad0e42" "d-application-02444a45288013764q"
```

**Output:**

```
{
  "configurations": [
    {
      "application.serverCount": "0",
      "application.name": "Application-12345",
      "application.lastModifiedTime": "2016-12-13 23:53:27.0",
      "application.description": "",
      "application.timeOfCreation": "2016-12-13 23:53:27.0",
      "application.configurationId": "d-application-0ac39bc0e4fad0e42"
    },
    {
      "application.serverCount": "0",
      "application.name": "Application-67890",
      "application.lastModifiedTime": "2016-12-13 23:53:33.0",
      "application.description": "",
      "application.timeOfCreation": "2016-12-13 23:53:33.0",
      "application.configurationId": "d-application-02444a45288013764"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeConfigurations](#) di Referensi AWS CLI Perintah.

**list-configurations**

Contoh kode berikut menunjukkan cara menggunakan `list-configurations`.

**AWS CLI**

Untuk membuat daftar semua server yang ditemukan yang memenuhi satu set kondisi filter

Contoh perintah ini mencantumkan server yang ditemukan yang cocok dengan salah satu dari dua pola nama host dan tidak menjalankan Ubuntu.

**Perintah:**

```
aws discovery list-configurations --configuration-type SERVER --filters
name="server.hostName",values="172-31-35","172-31-42",condition="CONTAINS"
name="server.osName",values="Ubuntu",condition="NOT_CONTAINS"
```

**Output:**

```
{
  "configurations": [
    {
      "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
      "server.type": "EC2",
      "server.hostName": "ip-172-31-42-208",
      "server.timeOfCreation": "2016-10-28 23:44:30.0",
      "server.configurationId": "d-server-099385097ef9fbcfb",
      "server.osName": "Linux - Amazon Linux AMI release 2015.03",
      "server.agentId": "i-c142b99e"
    },
    {
      "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
      "server.type": "EC2",
      "server.hostName": "ip-172-31-35-152",
      "server.timeOfCreation": "2016-10-28 23:44:00.0",
      "server.configurationId": "d-server-0c4f2dd1fee22c6c1",
      "server.osName": "Linux - Amazon Linux AMI release 2015.03",
      "server.agentId": "i-4447bc1b"
    }
  ]
}
```

- Untuk detail API, lihat [ListConfigurations](#) di Referensi AWS CLI Perintah.

## AppRegistry contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AppRegistry.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **associate-attribute-group**

Contoh kode berikut menunjukkan cara menggunakan `associate-attribute-group`.

## AWS CLI

Untuk mengaitkan grup atribut

`associate-attribute-group` Contoh berikut mengaitkan grup atribut tertentu di AWS akun Anda ke aplikasi tertentu di AWS akun Anda.

```
aws servicecatalog-appregistry associate-attribute-group \  
  --application "ExampleApplication" \  
  --attribute-group "ExampleAttributeGroup"
```

Output:

```
{  
  "applicationArn": "arn:aws:servicecatalog:us-west-2:813737243517:/  
applications/0ars38r6btoohvpvd9gqrptt91",  
  "attributeGroupArn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-  
groups/01sj5xdwhbw54kejwnt09fnpc1"  
}
```

Untuk informasi selengkapnya, lihat [Mengaitkan dan melepaskan grup atribut](#) dalam Panduan Administrator AWS Service Catalog AppRegistry .

- Untuk detail API, lihat [AssociateAttributeGroup](#) di Referensi AWS CLI Perintah.

### **create-application**

Contoh kode berikut menunjukkan cara menggunakan `create-application`.

## AWS CLI

Untuk membuat aplikasi

**create-application** Contoh berikut membuat aplikasi baru di AWS akun Anda.

```
aws servicecatalog-appregistry create-application \  
  --name "ExampleApplication"
```

Output:

```
{  
  "application": {  
    "id": "0ars38r6btoohvpvd9gqrptt91",  
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/  
applications/0ars38r6btoohvpvd9gqrptt91",  
    "name": "ExampleApplication",  
    "creationTime": "2023-02-28T21:10:10.820000+00:00",  
    "lastUpdateTime": "2023-02-28T21:10:10.820000+00:00",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat aplikasi](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [CreateApplication](#) di Referensi AWS CLI Perintah.

## **create-attribute-group**

Contoh kode berikut menunjukkan cara menggunakan `create-attribute-group`.

AWS CLI

Untuk membuat grup atribut

**create-attribute-group** Contoh berikut membuat grup atribut baru di AWS akun Anda.

```
aws servicecatalog-appregistry create-attribute-group \  
  --name "ExampleAttributeGroup" \  
  --attributes '{"SomeKey1":"SomeValue1","SomeKey2":"SomeValue2"}'
```

Output:

```
{
```

```
"attributeGroup": {
  "id": "01sj5xdwhbw54kejwnt09fnpc1",
  "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpc1",
  "name": "ExampleAttributeGroup",
  "creationTime": "2023-02-28T20:38:01.389000+00:00",
  "lastUpdateTime": "2023-02-28T20:38:01.389000+00:00",
  "tags": {}
}
}
```

Untuk selengkapnya, lihat [Membuat grup atribut](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [CreateAttributeGroup](#) di Referensi AWS CLI Perintah.

## delete-application

Contoh kode berikut menunjukkan cara menggunakan `delete-application`.

### AWS CLI

Untuk menghapus aplikasi

`delete-application` Contoh berikut menghapus aplikasi tertentu di AWS akun Anda.

```
aws servicecatalog-appregistry delete-application \
  --application "ExampleApplication3"
```

Output:

```
{
  "application": {
    "id": "055gw7aynr1i5mbv7kjwtzx5945",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/055gw7aynr1i5mbv7kjwtzx5945",
    "name": "ExampleApplication3",
    "creationTime": "2023-02-28T22:06:28.228000+00:00",
    "lastUpdateTime": "2023-02-28T22:06:28.228000+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus aplikasi](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [DeleteApplication](#) di Referensi AWS CLI Perintah.

## delete-attribute-group

Contoh kode berikut menunjukkan cara menggunakan `delete-attribute-group`.

### AWS CLI

Contoh 8: Untuk menghapus grup atribut

`delete-attribute-group` Contoh berikut menghapus grup atribut tertentu di AWS akun Anda.

```
aws servicecatalog-appregistry delete-attribute-group \  
  --attribute-group "ExampleAttributeGroup3"
```

Output:

```
{  
  "attributeGroup": {  
    "id": "011ge6y3emyjijt8dw8jn6r0hv",  
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-  
groups/011ge6y3emyjijt8dw8jn6r0hv",  
    "name": "ExampleAttributeGroup3",  
    "creationTime": "2023-02-28T22:05:35.224000+00:00",  
    "lastUpdateTime": "2023-02-28T22:05:35.224000+00:00"  
  }  
}
```

Untuk selengkapnya, lihat [Menghapus grup atribut](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [DeleteAttributeGroup](#) di Referensi AWS CLI Perintah.

## get-application

Contoh kode berikut menunjukkan cara menggunakan `get-application`.

### AWS CLI

Untuk mendapatkan aplikasi



`get-application` Contoh berikut mengambil informasi metadata tentang aplikasi tertentu di akun Anda. AWS

```
aws servicecatalog-appregistry get-application \  
--application "ExampleApplication"
```

Output:

```
{  
  "id": "0ars38r6btoohvpvd9gqrptt91",  
  "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/  
applications/0ars38r6btoohvpvd9gqrptt91",  
  "name": "ExampleApplication",  
  "creationTime": "2023-02-28T21:10:10.820000+00:00",  
  "lastUpdateTime": "2023-02-28T21:10:10.820000+00:00",  
  "associatedResourceCount": 0,  
  "tags": {  
    "aws:servicecatalog:applicationName": "ExampleApplication"  
  },  
  "integrations": {  
    "resourceGroup": {  
      "state": "CREATE_COMPLETE",  
      "arn": "arn:aws:resource-groups:us-west-2:813737243517:group/  
AWS_AppRegistry_Application-ExampleApplication"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan detail Aplikasi](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [GetApplication](#) di Referensi AWS CLI Perintah.

## **get-attribute-group**

Contoh kode berikut menunjukkan cara menggunakan `get-attribute-group`.

AWS CLI

Untuk mendapatkan grup atribut

`get-attribute-group` Contoh berikut mengambil grup atribut tertentu di AWS akun Anda.

```
aws servicecatalog-appregistry get-attribute-group \  
  --attribute-group "ExampleAttributeGroup"
```

Output:

```
{  
  "id": "01sj5xdwhbw54kejwnt09fnpc1",  
  "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-  
groups/01sj5xdwhbw54kejwnt09fnpc1",  
  "name": "ExampleAttributeGroup",  
  "attributes": "{\"SomeKey1\": \"SomeValue1\", \"SomeKey2\": \"SomeValue2\"}",  
  "creationTime": "2023-02-28T20:38:01.389000+00:00",  
  "lastUpdateTime": "2023-02-28T20:38:01.389000+00:00",  
  "tags": {  
    "aws:servicecatalog:attributeGroupName": "ExampleAttributeGroup"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola metadata untuk grup atribut](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [GetAttributeGroup](#) di Referensi AWS CLI Perintah.

## list-applications

Contoh kode berikut menunjukkan cara menggunakan `list-applications`.

AWS CLI

Untuk daftar aplikasi

`list-applications` Contoh berikut mengambil daftar semua aplikasi di AWS akun Anda.

```
aws servicecatalog-appregistry list-applications
```

Output:

```
{  
  "applications": [  
    {  
      "id": "03axw94pjfj3uan00tcgbrxnkw",
```

```

        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/03axw94pjfj3uan00tcgbrxnkw",
        "name": "ExampleApplication2",
        "creationTime": "2023-02-28T21:59:34.094000+00:00",
        "lastUpdateTime": "2023-02-28T21:59:34.094000+00:00"
    },
    {
        "id": "055gw7aynr1i5mbv7kjwzx5945",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/055gw7aynr1i5mbv7kjwzx5945",
        "name": "ExampleApplication3",
        "creationTime": "2023-02-28T22:06:28.228000+00:00",
        "lastUpdateTime": "2023-02-28T22:06:28.228000+00:00"
    },
    {
        "id": "0ars38r6btoohvpvd9gqrptt91",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt91",
        "name": "ExampleApplication",
        "description": "This is an example application",
        "creationTime": "2023-02-28T21:10:10.820000+00:00",
        "lastUpdateTime": "2023-02-28T21:24:19.729000+00:00"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat detail aplikasi](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [ListApplications](#) di Referensi AWS CLI Perintah.

## list-associated-attribute-groups

Contoh kode berikut menunjukkan cara menggunakan `list-associated-attribute-groups`.

### AWS CLI

Untuk daftar grup atribut terkait

`list-associated-attribute-groups` Contoh berikut mengambil daftar semua grup atribut di AWS akun Anda yang terkait dengan aplikasi tertentu di AWS akun Anda.

```
aws servicecatalog-appregistry list-associated-attribute-groups \
```

```
--application "ExampleApplication"
```

Output:

```
{
  "attributeGroups": [
    "01sj5xdwhbw54kejwnt09fnpc1"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaitkan dan melepaskan grup atribut](#) dalam Panduan Administrator AWS Service Catalog AppRegistry .

- Untuk detail API, lihat [ListAssociatedAttributeGroups](#) di Referensi AWS CLI Perintah.

## list-attribute-groups-for-application

Contoh kode berikut menunjukkan cara menggunakan `list-attribute-groups-for-application`.

AWS CLI

Untuk daftar grup atribut untuk aplikasi

`list-attribute-groups-for-application` Contoh berikut mencantumkan rincian semua grup atribut di AWS akun Anda yang terkait dengan aplikasi tertentu di AWS akun Anda.

```
aws servicecatalog-appregistry list-attribute-groups-for-application \
  --application "ExampleApplication"
```

Output:

```
{
  "attributeGroupsDetails": [
    {
      "id": "01sj5xdwhbw54kejwnt09fnpc1",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpc1",
      "name": "ExampleAttributeGroup"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Melihat detail grup atribut](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [ListAttributeGroupsForApplication](#) di Referensi AWS CLI Perintah.

## list-attribute-groups

Contoh kode berikut menunjukkan cara menggunakan `list-attribute-groups`.

### AWS CLI

Untuk daftar grup atribut

`list-attribute-groups` Contoh berikut mengambil daftar semua grup atribut di AWS akun Anda.

```
aws servicecatalog-appregistry list-attribute-groups
```

Output:

```
{
  "attributeGroups": [
    {
      "id": "011ge6y3emyjijt8dw8jn6r0hv",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-groups/011ge6y3emyjijt8dw8jn6r0hv",
      "name": "ExampleAttributeGroup3",
      "creationTime": "2023-02-28T22:05:35.224000+00:00",
      "lastUpdateTime": "2023-02-28T22:05:35.224000+00:00"
    },
    {
      "id": "01sj5xdwhbw54kejwnt09fnpc1",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-groups/01sj5xdwhbw54kejwnt09fnpc1",
      "name": "ExampleAttributeGroup",
      "description": "This is an example attribute group",
      "creationTime": "2023-02-28T20:38:01.389000+00:00",
      "lastUpdateTime": "2023-02-28T21:02:04.559000+00:00"
    }
  ]
}
```

```

    },
    {
      "id": "03n1yffgq6d18vwrzxf0c70nm3",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/03n1yffgq6d18vwrzxf0c70nm3",
      "name": "ExampleAttributeGroup2",
      "creationTime": "2023-02-28T21:57:30.687000+00:00",
      "lastUpdateTime": "2023-02-28T21:57:30.687000+00:00"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat detail grup atribut](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [ListAttributeGroups](#) di Referensi AWS CLI Perintah.

## update-application

Contoh kode berikut menunjukkan cara menggunakan `update-application`.

### AWS CLI

Untuk memperbarui aplikasi

`update-application` Contoh berikut memperbarui aplikasi tertentu di AWS akun Anda untuk menyertakan deskripsi.

```

aws servicecatalog-appregistry update-application \
  --application "ExampleApplication" \
  --description "This is an example application"

```

Output:

```

{
  "application": {
    "id": "0ars38r6btoohvpvd9gqrptt9l",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt9l",
    "name": "ExampleApplication",
    "description": "This is an example application",
    "creationTime": "2023-02-28T21:10:10.820000+00:00",

```

```

    "lastUpdateTime": "2023-02-28T21:24:19.729000+00:00",
    "tags": {
      "aws:servicecatalog:applicationName": "ExampleApplication"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Mengedit aplikasi](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [UpdateApplication](#) di Referensi AWS CLI Perintah.

## update-attribute-group

Contoh kode berikut menunjukkan cara menggunakan `update-attribute-group`.

### AWS CLI

Untuk memperbarui grup atribut

`update-attribute-group` Contoh berikut memperbarui grup atribut tertentu di AWS akun Anda untuk menyertakan deskripsi.

```

aws servicecatalog-appregistry update-attribute-group \
  --attribute-group "ExampleAttributeGroup" \
  --description "This is an example attribute group"

```

Output:

```

{
  "attributeGroup": {
    "id": "01sj5xdwhbw54kejwnt09fnpc1",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-groups/01sj5xdwhbw54kejwnt09fnpc1",
    "name": "ExampleAttributeGroup",
    "description": "This is an example attribute group",
    "creationTime": "2023-02-28T20:38:01.389000+00:00",
    "lastUpdateTime": "2023-02-28T21:02:04.559000+00:00",
    "tags": {
      "aws:servicecatalog:attributeGroupName": "ExampleAttributeGroup"
    }
  }
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Mengedit grup atribut](#) di Panduan AppRegistry Administrator AWS Service Catalog.

- Untuk detail API, lihat [UpdateAttributeGroup](#) di Referensi AWS CLI Perintah.

## Athena contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with Athena.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **batch-get-named-query**

Contoh kode berikut menunjukkan cara menggunakan `batch-get-named-query`.

### AWS CLI

Untuk mengembalikan informasi tentang lebih dari satu kueri

`batch-get-named-query` Contoh berikut mengembalikan informasi tentang query bernama yang memiliki ID tertentu.

```
aws athena batch-get-named-query \
```



```
--named-query-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 a1b2c3d4-5678-90ab-cdef-EXAMPLE33333
```

### Output:

```
{
  "NamedQueries": [
    {
      "Name": "Flights Select Query",
      "Description": "Sample query to get the top 10 airports with the most
number of departures since 2000",
      "Database": "sampledb",
      "QueryString": "SELECT origin, count(*) AS total_departures\nFROM
\nflights_parquet\nWHERE year >= '2000'\nGROUP BY origin\nORDER BY total_departures
DESC\nLIMIT 10;",
      "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "WorkGroup": "primary"
    },
    {
      "Name": "Load flights table partitions",
      "Description": "Sample query to load flights table partitions using MSCK
REPAIR TABLE statement",
      "Database": "sampledb",
      "QueryString": "MSCK REPAIR TABLE flights_parquet;",
      "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "WorkGroup": "primary"
    },
    {
      "Name": "CloudFront Select Query",
      "Description": "Sample query to view requests per operating system
during a particular time frame",
      "Database": "sampledb",
      "QueryString": "SELECT os, COUNT(*) count FROM cloudfront_logs WHERE
date BETWEEN date '2014-07-05' AND date '2014-08-05' GROUP BY os;",
      "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "WorkGroup": "primary"
    }
  ],
  "UnprocessedNamedQueryIds": []
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [BatchGetNamedQuery](#) di Referensi AWS CLI Perintah.

## batch-get-query-execution

Contoh kode berikut menunjukkan cara menggunakan `batch-get-query-execution`.

### AWS CLI

Untuk mengembalikan informasi tentang satu atau lebih eksekusi kueri

`batch-get-query-execution` Contoh berikut mengembalikan informasi eksekusi query untuk query yang memiliki ID query tertentu.

```
aws athena batch-get-query-execution \  
  --query-execution-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 a1b2c3d4-5678-90ab-  
  cdef-EXAMPLE22222
```

Output:

```
{  
  "QueryExecutions": [  
    {  
      "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Query": "create database if not exists webdata",  
      "StatementType": "DDL",  
      "ResultConfiguration": {  
        "OutputLocation": "s3://awsdoc-example-bucket/a1b2c3d4-5678-90ab-  
cdef-EXAMPLE11111.txt"  
      },  
      "QueryExecutionContext": {},  
      "Status": {  
        "State": "SUCCEEDED",  
        "SubmissionDateTime": 1593470720.592,  
        "CompletionDateTime": 1593470720.902  
      },  
      "Statistics": {  
        "EngineExecutionTimeInMillis": 232,  
        "DataScannedInBytes": 0,  
        "TotalExecutionTimeInMillis": 310,  
        "ResultConfiguration": {  
          "QueryQueueTimeInMillis": 50,  
          "ServiceProcessingTimeInMillis": 28
```

```

    },
    "WorkGroup": "AthenaAdmin"
  },
  {
    "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Query": "select date, location, browser, uri, status from
cloudfront_logs where method = 'GET' and status = 200 and location like 'SF0%'
limit 10",
    "StatementType": "DML",
    "ResultConfiguration": {
      "OutputLocation": "s3://awsdoc-example-bucket/a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222.csv"
    },
    "QueryExecutionContext": {
      "Database": "mydatabase",
      "Catalog": "awsdatacatalog"
    },
    "Status": {
      "State": "SUCCEEDED",
      "SubmissionDateTime": 1593469842.665,
      "CompletionDateTime": 1593469846.486
    },
    "Statistics": {
      "EngineExecutionTimeInMillis": 3600,
      "DataScannedInBytes": 203089,
      "TotalExecutionTimeInMillis": 3821,
      "QueryQueueTimeInMillis": 267,
      "QueryPlanningTimeInMillis": 1175
    },
    "WorkGroup": "AthenaAdmin"
  }
],
"UnprocessedQueryExecutionIds": []
}

```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [BatchGetQueryExecution](#) di Referensi AWS CLI Perintah.

## create-data-catalog

Contoh kode berikut menunjukkan cara menggunakan `create-data-catalog`.

## AWS CLI

Untuk membuat katalog data

`create-data-catalog` Contoh berikut membuat katalog `dynamo_db_catalog` data.

```
aws athena create-data-catalog \  
  --name dynamo_db_catalog \  
  --type LAMBDA \  
  --description "DynamoDB Catalog" \  
  --parameters function=arn:aws:lambda:us-  
west-2:111122223333:function:dynamo_db_lambda
```

Perintah ini tidak menghasilkan output. Untuk melihat hasilnya, gunakan `aws athena get-data-catalog --name dynamo_db_catalog`.

Untuk informasi selengkapnya, lihat [Mendaftarkan Katalog: create-data-catalog](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [CreateDataCatalog](#) di Referensi AWS CLI Perintah.

## `create-named-query`

Contoh kode berikut menunjukkan cara menggunakan `create-named-query`.

## AWS CLI

Untuk membuat kueri bernama

`create-named-query` Contoh berikut membuat kueri tersimpan di AthenaAdmin workgroup yang menanyakan `flights_parquet` tabel untuk penerbangan dari Seattle ke New York pada bulan Januari 2016 yang keberangkatan dan kedatangannya ditunda lebih dari sepuluh menit. Karena nilai kode bandara dalam tabel adalah string yang menyertakan tanda kutip ganda (misalnya, "SEA"), mereka lolos oleh garis miring terbalik dan dikelilingi oleh tanda kutip tunggal.

```
aws athena create-named-query \  
  --name "SEA to JFK delayed flights Jan 2016" \  
  --description "Both arrival and departure delayed more than 10 minutes." \  
  --database sampledb \  
  --query-string "SELECT flightdate, carrier, flightnum, origin, dest,  
depdelayminutes, arrdelayminutes FROM sampledb.flights_parquet WHERE yr = 2016 AND  
month = 1 AND origin = '\"SEA\"' AND dest = '\"JFK\"' AND depdelayminutes > 10 AND  
arrdelayminutes > 10" \  
  --workgroup athenaadmin
```

```
--work-group AthenaAdmin
```

Output:

```
{
  "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [CreateNamedQuery](#) di Referensi AWS CLI Perintah.

## create-work-group

Contoh kode berikut menunjukkan cara menggunakan `create-work-group`.

### AWS CLI

Untuk membuat workgroup

`create-work-group` Contoh berikut membuat workgroup yang disebut `Data_Analyst_Group` yang memiliki lokasi `s3://awsdoc-example-bucket` output hasil query. Perintah membuat workgroup yang mengesampingkan pengaturan konfigurasi klien, yang mencakup lokasi keluaran hasil kueri. Perintah ini juga memungkinkan CloudWatch metrik dan menambahkan tiga pasangan tag nilai kunci ke workgroup untuk membedakannya dari kelompok kerja lainnya. Perhatikan bahwa `--configuration` argumen tidak memiliki spasi sebelum koma yang memisahkan opsinya.

```
aws athena create-work-group \
  --name Data_Analyst_Group \
  --configuration ResultConfiguration={OutputLocation="s3://awsdoc-example-
bucket"},EnforceWorkGroupConfiguration="true",PublishCloudWatchMetricsEnabled="true" \
  --description "Workgroup for data analysts" \
  --tags Key=Division,Value=West Key=Location,Value=Seattle Key=Team,Value="Big
Data"
```

Perintah ini tidak menghasilkan output. Untuk melihat hasilnya, gunakan `aws athena get-work-group --work-group Data_Analyst_Group`.

Untuk informasi selengkapnya, lihat [Mengelola Grup Kerja](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [CreateWorkGroup](#) di Referensi AWS CLI Perintah.

## **delete-data-catalog**

Contoh kode berikut menunjukkan cara menggunakan `delete-data-catalog`.

### AWS CLI

Untuk menghapus katalog data

`delete-data-catalog` Contoh berikut menghapus katalog `UnusedDataCatalog` data.

```
aws athena delete-data-catalog \  
  --name UnusedDataCatalog
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Katalog: delete-data-catalog](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [DeleteDataCatalog](#) di Referensi AWS CLI Perintah.

## **delete-named-query**

Contoh kode berikut menunjukkan cara menggunakan `delete-named-query`.

### AWS CLI

Untuk menghapus kueri bernama

`delete-named-query` Contoh berikut menghapus query bernama yang memiliki ID tertentu.

```
aws athena delete-named-query \  
  --named-query-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [DeleteNamedQuery](#) di Referensi AWS CLI Perintah.

## delete-work-group

Contoh kode berikut menunjukkan cara menggunakan `delete-work-group`.

### AWS CLI

Untuk menghapus workgroup

`delete-work-group` Contoh berikut menghapus TeamB workgroup.

```
aws athena delete-work-group \  
  --work-group TeamB
```

Perintah ini tidak menghasilkan output. Untuk mengkonfirmasi penghapusan, gunakan `aws athena list-work-groups`

Untuk informasi selengkapnya, lihat [Mengelola Grup Kerja](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [DeleteWorkGroup](#) di Referensi AWS CLI Perintah.

## get-data-catalog

Contoh kode berikut menunjukkan cara menggunakan `get-data-catalog`.

### AWS CLI

Untuk mengembalikan informasi tentang katalog data

`get-data-catalog` Contoh berikut mengembalikan informasi tentang katalog `dynamo_db_catalog` data.

```
aws athena get-data-catalog \  
  --name dynamo_db_catalog
```

Output:

```
{  
  "DataCatalog": {
```

```

    "Name": "dynamo_db_catalog",
    "Description": "DynamoDB Catalog",
    "Type": "LAMBDA",
    "Parameters": {
      "catalog": "dynamo_db_catalog",
      "metadata-function": "arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda",
      "record-function": "arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Menampilkan Detail Katalog: get-data-catalog](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetDataCatalog](#) di Referensi AWS CLI Perintah.

## get-database

Contoh kode berikut menunjukkan cara menggunakan `get-database`.

### AWS CLI

Untuk mengembalikan informasi tentang database dalam katalog data

`get-database` Contoh berikut mengembalikan informasi tentang `sampledb` database dalam katalog `AwsDataCatalog` data.

```

aws athena get-database \
  --catalog-name AwsDataCatalog \
  --database-name sampledb

```

Output:

```

{
  "Database": {
    "Name": "sampledb",
    "Description": "Sample database",
    "Parameters": {
      "CreatedBy": "Athena",
      "EXTERNAL": "TRUE"
    }
  }
}

```



```
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menampilkan Detail Database: get-database](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetDatabase](#) di Referensi AWS CLI Perintah.

## get-named-query

Contoh kode berikut menunjukkan cara menggunakan `get-named-query`.

### AWS CLI

Untuk mengembalikan kueri bernama

`get-named-query` Contoh berikut mengembalikan informasi tentang query yang memiliki ID tertentu.

```
aws athena get-named-query \  
  --named-query-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "NamedQuery": {  
    "Name": "CloudFront Logs - SF0",  
    "Description": "Shows successful GET request data for SF0",  
    "Database": "default",  
    "QueryString": "select date, location, browser, uri, status from  
cloudfront_logs where method = 'GET' and status = 200 and location like 'SF0%'  
limit 10",  
    "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "WorkGroup": "AthenaAdmin"  
  }  
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetNamedQuery](#) di Referensi AWS CLI Perintah.

## get-query-execution

Contoh kode berikut menunjukkan cara menggunakan `get-query-execution`.

### AWS CLI

Untuk mengembalikan informasi tentang eksekusi kueri

`get-query-execution` Contoh berikut mengembalikan informasi tentang query yang memiliki ID query tertentu.

```
aws athena get-query-execution \  
--query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "QueryExecution": {  
    "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "Query": "select date, location, browser, uri, status from cloudfront_logs  
where method = 'GET  
' and status = 200 and location like 'SF0%' limit 10",  
    "StatementType": "DML",  
    "ResultConfiguration": {  
      "OutputLocation": "s3://awsdoc-example-bucket/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111.csv"  
    },  
    "QueryExecutionContext": {  
      "Database": "mydatabase",  
      "Catalog": "awsdatacatalog"  
    },  
    "Status": {  
      "State": "SUCCEEDED",  
      "SubmissionDateTime": 1593469842.665,  
      "CompletionDateTime": 1593469846.486  
    },  
    "Statistics": {  
      "EngineExecutionTimeInMillis": 3600,  
      "DataScannedInBytes": 203089,  
      "TotalExecutionTimeInMillis": 3821,  
      "QueryQueueTimeInMillis": 267,  
      "QueryPlanningTimeInMillis": 1175  
    },  
  },  
}
```

```
    "WorkGroup": "AthenaAdmin"
  }
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetQueryExecution](#) di Referensi AWS CLI Perintah.

## get-query-results

Contoh kode berikut menunjukkan cara menggunakan `get-query-results`.

### AWS CLI

Untuk mengembalikan hasil kueri

`get-query-results` Contoh berikut mengembalikan hasil query yang memiliki ID query tertentu.

```
aws athena get-query-results \
  --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
  "ResultSet": {
    "Rows": [
      {
        "Data": [
          {
            "VarCharValue": "date"
          },
          {
            "VarCharValue": "location"
          },
          {
            "VarCharValue": "browser"
          },
          {
            "VarCharValue": "uri"
          },
          {
            "VarCharValue": "status"
          }
        ]
      }
    ]
  }
}
```

```
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Safari"
    },
    {
      "VarCharValue": "/test-image-2.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Opera"
    },
    {
      "VarCharValue": "/test-image-2.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
```

```
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Firefox"
      },
      {
        "VarCharValue": "/test-image-3.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Lynx"
      },
      {
        "VarCharValue": "/test-image-3.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
```

```
        "VarCharValue": "IE"
      },
      {
        "VarCharValue": "/test-image-2.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Opera"
      },
      {
        "VarCharValue": "/test-image-1.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Chrome"
      },
      {
        "VarCharValue": "/test-image-3.jpeg"
      }
    ]
  }
}
```

```
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Firefox"
      },
      {
        "VarCharValue": "/test-image-2.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Chrome"
      },
      {
        "VarCharValue": "/test-image-3.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
```

```
        {
          "VarCharValue": "2014-07-05"
        },
        {
          "VarCharValue": "SF04"
        },
        {
          "VarCharValue": "IE"
        },
        {
          "VarCharValue": "/test-image-2.jpeg"
        },
        {
          "VarCharValue": "200"
        }
      ]
    }
  ],
  "ResultSetMetadata": {
    "ColumnInfo": [
      {
        "CatalogName": "hive",
        "SchemaName": "",
        "TableName": "",
        "Name": "date",
        "Label": "date",
        "Type": "date",
        "Precision": 0,
        "Scale": 0,
        "Nullable": "UNKNOWN",
        "CaseSensitive": false
      },
      {
        "CatalogName": "hive",
        "SchemaName": "",
        "TableName": "",
        "Name": "location",
        "Label": "location",
        "Type": "varchar",
        "Precision": 2147483647,
        "Data": [
          "Scale": 0,
          "Nullable": "UNKNOWN",
```



```
        "CaseSensitive": true
    },
    {
        "CatalogName": "hive",
        "SchemaName": "",
        "TableName": "",
        "Name": "browser",
        "Label": "browser",
        "Type": "varchar",
        "Precision": 2147483647,
        "Scale": 0,
        "Nullable": "UNKNOWN",
        "CaseSensitive": true
    },
    {
        "CatalogName": "hive",
        "SchemaName": "",
        "TableName": "",
        "Name": "uri",
        "Label": "uri",
        "Type": "varchar",
        "Precision": 2147483647,
        "Scale": 0,
        "Nullable": "UNKNOWN",
        "CaseSensitive": true
    },
    {
        "CatalogName": "hive",
        "SchemaName": "",
        "TableName": "",
        "Name": "status",
        "Label": "status",
        "Type": "integer",
        "Precision": 10,
        "Scale": 0,
        "Nullable": "UNKNOWN",
        "CaseSensitive": false
    }
]
}
},
"UpdateCount": 0
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Hasil Kueri, File Output, dan Riwayat Kueri](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetQueryResults](#) di Referensi AWS CLI Perintah.

## get-table-metadata

Contoh kode berikut menunjukkan cara menggunakan `get-table-metadata`.

### AWS CLI

Untuk mengembalikan informasi metadata tentang tabel

`get-table-metadata` Contoh berikut mengembalikan informasi metadata tentang `counties` tabel, termasuk termasuk nama kolom dan tipe datanya, dari `sampledb` database katalog data. `AwsDataCatalog`

```
aws athena get-table-metadata \  
  --catalog-name AwsDataCatalog \  
  --database-name sampledb \  
  --table-name counties
```

Output:

```
{  
  "TableMetadata": {  
    "Name": "counties",  
    "CreateTime": 1593559968.0,  
    "LastAccessTime": 0.0,  
    "TableType": "EXTERNAL_TABLE",  
    "Columns": [  
      {  
        "Name": "name",  
        "Type": "string",  
        "Comment": "from deserializer"  
      },  
      {  
        "Name": "boundaryshape",  
        "Type": "binary",  
        "Comment": "from deserializer"  
      },  
      {
```

```

        "Name": "motto",
        "Type": "string",
        "Comment": "from deserializer"
    },
    {
        "Name": "population",
        "Type": "int",
        "Comment": "from deserializer"
    }
],
"PartitionKeys": [],
"Parameters": {
    "EXTERNAL": "TRUE",
    "inputformat": "com.esri.json.hadoop.EnclosedJsonInputFormat",
    "location": "s3://awsdoc-example-bucket/json",
    "outputformat":
"org.apache.hadoop.hive.q1.io.HiveIgnoreKeyTextOutputFormat",
    "serde.param.serialization.format": "1",
    "serde.serialization.lib": "com.esri.hadoop.hive.serde.JsonSerde",
    "transient_lastDdlTime": "1593559968"
}
}
}

```

Untuk informasi selengkapnya, lihat [Menampilkan Detail Tabel: `get-table-metadata`](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetTableMetadata](#) di Referensi AWS CLI Perintah.

## get-work-group

Contoh kode berikut menunjukkan cara menggunakan `get-work-group`.

### AWS CLI

Untuk mengembalikan informasi tentang workgroup

`get-work-group` Contoh berikut mengembalikan informasi tentang AthenaAdmin workgroup.

```
aws athena get-work-group \
  --work-group AthenaAdmin
```

Output:

```
{
  "WorkGroup": {
    "Name": "AthenaAdmin",
    "State": "ENABLED",
    "Configuration": {
      "ResultConfiguration": {
        "OutputLocation": "s3://awsdoc-example-bucket/"
      },
      "EnforceWorkGroupConfiguration": false,
      "PublishCloudWatchMetricsEnabled": true,
      "RequesterPaysEnabled": false
    },
    "Description": "Workgroup for Athena administrators",
    "CreationTime": 1573677174.105
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola Grup Kerja](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [GetWorkGroup](#) di Referensi AWS CLI Perintah.

## list-data-catalogs

Contoh kode berikut menunjukkan cara menggunakan `list-data-catalogs`.

### AWS CLI

Untuk membuat daftar katalog data yang terdaftar di Athena

`list-data-catalogs` Contoh berikut mencantumkan katalog data yang terdaftar di Athena.

```
aws athena list-data-catalogs
```

Output:

```
{
  "DataCatalogsSummary": [
    {
      "CatalogName": "AwsDataCatalog",
      "Type": "GLUE"
    },
  ],
}
```

```
{
  "CatalogName": "cw_logs_catalog",
  "Type": "LAMBDA"
},
{
  "CatalogName": "cw_metrics_catalog",
  "Type": "LAMBDA"
}
]
```

Untuk informasi selengkapnya, lihat [Daftar Katalog Terdaftar: list-data-catalogs](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [ListDataCatalogs](#) di Referensi AWS CLI Perintah.

## list-databases

Contoh kode berikut menunjukkan cara menggunakan `list-databases`.

### AWS CLI

Untuk membuat daftar database dalam katalog data

`list-databases` Contoh berikut mencantumkan database dalam katalog `AwsDataCatalog` data.

```
aws athena list-databases \
  --catalog-name AwsDataCatalog
```

Output:

```
{
  "DatabaseList": [
    {
      "Name": "default"
    },
    {
      "Name": "mydatabase"
    },
    {
      "Name": "newdb"
    }
  ],
}
```

```
{
  "Name": "sampledb",
  "Description": "Sample database",
  "Parameters": {
    "CreatedBy": "Athena",
    "EXTERNAL": "TRUE"
  }
},
{
  "Name": "webdata"
}
]
```

Untuk informasi selengkapnya, lihat [Daftar Database dalam Katalog: daftar-database di Panduan Pengguna Amazon Athena](#).

- Untuk detail API, lihat [ListDatabases](#) di Referensi AWS CLI Perintah.

## list-named-queries

Contoh kode berikut menunjukkan cara menggunakan `list-named-queries`.

### AWS CLI

Untuk mencantumkan kueri bernama untuk grup kerja

`list-named-queries` Contoh berikut mencantumkan kueri bernama untuk AthenaAdmin workgroup.

```
aws athena list-named-queries \
  --work-group AthenaAdmin
```

Output:

```
{
  "NamedQueryIds": [
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"
  ]
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [ListNamedQueries](#) di Referensi AWS CLI Perintah.

## list-query-executions

Contoh kode berikut menunjukkan cara menggunakan `list-query-executions`.

### AWS CLI

Untuk mencantumkan ID kueri dari kueri dalam workgroup tertentu

`list-query-executions` Contoh berikut mencantumkan maksimal sepuluh ID kueri di AthenaAdmin workgroup.

```
aws athena list-query-executions \
  --work-group AthenaAdmin \
  --max-items 10
```

### Output:

```
{
  "QueryExecutionIds": [
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11110",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11114",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11115",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11116",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11117",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11118",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11119"
  ],
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0="
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Hasil Kueri, File Output, dan Riwayat Kueri](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [ListQueryExecutions](#) di Referensi AWS CLI Perintah.

## list-table-metadata

Contoh kode berikut menunjukkan cara menggunakan `list-table-metadata`.

### AWS CLI

Untuk daftar metadata untuk tabel dalam database tertentu dari katalog data

`list-table-metadata` Contoh berikut mengembalikan informasi metadata untuk maksimal dua tabel dalam geography database katalog `AwsDataCatalog` data.

```
aws athena list-table-metadata \  
  --catalog-name AwsDataCatalog \  
  --database-name geography \  
  --max-items 2
```

### Output:

```
{  
  "TableMetadataList": [  
    {  
      "Name": "country_codes",  
      "CreateTime": 1586553454.0,  
      "TableType": "EXTERNAL_TABLE",  
      "Columns": [  
        {  
          "Name": "country",  
          "Type": "string",  
          "Comment": "geo id"  
        },  
        {  
          "Name": "alpha-2 code",  
          "Type": "string",  
          "Comment": "geo id2"  
        },  
        {  
          "Name": "alpha-3 code",  
          "Type": "string",  
          "Comment": "state name"  
        },  
        {  
          "Name": "numeric code",  
          "Type": "bigint",
```



```

        "Comment": ""
    },
    {
        "Name": "latitude",
        "Type": "bigint",
        "Comment": "location (latitude)"
    },
    {
        "Name": "longitude",
        "Type": "bigint",
        "Comment": "location (longitude)"
    }
],
"Parameters": {
    "areColumnsQuoted": "false",
    "classification": "csv",
    "columnsOrdered": "true",
    "delimiter": ",",
    "has_encrypted_data": "false",
    "inputformat": "org.apache.hadoop.mapred.TextInputFormat",
    "location": "s3://awsdoc-example-bucket/csv/countrycode",
    "outputformat":
"org.apache.hadoop.hive ql.io.HiveIgnoreKeyTextOutputFormat",
    "serde.param.field.delim": ",",
    "serde.serialization.lib":
"org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe",
    "skip.header.line.count": "1",
    "typeOfData": "file"
}
},
{
    "Name": "county_populations",
    "CreateTime": 1586553446.0,
    "TableType": "EXTERNAL_TABLE",
    "Columns": [
        {
            "Name": "id",
            "Type": "string",
            "Comment": "geo id"
        },
        {
            "Name": "country",
            "Type": "string",
            "Comment": "country"
        },
        {
            "Name": "id2",
            "Type": "string",
            "Comment": "geo id"
        }
    ]
}
]
}
}

```

```

        "Type": "string",
        "Comment": "geo id2"
    },
    {
        "Name": "county",
        "Type": "string",
        "Comment": "county name"
    },
    {
        "Name": "state",
        "Type": "string",
        "Comment": "state name"
    },
    {
        "Name": "population estimate 2018",
        "Type": "string",
        "Comment": ""
    }
],
"Parameters": {
    "areColumnsQuoted": "false",
    "classification": "csv",
    "columnsOrdered": "true",
    "delimiter": ",",
    "has_encrypted_data": "false",
    "inputformat": "org.apache.hadoop.mapred.TextInputFormat",
    "location": "s3://awsdoc-example-bucket/csv/CountyPopulation",
    "outputformat":
"org.apache.hadoop.hive ql.io.HiveIgnoreKeyTextOutputFormat",
    "serde.param.field.delim": ",",
    "serde.serialization.lib":
"org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe",
    "skip.header.line.count": "1",
    "typeOfData": "file"
}
},
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

Untuk informasi selengkapnya, lihat [Menampilkan Metadata untuk Semua Tabel dalam Database: list-table-metadata](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [ListTableMetadata](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Contoh 1: Untuk daftar tag untuk workgroup

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk `Data_Analyst_Group` workgroup.

```
aws athena list-tags-for-resource \  
  --resource-arn arn:aws:athena:us-west-2:111122223333:workgroup/  
Data_Analyst_Group
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Division",  
      "Value": "West"  
    },  
    {  
      "Key": "Team",  
      "Value": "Big Data"  
    },  
    {  
      "Key": "Location",  
      "Value": "Seattle"  
    }  
  ]  
}
```

Contoh 2: Untuk daftar tag untuk katalog data

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk katalog `dynamo_db_catalog` data.

```
aws athena list-tags-for-resource \  
  --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/  
dynamo_db_catalog
```

**Output:**

```
{
  "Tags": [
    {
      "Key": "Division",
      "Value": "Mountain"
    },
    {
      "Key": "Organization",
      "Value": "Retail"
    },
    {
      "Key": "Product_Line",
      "Value": "Shoes"
    },
    {
      "Key": "Location",
      "Value": "Denver"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mencantumkan tag untuk sumber daya: list-tags-for-resource](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

**list-work-groups**

Contoh kode berikut menunjukkan cara menggunakan `list-work-groups`.

**AWS CLI**

Untuk daftar kelompok kerja

`list-work-groups` Contoh berikut mencantumkan workgroup di akun saat ini.

```
aws athena list-work-groups
```

**Output:**

```
{
```

```

    "WorkGroups": [
      {
        "Name": "Data_Analyst_Group",
        "State": "ENABLED",
        "Description": "",
        "CreationTime": 1578006683.016
      },
      {
        "Name": "AthenaAdmin",
        "State": "ENABLED",
        "Description": "",
        "CreationTime": 1573677174.105
      },
      {
        "Name": "primary",
        "State": "ENABLED",
        "Description": "",
        "CreationTime": 1567465222.723
      }
    ]
  }

```

Untuk informasi selengkapnya, lihat [Mengelola Grup Kerja](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [ListWorkGroups](#) di Referensi AWS CLI Perintah.

## start-query-execution

Contoh kode berikut menunjukkan cara menggunakan `start-query-execution`.

### AWS CLI

Contoh 1: Untuk menjalankan kueri dalam workgroup pada tabel yang ditentukan dalam database dan katalog data yang ditentukan

`start-query-execution` Contoh berikut menggunakan `AthenaAdmin` workgroup untuk menjalankan query pada `cloudfront_logs` tabel di `cflogsdatabase` dalam katalog `AwsDataCatalog` data.

```

aws athena start-query-execution \
  --query-string "select date, location, browser, uri, status from cloudfront_logs
  where method = 'GET' and status = 200 and location like 'SF0%' limit 10" \

```

```
--work-group "AthenaAdmin" \  
--query-execution-context Database=cflogsdatabase,Catalog=AwsDataCatalog
```

Output:

```
{  
  "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

Contoh 2: Untuk menjalankan kueri yang menggunakan workgroup tertentu untuk membuat database dalam katalog data tertentu

`start-query-execution` Contoh berikut menggunakan AthenaAdmin workgroup untuk membuat database newdb dalam katalog AwsDataCatalog data default.

```
aws athena start-query-execution \  
  --query-string "create database if not exists newdb" \  
  --work-group "AthenaAdmin"
```

Output:

```
{  
  "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11112"  
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

Contoh 3: Untuk menjalankan kueri yang membuat tampilan pada tabel dalam database dan katalog data yang ditentukan

`start-query-execution` Contoh berikut menggunakan SELECT pernyataan pada `cloudfront_logs` tabel di `cflogsdatabase` untuk membuat tampilancf10.

```
aws athena start-query-execution \  
  --query-string "CREATE OR REPLACE VIEW cf10 AS SELECT * FROM cloudfront_logs  
  limit 10" \  
  --work-group "AthenaAdmin"
```

```
--query-execution-context Database=cflogsdatabase
```

Output:

```
{
  "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11113"
}
```

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [StartQueryExecution](#) di Referensi AWS CLI Perintah.

## stop-query-execution

Contoh kode berikut menunjukkan cara menggunakan `stop-query-execution`.

AWS CLI

Untuk menghentikan kueri yang sedang berjalan

`stop-query-execution` Contoh berikut menghentikan query yang memiliki ID query tertentu.

```
aws athena stop-query-execution \  
  --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menjalankan Kueri SQL Menggunakan Amazon Athena](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [StopQueryExecution](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

AWS CLI

Untuk menambahkan tanda ke sumber daya

`tag-resource` Contoh berikut menambahkan tiga tag ke katalog `dynamo_db_catalog` data.

```
aws athena tag-resource \  
  --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/  
dynamo_db_catalog \  
  --tags Key=Organization,Value=Retail Key=Division,Value=Mountain  
Key=Product_Line,Value=Shoes Key=Location,Value=Denver
```

Perintah ini tidak menghasilkan output. Untuk melihat hasilnya, gunakan `aws athena list-tags-for-resource --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/dynamo_db_catalog`.

Untuk informasi selengkapnya, lihat [Menambahkan tag ke sumber daya: tag-resource](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus Focus kunci Specialization dan nilai yang terkait dari sumber daya katalog `dynamo_db_catalog` data.

```
aws athena untag-resource \  
  --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/  
dynamo_db_catalog \  
  --tag-keys Specialization Focus
```

Perintah ini tidak menghasilkan output. Untuk melihat hasilnya, gunakan `list-tags-for-resource` perintah.

Untuk informasi selengkapnya, lihat [Menghapus tag dari sumber daya: untag-resource di Panduan](#) Pengguna Amazon Athena.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.



## update-data-catalog

Contoh kode berikut menunjukkan cara menggunakan `update-data-catalog`.

### AWS CLI

Untuk memperbarui katalog data

`update-data-catalog` Contoh berikut memperbarui fungsi Lambda dan deskripsi katalog `cw_logs_catalog` data.

```
aws athena update-data-catalog \  
  --name cw_logs_catalog \  
  --type LAMBDA \  
  --description "New CloudWatch Logs Catalog" \  
  --function=arn:aws:lambda:us-west-2:111122223333:function:new_cw_logs_lambda
```

Perintah ini tidak menghasilkan output. Untuk melihat hasilnya, gunakan `aws athena get-data-catalog --name cw_logs_catalog`.

Untuk informasi selengkapnya, lihat [Memperbarui Katalog: update-data-catalog](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [UpdateDataCatalog](#) di Referensi AWS CLI Perintah.

## update-work-group

Contoh kode berikut menunjukkan cara menggunakan `update-work-group`.

### AWS CLI

Untuk memperbarui workgroup

`update-work-group` Contoh berikut menonaktifkan `Data_Analyst_Group` workgroup. Pengguna tidak dapat menjalankan atau membuat kueri di grup kerja yang dinonaktifkan, tetapi masih dapat melihat metrik, kontrol batas penggunaan data, pengaturan grup kerja, riwayat kueri, dan kueri yang disimpan.

```
aws athena update-work-group \  
  --work-group Data_Analyst_Group \  
  --state DISABLED
```

Perintah ini tidak menghasilkan output. Untuk memverifikasi perubahan status, gunakan `aws athena get-work-group --work-group Data_Analyst_Group` dan periksa State properti di output.

Untuk informasi selengkapnya, lihat [Mengelola Grup Kerja](#) di Panduan Pengguna Amazon Athena.

- Untuk detail API, lihat [UpdateWorkGroup](#) di Referensi AWS CLI Perintah.

## Contoh Auto Scaling menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Auto Scaling AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **attach-instances**

Contoh kode berikut menunjukkan cara menggunakan `attach-instances`.

AWS CLI

Untuk melampirkan instance ke grup Auto Scaling

Contoh ini melampirkan instance yang ditentukan ke grup Auto Scaling yang ditentukan.

```
aws autoscaling attach-instances \
```

```
--instance-ids i-061c63c5eb45f0416 \  
--auto-scaling-group-name my-asg
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AttachInstances](#) di Referensi AWS CLI Perintah.

## **attach-load-balancer-target-groups**

Contoh kode berikut menunjukkan cara menggunakan `attach-load-balancer-target-groups`.

### AWS CLI

Untuk melampirkan grup target ke grup Auto Scaling

Contoh ini melampirkan grup target yang ditentukan ke grup Auto Scaling yang ditentukan.

```
aws autoscaling attach-load-balancer-target-groups \  
  --auto-scaling-group-name my-asg \  
  --target-group-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Elastic Load Balancing dan Amazon EC2 Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [AttachLoadBalancerTargetGroups](#) di Referensi AWS CLI Perintah.

## **attach-load-balancers**

Contoh kode berikut menunjukkan cara menggunakan `attach-load-balancers`.

### AWS CLI

Untuk melampirkan Classic Load Balancer ke grup Auto Scaling

Contoh ini melampirkan Classic Load Balancer yang ditentukan ke grup Auto Scaling yang ditentukan.

```
aws autoscaling attach-load-balancers \  
  --load-balancer-names my-load-balancer \  
  --target-group-arns arn:aws:elasticloadbalancing:us-
```

```
--auto-scaling-group-name my-asg
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Elastic Load Balancing dan Amazon EC2 Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [AttachLoadBalancers](#) di Referensi AWS CLI Perintah.

## cancel-instance-refresh

Contoh kode berikut menunjukkan cara menggunakan `cancel-instance-refresh`.

### AWS CLI

Untuk membatalkan penyegaran instans

`cancel-instance-refresh` Contoh berikut membatalkan penyegaran instans yang sedang berlangsung untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling cancel-instance-refresh \  
  --auto-scaling-group-name my-asg
```

Output:

```
{  
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"  
}
```

Untuk informasi selengkapnya, lihat [Membatalkan penyegaran instans](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [CancelInstanceRefresh](#) di Referensi AWS CLI Perintah.

## complete-lifecycle-action

Contoh kode berikut menunjukkan cara menggunakan `complete-lifecycle-action`.

### AWS CLI

Untuk menyelesaikan tindakan siklus hidup

Contoh ini memberi tahu Amazon EC2 Auto Scaling bahwa tindakan siklus hidup yang ditentukan telah selesai sehingga dapat menyelesaikan peluncuran atau penghentian instance.

```
aws autoscaling complete-lifecycle-action \  
  --lifecycle-hook-name my-launch-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-action-result CONTINUE \  
  --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [CompleteLifecycleAction](#) di Referensi AWS CLI Perintah.

## create-auto-scaling-group

Contoh kode berikut menunjukkan cara menggunakan `create-auto-scaling-group`.

### AWS CLI

Contoh 1: Untuk membuat grup Auto Scaling

`create-auto-scaling-group` Contoh berikut membuat grup Auto Scaling di subnet di beberapa Availability Zone dalam suatu Region. Instans diluncurkan dengan versi default dari template peluncuran yang ditentukan. Perhatikan bahwa default digunakan untuk sebagian besar pengaturan lain, seperti kebijakan penghentian dan konfigurasi pemeriksaan kesehatan.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12 \  
  --min-size 1 \  
  --max-size 5 \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [grup Auto Scaling di Panduan Pengguna Auto Scaling Amazon EC2](#).

## Contoh 2: Untuk melampirkan Application Load Balancer, Network Load Balancer, atau Gateway Load Balancer

Contoh ini menentukan ARN dari kelompok target untuk penyeimbang beban yang mendukung lalu lintas yang diharapkan. Jenis pemeriksaan kesehatan menentukan ELB sehingga ketika Elastic Load Balancing melaporkan sebuah instance sebagai tidak sehat, grup Auto Scaling menggantikannya. Perintah ini juga mendefinisikan masa tenggang pemeriksaan kesehatan 600 detik. Masa tenggang membantu mencegah penghentian prematur dari instans yang baru diluncurkan.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12 \  
  --target-group-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/943f017f100becff \  
  --health-check-type ELB \  
  --health-check-grace-period 600 \  
  --min-size 1 \  
  --max-size 5 \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Elastic Load Balancing dan Amazon EC2 Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

## Contoh 3: Untuk menentukan grup penempatan dan menggunakan versi terbaru dari template peluncuran

Contoh ini meluncurkan instance ke grup penempatan dalam Availability Zone tunggal. Ini dapat berguna untuk grup latensi rendah dengan beban kerja HPC. Contoh ini juga menentukan ukuran minimum, ukuran maksimum, dan kapasitas grup yang diinginkan.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12,Version='$Latest' \  
  --min-size 1 \  
  --max-size 5 \  
  --desired-capacity 3 \  
  --placement-group my-placement-group \  
  --vpc-zone-identifier "subnet-6194ea3b"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Grup penempatan](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Contoh 4: Untuk menentukan satu contoh grup Auto Scaling dan menggunakan versi spesifik dari template peluncuran

Contoh ini membuat grup Auto Scaling dengan kapasitas minimum dan maksimum yang disetel 1 untuk menegaskan bahwa satu instance akan berjalan. Perintah ini juga menentukan v1 dari template peluncuran di mana ID ENI yang ada ditentukan. Bila Anda menggunakan template peluncuran yang menentukan ENI yang ada untuk eth0, Anda harus menentukan Availability Zone untuk grup Auto Scaling yang cocok dengan antarmuka jaringan, tanpa juga menentukan ID subnet dalam permintaan.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg-single-instance \  
  --launch-template LaunchTemplateName=my-template-for-auto-scaling,Version='1' \  
  --min-size 1 \  
  --max-size 1 \  
  --availability-zones us-west-2a
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [grup Auto Scaling di Panduan](#) Pengguna Auto Scaling Amazon EC2.

Contoh 5: Untuk menentukan kebijakan penghentian yang berbeda

Contoh ini membuat grup Auto Scaling menggunakan konfigurasi peluncuran dan menetapkan kebijakan penghentian untuk menghentikan instance tertua terlebih dahulu. Perintah ini juga menerapkan tag ke grup dan instance-nya, dengan kunci dari Role dan nilai. WebServer

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-configuration-name my-lc \  
  --min-size 1 \  
  --max-size 5 \  
  --termination-policies "OldestInstance" \  
  --tags "ResourceId=my-asg,ResourceType=auto-scaling-  
group,Key=Role,Value=WebServer,PropagateAtLaunch=true" \  
  \
```

```
--vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan kebijakan penghentian Penskalaan Otomatis Amazon EC2 di Panduan Pengguna Auto Scaling Amazon EC2](#).

Contoh 6: Untuk menentukan hook siklus hidup peluncuran

Contoh ini membuat grup Auto Scaling dengan kait siklus hidup yang mendukung tindakan kustom saat peluncuran instance.

```
aws autoscaling create-auto-scaling-group \  
  --cli-input-json file://~/config.json
```

Isi config.json file:

```
{  
  "AutoScalingGroupName": "my-asg",  
  "LaunchTemplate": {  
    "LaunchTemplateId": "lt-1234567890abcde12"  
  },  
  "LifecycleHookSpecificationList": [{  
    "LifecycleHookName": "my-launch-hook",  
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING",  
    "NotificationTargetARN": "arn:aws:sqs:us-west-2:123456789012:my-sqs-queue",  
    "RoleARN": "arn:aws:iam::123456789012:role/my-notification-role",  
    "NotificationMetadata": "SQS message metadata",  
    "HeartbeatTimeout": 4800,  
    "DefaultResult": "ABANDON"  
  }],  
  "MinSize": 1,  
  "MaxSize": 5,  
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",  
  "Tags": [{  
    "ResourceType": "auto-scaling-group",  
    "ResourceId": "my-asg",  
    "PropagateAtLaunch": true,  
    "Value": "test",  
    "Key": "environment"  
  }]  
}
```



Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 7: Untuk menentukan hook siklus hidup penghentian

Contoh ini membuat grup Auto Scaling dengan pengait siklus hidup yang mendukung tindakan kustom saat penghentian instance.

```
aws autoscaling create-auto-scaling-group \  
  --cli-input-json file://~/config.json
```

Isi dari config.json:

```
{  
  "AutoScalingGroupName": "my-asg",  
  "LaunchTemplate": {  
    "LaunchTemplateId": "lt-1234567890abcde12"  
  },  
  "LifecycleHookSpecificationList": [{  
    "LifecycleHookName": "my-termination-hook",  
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING",  
    "HeartbeatTimeout": 120,  
    "DefaultResult": "CONTINUE"  
  }],  
  "MinSize": 1,  
  "MaxSize": 5,  
  "TargetGroupARNs": [  
    "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-  
targets/73e2d6bc24d8a067"  
  ],  
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 8: Untuk menentukan kebijakan penghentian kustom

Contoh ini membuat grup Auto Scaling yang menentukan kebijakan penghentian fungsi Lambda kustom yang memberi tahu Auto Scaling Amazon EC2 instans mana yang aman untuk dihentikan pada skala.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg-single-instance \  
  --launch-template LaunchTemplateName=my-template-for-auto-scaling \  
  --min-size 1 \  
  --max-size 5 \  
  --termination-policies "arn:aws:lambda:us-  
west-2:123456789012:function:HelloFunction:prod" \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat kebijakan penghentian kustom dengan Lambda di Panduan](#) Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [CreateAutoScalingGroup](#) di Referensi AWS CLI Perintah.

## create-launch-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-launch-configuration`.

### AWS CLI

Contoh 1: Untuk membuat konfigurasi peluncuran

Contoh ini membuat konfigurasi peluncuran sederhana.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat konfigurasi peluncuran](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk membuat konfigurasi peluncuran dengan grup keamanan, key pair, dan skrip bootstrapping

Contoh ini membuat konfigurasi peluncuran dengan grup keamanan, key pair, dan skrip bootstrapping yang terdapat dalam data pengguna.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --security-groups sg-eb2af88example \  
  --key-name my-key-pair \  
  --user-data file://myuserdata.txt
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat konfigurasi peluncuran](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 3: Untuk membuat konfigurasi peluncuran dengan peran IAM

Contoh ini membuat konfigurasi peluncuran dengan nama profil instance peran IAM.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --iam-instance-profile my-autoscaling-role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [IAM role untuk aplikasi yang berjalan di Instans Amazon EC2](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 4: Untuk membuat konfigurasi peluncuran dengan pemantauan terperinci diaktifkan

Contoh ini membuat konfigurasi peluncuran dengan pemantauan terperinci EC2 diaktifkan, yang mengirimkan metrik EC2 CloudWatch dalam periode 1 menit.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --instance-monitoring Enabled=true
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi pemantauan untuk instans Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 5: Untuk membuat konfigurasi peluncuran yang meluncurkan Instans Spot

Contoh ini membuat konfigurasi peluncuran yang menggunakan Instans Spot sebagai satu-satunya opsi pembelian.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --spot-price "0.50"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Meminta Instans Spot](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 6: Untuk membuat konfigurasi peluncuran menggunakan instans EC2

Contoh ini membuat konfigurasi peluncuran berdasarkan atribut dari instance yang ada. Ini mengesampingkan penyewaan penempatan dan apakah alamat IP publik diatur dengan menyertakan opsi `dan--placement-tenancy`. `--no-associate-public-ip-address`

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc-from-instance \  
  --instance-id i-0123a456700123456 \  
  --instance-type m5.large \  
  --no-associate-public-ip-address \  
  --placement-tenancy dedicated
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat konfigurasi peluncuran menggunakan instans EC2 di Panduan](#) Pengguna Auto Scaling Amazon EC2.

Contoh 7: Untuk membuat konfigurasi peluncuran dengan pemetaan perangkat blok untuk volume Amazon EBS

Contoh ini membuat konfigurasi peluncuran dengan pemetaan perangkat blok untuk gp3 volume Amazon EBS dengan nama perangkat /dev/sdh dan ukuran volume 20.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --block-device-mappings '[{"DeviceName":"/dev/sdh","Ebs":  
{"VolumeSize":20,"VolumeType":"gp3"}}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [EBS di Referensi API Auto Scaling Amazon EC2](#).

Untuk informasi tentang sintaks untuk mengutip nilai parameter berformat JSON, lihat [Menggunakan tanda kutip dengan string di CLI di Panduan Pengguna Antarmuka Baris Perintah AWS.AWS](#)

Contoh 8: Untuk membuat konfigurasi peluncuran dengan pemetaan perangkat blok untuk volume penyimpanan instance

Contoh ini membuat konfigurasi peluncuran ephemeral1 dengan volume penyimpanan instance dengan nama perangkat/dev/sdc.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --block-device-mappings '[{"DeviceName":"/dev/sdc","VirtualName":"ephemeral1"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [BlockDeviceMapping](#) di Referensi API Auto Scaling Amazon EC2.

Untuk informasi tentang sintaks untuk mengutip nilai parameter berformat JSON, lihat [Menggunakan tanda kutip dengan string di CLI di Panduan Pengguna Antarmuka Baris Perintah AWS.AWS](#)

Contoh 9: Untuk membuat konfigurasi peluncuran dan menekan perangkat blok agar tidak dilampirkan pada waktu peluncuran

Contoh ini membuat konfigurasi peluncuran yang menekan perangkat blok yang ditentukan oleh pemetaan perangkat blok AMI (misalnya, `/dev/sdf`)

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --block-device-mappings '[{"DeviceName":"/dev/sdf","NoDevice":""}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [BlockDeviceMapping](#) di Referensi API Auto Scaling Amazon EC2.

Untuk informasi tentang sintaks untuk mengutip nilai parameter berformat JSON, lihat [Menggunakan tanda kutip dengan string di CLI di Panduan Pengguna Antarmuka Baris Perintah AWS.AWS](#)

- Untuk detail API, lihat [CreateLaunchConfiguration](#) di Referensi AWS CLI Perintah.

## **create-or-update-tags**

Contoh kode berikut menunjukkan cara menggunakan `create-or-update-tags`.

### AWS CLI

Untuk membuat atau memperbarui tag untuk grup Auto Scaling

Contoh ini menambahkan dua tag ke grup Auto Scaling yang ditentukan.

```
aws autoscaling create-or-update-tags \  
  --tags ResourceId=my-asg,ResourceType=auto-scaling-  
group,Key=Role,Value=WebServer,PropagateAtLaunch=true ResourceId=my-  
asg,ResourceType=auto-scaling-group,Key=Dept,Value=Research,PropagateAtLaunch=true
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menandai instans dan grup Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [CreateOrUpdateTags](#) di Referensi AWS CLI Perintah.

## **delete-auto-scaling-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-auto-scaling-group`.

### AWS CLI

Contoh 1: Untuk menghapus grup Auto Scaling yang ditentukan

Contoh ini menghapus grup Auto Scaling yang ditentukan.

```
aws autoscaling delete-auto-scaling-group \  
  --auto-scaling-group-name my-asg
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus infrastruktur Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk menghapus paksa grup Auto Scaling yang ditentukan

Untuk menghapus grup Auto Scaling tanpa menunggu instance dalam grup dihentikan, gunakan opsi. `--force-delete`

```
aws autoscaling delete-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --force-delete
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus infrastruktur Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DeleteAutoScalingGroup](#) di Referensi AWS CLI Perintah.

## **delete-launch-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-launch-configuration`.

### AWS CLI

Untuk menghapus konfigurasi peluncuran

Contoh ini menghapus konfigurasi peluncuran yang ditentukan.

```
aws autoscaling delete-launch-configuration \  
  --launch-configuration-name my-launch-config
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus infrastruktur Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DeleteLaunchConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-lifecycle-hook**

Contoh kode berikut menunjukkan cara menggunakan `delete-lifecycle-hook`.

### AWS CLI

Untuk menghapus kait siklus hidup

Contoh ini menghapus hook siklus hidup yang ditentukan.

```
aws autoscaling delete-lifecycle-hook \  
  --lifecycle-hook-name my-lifecycle-hook \  
  --auto-scaling-group-name my-asg
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteLifecycleHook](#) di Referensi AWS CLI Perintah.

## **delete-notification-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-notification-configuration`.

### AWS CLI

Untuk menghapus notifikasi Auto Scaling

Contoh ini menghapus notifikasi yang ditentukan dari grup Auto Scaling yang ditentukan.

```
aws autoscaling delete-notification-configuration \  
  --auto-scaling-group-name my-asg \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic
```



Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus konfigurasi notifikasi](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DeleteNotificationConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-policy`.

### AWS CLI

Untuk menghapus kebijakan penskalaan

Contoh ini menghapus kebijakan penskalaan yang ditentukan.

```
aws autoscaling delete-policy \  
  --auto-scaling-group-name my-asg \  
  --policy-name alb1000-target-tracking-scaling-policy
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## **delete-scheduled-action**

Contoh kode berikut menunjukkan cara menggunakan `delete-scheduled-action`.

### AWS CLI

Untuk menghapus tindakan terjadwal dari grup Auto Scaling

Contoh ini menghapus tindakan terjadwal yang ditentukan dari grup Auto Scaling yang ditentukan.

```
aws autoscaling delete-scheduled-action \  
  --auto-scaling-group-name my-asg \  
  --scheduled-action-name my-scheduled-action
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteScheduledAction](#) di Referensi AWS CLI Perintah.

## delete-tags

Contoh kode berikut menunjukkan cara menggunakan `delete-tags`.

### AWS CLI

Untuk menghapus tag dari grup Auto Scaling

Contoh ini menghapus tag yang ditentukan dari grup Auto Scaling yang ditentukan.

```
aws autoscaling delete-tags \  
  --tags ResourceId=my-asg,ResourceType=auto-scaling-group,Key=Dept,Value=Research
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menandai instans dan grup Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [DeleteTags](#) di Referensi AWS CLI Perintah.

## delete-warm-pool

Contoh kode berikut menunjukkan cara menggunakan `delete-warm-pool`.

### AWS CLI

Contoh 1: Untuk menghapus kolam hangat

Contoh berikut menghapus kolam hangat untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling delete-warm-pool \  
  --auto-scaling-group-name my-asg
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kolam hangat untuk Penskalaan Otomatis Amazon EC2 di Panduan Pengguna Auto Scaling Amazon EC2](#).

Contoh 2: Untuk memaksa menghapus kolam hangat

Untuk menghapus kolam hangat tanpa menunggu instance-nya berakhir, gunakan opsi. `--force-delete`

```
aws autoscaling delete-warm-pool \  
  --auto-scaling-group-name my-asg \  
  --force-delete
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kolam hangat untuk Penskalaan Otomatis Amazon EC2 di Panduan Pengguna Auto Scaling Amazon EC2](#).

- Untuk detail API, lihat [DeleteWarmPool](#) di Referensi AWS CLI Perintah.

## describe-account-limits

Contoh kode berikut menunjukkan cara menggunakan `describe-account-limits`.

### AWS CLI

Untuk menjelaskan batas akun Auto Scaling Amazon EC2 Anda

Contoh ini menjelaskan batas Auto Scaling Amazon EC2 untuk akun Anda. AWS

```
aws autoscaling describe-account-limits
```

Output:

```
{  
  "NumberOfLaunchConfigurations": 5,  
  "MaxNumberOfLaunchConfigurations": 100,  
  "NumberOfAutoScalingGroups": 3,  
  "MaxNumberOfAutoScalingGroups": 20  
}
```

Untuk informasi selengkapnya, lihat kuota [layanan Auto Scaling Amazon EC2](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DescribeAccountLimits](#) di Referensi AWS CLI Perintah.

## describe-adjustment-types

Contoh kode berikut menunjukkan cara menggunakan `describe-adjustment-types`.

## AWS CLI

Untuk menjelaskan jenis penyesuaian penskalaan yang tersedia

Contoh ini menjelaskan jenis penyesuaian yang tersedia.

```
aws autoscaling describe-adjustment-types
```

Output:

```
{
  "AdjustmentTypes": [
    {
      "AdjustmentType": "ChangeInCapacity"
    },
    {
      "AdjustmentType": "ExactCapacity"
    },
    {
      "AdjustmentType": "PercentChangeInCapacity"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Jenis penskalaan langkah](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [DescribeAdjustmentTypes](#) di Referensi AWS CLI Perintah.

## **describe-auto-scaling-groups**

Contoh kode berikut menunjukkan cara menggunakan `describe-auto-scaling-groups`.

## AWS CLI

Contoh 1: Untuk menggambarkan grup Auto Scaling yang ditentukan

Contoh ini menjelaskan grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-auto-scaling-groups \
  --auto-scaling-group-name my-asg
```

## Output:

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupName": "my-asg",
      "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:930d940e-891e-4781-
a11a-7b0acd480f03:autoScalingGroupName/my-asg",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-1234567890abcde12"
      },
      "MinSize": 0,
      "MaxSize": 1,
      "DesiredCapacity": 1,
      "DefaultCooldown": 300,
      "AvailabilityZones": [
        "us-west-2a",
        "us-west-2b",
        "us-west-2c"
      ],
      "LoadBalancerNames": [],
      "TargetGroupARNs": [],
      "HealthCheckType": "EC2",
      "HealthCheckGracePeriod": 0,
      "Instances": [
        {
          "InstanceId": "i-06905f55584de02da",
          "InstanceType": "t2.micro",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService",
          "ProtectedFromScaleIn": false,
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-1234567890abcde12"
          }
        }
      ],
      "CreatedTime": "2023-10-28T02:39:22.152Z",
      "SuspendedProcesses": [],
    }
  ]
}
```

```

    "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
    "EnabledMetrics": [],
    "Tags": [],
    "TerminationPolicies": [
      "Default"
    ],
    "NewInstancesProtectedFromScaleIn": false,
    "ServiceLinkedRoleARN": "arn",
    "TrafficSources": []
  }
]
}

```

Contoh 2: Untuk mendeskripsikan 100 grup Auto Scaling pertama yang ditentukan

Contoh ini menjelaskan grup Auto Scaling yang ditentukan. Ini memungkinkan Anda untuk menentukan hingga 100 nama grup.

```

aws autoscaling describe-auto-scaling-groups \
  --max-items 100 \
  --auto-scaling-group-name "group1" "group2" "group3" "group4"

```

Lihat contoh 1 untuk output sampel.

Contoh 3: Untuk mendeskripsikan grup Auto Scaling di wilayah tertentu

Contoh ini menjelaskan grup Auto Scaling di wilayah tertentu, hingga maksimal 75 grup.

```

aws autoscaling describe-auto-scaling-groups \
  --max-items 75 \
  --region us-east-1

```

Lihat contoh 1 untuk output sampel.

Contoh 4: Untuk menjelaskan jumlah grup Auto Scaling yang ditentukan

Untuk mengembalikan sejumlah grup Auto Scaling tertentu, gunakan opsi. `--max-items`

```

aws autoscaling describe-auto-scaling-groups \
  --max-items 1

```

Lihat contoh 1 untuk output sampel.

Jika output termasuk NextToken bidang, ada lebih banyak grup. Untuk mendapatkan grup tambahan, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya sebagai berikut.

```
aws autoscaling describe-auto-scaling-groups \
  --starting-token Z3M3LMPEXAMPLE
```

Lihat contoh 1 untuk output sampel.

Contoh 5: Untuk menjelaskan grup Auto Scaling yang menggunakan konfigurasi peluncuran

Contoh ini menggunakan `--query` opsi untuk menjelaskan grup Auto Scaling yang menggunakan konfigurasi peluncuran.

```
aws autoscaling describe-auto-scaling-groups \
  --query 'AutoScalingGroups[?LaunchConfigurationName!=`null`]'
```

Output:

```
[
  {
    "AutoScalingGroupName": "my-asg",
    "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:930d940e-891e-4781-
a11a-7b0acd480f03:autoScalingGroupName/my-asg",
    "LaunchConfigurationName": "my-lc",
    "MinSize": 0,
    "MaxSize": 1,
    "DesiredCapacity": 1,
    "DefaultCooldown": 300,
    "AvailabilityZones": [
      "us-west-2a",
      "us-west-2b",
      "us-west-2c"
    ],
    "LoadBalancerNames": [],
    "TargetGroupARNs": [],
    "HealthCheckType": "EC2",
    "HealthCheckGracePeriod": 0,
    "Instances": [
      {
        "InstanceId": "i-088c57934a6449037",
```

```
        "InstanceType": "t2.micro",
        "AvailabilityZone": "us-west-2c",
        "HealthStatus": "Healthy",
        "LifecycleState": "InService",
        "LaunchConfigurationName": "my-lc",
        "ProtectedFromScaleIn": false
    }
],
"CreatedTime": "2023-10-28T02:39:22.152Z",
"SuspendedProcesses": [],
"VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
"EnabledMetrics": [],
"Tags": [],
"TerminationPolicies": [
    "Default"
],
"NewInstancesProtectedFromScaleIn": false,
"ServiceLinkedRoleARN": "arn",
"TrafficSources": []
}
]
```

Untuk informasi selengkapnya, lihat [Memfilter output AWS CLI](#) di Panduan Pengguna Antarmuka Baris AWS Perintah.

- Untuk detail API, lihat [DescribeAutoScalingGroups](#) di Referensi AWS CLI Perintah.

## describe-auto-scaling-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-auto-scaling-instances`.

### AWS CLI

Contoh 1: Untuk menggambarkan satu atau lebih contoh

Contoh ini menjelaskan contoh yang ditentukan.

```
aws autoscaling describe-auto-scaling-instances \
  --instance-ids i-06905f55584de02da
```

Output:

```
{
```



```
"AutoScalingInstances": [  
  {  
    "InstanceId": "i-06905f55584de02da",  
    "InstanceType": "t2.micro",  
    "AutoScalingGroupName": "my-asg",  
    "AvailabilityZone": "us-west-2b",  
    "LifecycleState": "InService",  
    "HealthStatus": "HEALTHY",  
    "ProtectedFromScaleIn": false,  
    "LaunchTemplate": {  
      "LaunchTemplateId": "lt-1234567890abcde12",  
      "LaunchTemplateName": "my-launch-template",  
      "Version": "1"  
    }  
  }  
]
```

Contoh 2: Untuk menggambarkan satu atau lebih contoh

Contoh ini menggunakan `--max-items` opsi untuk menentukan berapa banyak instance yang akan dikembalikan dengan panggilan ini.

```
aws autoscaling describe-auto-scaling-instances \  
  --max-items 1
```

Jika output menyertakan `NextToken` bidang, ada lebih banyak contoh. Untuk mendapatkan instance tambahan, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya sebagai berikut.

```
aws autoscaling describe-auto-scaling-instances \  
  --starting-token Z3M3LMPEXAMPLE
```

Lihat contoh 1 untuk output sampel.

Contoh 3: Untuk mendeskripsikan instance yang menggunakan konfigurasi peluncuran

Contoh ini menggunakan `--query` opsi untuk mendeskripsikan instance yang menggunakan konfigurasi peluncuran.

```
aws autoscaling describe-auto-scaling-instances \  
  --query 'AutoScalingInstances[?LaunchConfigurationName!=`null`]'
```

**Output:**

```
[
  {
    "InstanceId": "i-088c57934a6449037",
    "InstanceType": "t2.micro",
    "AutoScalingGroupName": "my-asg",
    "AvailabilityZone": "us-west-2c",
    "LifecycleState": "InService",
    "HealthStatus": "HEALTHY",
    "LaunchConfigurationName": "my-lc",
    "ProtectedFromScaleIn": false
  }
]
```

Untuk informasi selengkapnya, lihat [Memfilter output AWS CLI](#) di Panduan Pengguna Antarmuka Baris AWS Perintah.

- Untuk detail API, lihat [DescribeAutoScalingInstances](#) di Referensi AWS CLI Perintah.

**describe-auto-scaling-notification-types**

Contoh kode berikut menunjukkan cara menggunakan `describe-auto-scaling-notification-types`.

**AWS CLI**

Untuk menjelaskan jenis notifikasi yang tersedia

Contoh ini menjelaskan jenis notifikasi yang tersedia.

```
aws autoscaling describe-auto-scaling-notification-types
```

**Output:**

```
{
  "AutoScalingNotificationTypes": [
    "autoscaling:EC2_INSTANCE_LAUNCH",
    "autoscaling:EC2_INSTANCE_LAUNCH_ERROR",
    "autoscaling:EC2_INSTANCE_TERMINATE",
    "autoscaling:EC2_INSTANCE_TERMINATE_ERROR",
    "autoscaling:TEST_NOTIFICATION"
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan notifikasi Amazon SNS ketika grup Auto Scaling Anda menskalakan](#) ke dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [DescribeAutoScalingNotificationTypes](#) di Referensi AWS CLI Perintah.

## **describe-instance-refreshes**

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-refreshes`.

### AWS CLI

Untuk menggambarkan penyegaran instance

`describe-instance-refreshes` Contoh berikut menampilkan deskripsi semua permintaan penyegaran instans untuk grup Auto Scaling yang ditentukan, termasuk pesan status dan (jika tersedia) alasan status.

```
aws autoscaling describe-instance-refreshes \
  --auto-scaling-group-name my-asg
```

### Output:

```
{
  "InstanceRefreshes": [
    {
      "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b",
      "AutoScalingGroupName": "my-asg",
      "Status": "InProgress",
      "StatusReason": "Waiting for instances to warm up before continuing. For
example: 0e69cc3f05f825f4f is warming up.",
      "EndTime": "2023-03-23T16:42:55Z",
      "PercentageComplete": 0,
      "InstancesToUpdate": 0,
      "Preferences": {
        "MinHealthyPercentage": 100,
        "InstanceWarmup": 300,
        "CheckpointPercentages": [
          50
        ],
      },
    },
  ],
}
```

```
        "CheckpointDelay": 3600,  
        "SkipMatching": false,  
        "AutoRollback": true,  
        "ScaleInProtectedInstances": "Ignore",  
        "StandbyInstances": "Ignore"  
    }  
  },  
  {  
    "InstanceRefreshId": "dd7728d0-5bc4-4575-96a3-1b2c52bf8bb1",  
    "AutoScalingGroupName": "my-asg",  
    "Status": "Successful",  
    "EndTime": "2022-06-02T16:53:37Z",  
    "PercentageComplete": 100,  
    "InstancesToUpdate": 0,  
    "Preferences": {  
      "MinHealthyPercentage": 90,  
      "InstanceWarmup": 300,  
      "SkipMatching": true,  
      "AutoRollback": true,  
      "ScaleInProtectedInstances": "Ignore",  
      "StandbyInstances": "Ignore"  
    }  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Memeriksa status penyegaran instans](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DescribeInstanceRefreshes](#) di Referensi AWS CLI Perintah.

## **describe-launch-configurations**

Contoh kode berikut menunjukkan cara menggunakan `describe-launch-configurations`.

### **AWS CLI**

Contoh 1: Untuk menjelaskan konfigurasi peluncuran yang ditentukan

Contoh ini menjelaskan konfigurasi peluncuran yang ditentukan.

```
aws autoscaling describe-launch-configurations \  
  --launch-configuration-names my-launch-config
```

## Output:

```

{
  "LaunchConfigurations": [
    {
      "LaunchConfigurationName": "my-launch-config",
      "LaunchConfigurationARN": "arn:aws:autoscaling:us-
west-2:123456789012:launchConfiguration:98d3b196-4cf9-4e88-8ca1-8547c24ced8b:launchConfigura
my-launch-config",
      "ImageId": "ami-0528a5175983e7f28",
      "KeyName": "my-key-pair-uswest2",
      "SecurityGroups": [
        "sg-05eaec502fcdadc2e"
      ],
      "ClassicLinkVPCSecurityGroups": [],
      "UserData": "",
      "InstanceType": "t2.micro",
      "KernelId": "",
      "RamdiskId": "",
      "BlockDeviceMappings": [
        {
          "DeviceName": "/dev/xvda",
          "Ebs": {
            "SnapshotId": "snap-06c1606ba5ca274b1",
            "VolumeSize": 8,
            "VolumeType": "gp2",
            "DeleteOnTermination": true,
            "Encrypted": false
          }
        }
      ],
      "InstanceMonitoring": {
        "Enabled": true
      },
      "CreatedTime": "2020-10-28T02:39:22.321Z",
      "EbsOptimized": false,
      "AssociatePublicIpAddress": true,
      "MetadataOptions": {
        "HttpTokens": "required",
        "HttpPutResponseHopLimit": 1,
        "HttpEndpoint": "disabled"
      }
    }
  ]
}

```

```
}

```

Contoh 2: Untuk menjelaskan sejumlah konfigurasi peluncuran yang ditentukan

Untuk mengembalikan sejumlah konfigurasi peluncuran tertentu, gunakan `--max-items` opsi.

```
aws autoscaling describe-launch-configurations \
  --max-items 1

```

Jika output menyertakan `NextToken` bidang, ada lebih banyak konfigurasi peluncuran. Untuk mendapatkan konfigurasi peluncuran tambahan, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya sebagai berikut.

```
aws autoscaling describe-launch-configurations \
  --starting-token Z3M3LMPEXAMPLE

```

- Untuk detail API, lihat [DescribeLaunchConfigurations](#) di Referensi AWS CLI Perintah.

## **describe-lifecycle-hook-types**

Contoh kode berikut menunjukkan cara menggunakan `describe-lifecycle-hook-types`.

### AWS CLI

Untuk menjelaskan jenis kait siklus hidup yang tersedia

Contoh ini menjelaskan jenis kait siklus hidup yang tersedia.

```
aws autoscaling describe-lifecycle-hook-types

```

Output:

```
{
  "LifecycleHookTypes": [
    "autoscaling:EC2_INSTANCE_LAUNCHING",
    "autoscaling:EC2_INSTANCE_TERMINATING"
  ]
}
```

- Untuk detail API, lihat [DescribeLifecycleHookTypes](#) di Referensi AWS CLI Perintah.

## describe-lifecycle-hooks

Contoh kode berikut menunjukkan cara menggunakan `describe-lifecycle-hooks`.

### AWS CLI

Untuk mendeskripsikan kait siklus hidup Anda

Contoh ini menjelaskan kait siklus hidup untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-lifecycle-hooks \
  --auto-scaling-group-name my-asg
```

### Output:

```
{
  "LifecycleHooks": [
    {
      "GlobalTimeout": 3000,
      "HeartbeatTimeout": 30,
      "AutoScalingGroupName": "my-asg",
      "LifecycleHookName": "my-launch-hook",
      "DefaultResult": "ABANDON",
      "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING"
    },
    {
      "GlobalTimeout": 6000,
      "HeartbeatTimeout": 60,
      "AutoScalingGroupName": "my-asg",
      "LifecycleHookName": "my-termination-hook",
      "DefaultResult": "CONTINUE",
      "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeLifecycleHooks](#) di Referensi AWS CLI Perintah.

## describe-load-balancer-target-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancer-target-groups`.

## AWS CLI

Untuk mendeskripsikan grup target penyeimbang beban untuk grup Auto Scaling

Contoh ini menjelaskan grup target penyeimbang beban yang dilampirkan ke grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-load-balancer-target-groups \  
  --auto-scaling-group-name my-asg
```

Output:

```
{  
  "LoadBalancerTargetGroups": [  
    {  
      "LoadBalancerTargetGroupARN": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",  
      "State": "Added"  
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeLoadBalancerTargetGroups](#) di Referensi AWS CLI Perintah.

## describe-load-balancers

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancers`.

## AWS CLI

Untuk mendeskripsikan Classic Load Balancer untuk grup Auto Scaling

Contoh ini menjelaskan Classic Load Balancer untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-load-balancers \  
  --auto-scaling-group-name my-asg
```

Output:

```
{  
  "LoadBalancers": [  
    {
```



```
        "State": "Added",
        "LoadBalancerName": "my-load-balancer"
    }
]
}
```

- Untuk detail API, lihat [DescribeLoadBalancers](#) di Referensi AWS CLI Perintah.

## describe-metric-collection-types

Contoh kode berikut menunjukkan cara menggunakan `describe-metric-collection-types`.

### AWS CLI

Untuk menjelaskan jenis koleksi metrik yang tersedia

Contoh ini menjelaskan jenis koleksi metrik yang tersedia.

```
aws autoscaling describe-metric-collection-types
```

Output:

```
{
  "Metrics": [
    {
      "Metric": "GroupMinSize"
    },
    {
      "Metric": "GroupMaxSize"
    },
    {
      "Metric": "GroupDesiredCapacity"
    },
    {
      "Metric": "GroupInServiceInstances"
    },
    {
      "Metric": "GroupInServiceCapacity"
    },
    {
      "Metric": "GroupPendingInstances"
    },
    {
```

```

    "Metric": "GroupPendingCapacity"
  },
  {
    "Metric": "GroupTerminatingInstances"
  },
  {
    "Metric": "GroupTerminatingCapacity"
  },
  {
    "Metric": "GroupStandbyInstances"
  },
  {
    "Metric": "GroupStandbyCapacity"
  },
  {
    "Metric": "GroupTotalInstances"
  },
  {
    "Metric": "GroupTotalCapacity"
  }
],
"Granularities": [
  {
    "Granularity": "1Minute"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Metrik grup Auto Scaling di Panduan Pengguna Auto Scaling Amazon EC2](#).

- Untuk detail API, lihat [DescribeMetricCollectionTypes](#) di Referensi AWS CLI Perintah.

## describe-notification-configurations

Contoh kode berikut menunjukkan cara menggunakan `describe-notification-configurations`.

### AWS CLI

Contoh 1: Untuk menjelaskan konfigurasi notifikasi dari grup tertentu

Contoh ini menjelaskan konfigurasi notifikasi untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-notification-configurations \  
  --auto-scaling-group-name my-asg
```

Output:

```
{  
  "NotificationConfigurations": [  
    {  
      "AutoScalingGroupName": "my-asg",  
      "NotificationType": "autoscaling:TEST_NOTIFICATION",  
      "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic-2"  
    },  
    {  
      "AutoScalingGroupName": "my-asg",  
      "NotificationType": "autoscaling:TEST_NOTIFICATION",  
      "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan notifikasi Amazon SNS ketika grup Auto Scaling Anda menskalakan](#) ke dalam Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 1: Untuk menjelaskan sejumlah konfigurasi notifikasi tertentu

Untuk mengembalikan sejumlah konfigurasi notifikasi tertentu, gunakan `max-items` parameter.

```
aws autoscaling describe-notification-configurations \  
  --auto-scaling-group-name my-auto-scaling-group \  
  --max-items 1
```

Output:

```
{  
  "NotificationConfigurations": [  
    {  
      "AutoScalingGroupName": "my-asg",  
      "NotificationType": "autoscaling:TEST_NOTIFICATION",  
      "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic-2"  
    },  
    {  
      "AutoScalingGroupName": "my-asg",
```

```

        "NotificationType": "autoscaling:TEST_NOTIFICATION",
        "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic"
    }
]
}

```

Jika output menyertakan NextToken bidang, ada lebih banyak konfigurasi notifikasi. Untuk mendapatkan konfigurasi notifikasi tambahan, gunakan nilai bidang ini dengan starting-token parameter dalam panggilan berikutnya sebagai berikut.

```

aws autoscaling describe-notification-configurations \
  --auto-scaling-group-name my-asg \
  --starting-token Z3M3LMPEXAMPLE

```

Untuk informasi selengkapnya, lihat [Mendapatkan notifikasi Amazon SNS ketika grup Auto Scaling Anda menskalakan](#) ke dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [DescribeNotificationConfigurations](#) di Referensi AWS CLI Perintah.

## describe-policies

Contoh kode berikut menunjukkan cara menggunakan describe-policies.

### AWS CLI

Contoh 1: Untuk menjelaskan kebijakan penskalaan grup tertentu

Contoh ini menjelaskan kebijakan penskalaan untuk grup Auto Scaling yang ditentukan.

```

aws autoscaling describe-policies \
  --auto-scaling-group-name my-asg

```

Output:

```

{
  "ScalingPolicies": [
    {
      "AutoScalingGroupName": "my-asg",
      "PolicyName": "alb1000-target-tracking-scaling-policy",
      "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:3065d9c8-9969-4bec-

```

```

bb6a-3fbe5550fde6:autoScalingGroupName/my-asg:policyName/alb1000-target-tracking-
scaling-policy",
    "PolicyType": "TargetTrackingScaling",
    "StepAdjustments": [],
    "Alarms": [
        {
            "AlarmName": "TargetTracking-my-asg-
AlarmHigh-924887a9-12d7-4e01-8686-6f844d13a196",
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-
AlarmHigh-924887a9-12d7-4e01-8686-6f844d13a196"
        },
        {
            "AlarmName": "TargetTracking-my-asg-AlarmLow-f96f899d-b8e7-4d09-
a010-c1aaa35da296",
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-f96f899d-b8e7-4d09-a010-
c1aaa35da296"
        }
    ],
    "TargetTrackingConfiguration": {
        "PredefinedMetricSpecification": {
            "PredefinedMetricType": "ALBRequestCountPerTarget",
            "ResourceLabel": "app/my-alb/778d41231b141a0f/targetgroup/my-
alb-target-group/943f017f100becff"
        },
        "TargetValue": 1000.0,
        "DisableScaleIn": false
    },
    "Enabled": true
},
{
    "AutoScalingGroupName": "my-asg",
    "PolicyName": "cpu40-target-tracking-scaling-policy",
    "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:5fd26f71-39d4-4690-82a9-
b8515c45cdde:autoScalingGroupName/my-asg:policyName/cpu40-target-tracking-scaling-
policy",
    "PolicyType": "TargetTrackingScaling",
    "StepAdjustments": [],
    "Alarms": [
        {
            "AlarmName": "TargetTracking-my-asg-
AlarmHigh-139f9789-37b9-42ad-bea5-b5b147d7f473",

```

```

        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmHigh-139f9789-37b9-42ad-bea5-
b5b147d7f473"
      },
      {
        "AlarmName": "TargetTracking-my-asg-AlarmLow-bd681c67-
fc18-4c56-8468-fb8e413009c9",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-bd681c67-fc18-4c56-8468-
fb8e413009c9"
      }
    ],
    "TargetTrackingConfiguration": {
      "PredefinedMetricSpecification": {
        "PredefinedMetricType": "ASGAverageCPUUtilization"
      },
      "TargetValue": 40.0,
      "DisableScaleIn": false
    },
    "Enabled": true
  }
]
}

```

Untuk informasi selengkapnya, lihat [Penskalaan dinamis](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk menjelaskan kebijakan penskalaan dari nama tertentu

Untuk mengembalikan kebijakan penskalaan tertentu, gunakan `--policy-names` opsi.

```

aws autoscaling describe-policies \
  --auto-scaling-group-name my-asg \
  --policy-names cpu40-target-tracking-scaling-policy

```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Penskalaan dinamis](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 3: Untuk menjelaskan sejumlah kebijakan penskalaan

Untuk mengembalikan sejumlah kebijakan tertentu, gunakan `--max-items` opsi.

```
aws autoscaling describe-policies \  
  --auto-scaling-group-name my-asg \  
  --max-items 1
```

Lihat contoh 1 untuk output sampel.

Jika output menyertakan NextToken bidang, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya untuk mendapatkan kebijakan tambahan.

```
aws autoscaling describe-policies --auto-scaling-group-name my-asg --starting-token  
Z3M3LMPEXAMPLE
```

Untuk informasi selengkapnya, lihat [Penskalaan dinamis](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DescribePolicies](#) di Referensi AWS CLI Perintah.

## describe-scaling-activities

Contoh kode berikut menunjukkan cara menggunakan `describe-scaling-activities`.

### AWS CLI

Contoh 1: Untuk menjelaskan aktivitas penskalaan untuk grup yang ditentukan

Contoh ini menjelaskan aktivitas penskalaan untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-scaling-activities \  
  --auto-scaling-group-name my-asg
```

Output:

```
{  
  "Activities": [  
    {  
      "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",  
      "Description": "Launching a new EC2 instance: i-0d44425630326060f",  
      "AutoScalingGroupName": "my-asg",  
      "Cause": "At 2020-10-30T19:35:51Z a user request update of  
AutoScalingGroup constraints to min: 0, max: 16, desired: 16 changing the desired  
capacity from 0 to 16. At 2020-10-30T19:36:07Z an instance was started in response
```

```

to a difference between desired and actual capacity, increasing the capacity from 0
to 16.",
    "StartTime": "2020-10-30T19:36:09.766Z",
    "EndTime": "2020-10-30T19:36:41Z",
    "StatusCode": "Successful",
    "Progress": 100,
    "Details": "{\"Subnet ID\": \"subnet-5ea0c127\", \"Availability Zone\":
\"us-west-2b\"}"
  }
]
}

```

Untuk selengkapnya, lihat [Memverifikasi aktivitas penskalaan untuk grup Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk menjelaskan aktivitas penskalaan untuk grup yang dihapus

Untuk menjelaskan aktivitas penskalaan setelah grup Auto Scaling dihapus, tambahkan `--include-deleted-groups` opsi.

```

aws autoscaling describe-scaling-activities \
  --auto-scaling-group-name my-asg \
  --include-deleted-groups

```

Output:

```

{
  "Activities": [
    {
      "ActivityId": "e1f5de0e-f93e-1417-34ac-092a76fba220",
      "Description": "Launching a new EC2 instance. Status Reason: Your Spot
request price of 0.001 is lower than the minimum required Spot request fulfillment
price of 0.0031. Launching EC2 instance failed.",
      "AutoScalingGroupName": "my-asg",
      "Cause": "At 2021-01-13T20:47:24Z a user request update of
AutoScalingGroup constraints to min: 1, max: 5, desired: 3 changing the desired
capacity from 0 to 3. At 2021-01-13T20:47:27Z an instance was started in response
to a difference between desired and actual capacity, increasing the capacity from 0
to 3.",
      "StartTime": "2021-01-13T20:47:30.094Z",
      "EndTime": "2021-01-13T20:47:30Z",
      "StatusCode": "Failed",
    }
  ]
}

```



```

    "StatusMessage": "Your Spot request price of 0.001 is lower than the
minimum required Spot request fulfillment price of 0.0031. Launching EC2 instance
failed.",
    "Progress": 100,
    "Details": "{\"Subnet ID\": \"subnet-5ea0c127\", \"Availability Zone\":
\"us-west-2b\"}",
    "AutoScalingGroupState": "Deleted",
    "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:283179a2-
f3ce-423d-93f6-66bb518232f7:autoScalingGroupName/my-asg"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Memecahkan Masalah Penskalaan Otomatis Amazon EC2 di Panduan Pengguna Auto Scaling](#) Amazon EC2.

Contoh 3: Untuk menjelaskan sejumlah aktivitas penskalaan tertentu

Untuk mengembalikan sejumlah aktivitas tertentu, gunakan `--max-items` opsi.

```

aws autoscaling describe-scaling-activities \
  --max-items 1

```

Output:

```

{
  "Activities": [
    {
      "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",
      "Description": "Launching a new EC2 instance: i-0d44425630326060f",
      "AutoScalingGroupName": "my-asg",
      "Cause": "At 2020-10-30T19:35:51Z a user request update of
AutoScalingGroup constraints to min: 0, max: 16, desired: 16 changing the desired
capacity from 0 to 16. At 2020-10-30T19:36:07Z an instance was started in response
to a difference between desired and actual capacity, increasing the capacity from 0
to 16.",
      "StartTime": "2020-10-30T19:36:09.766Z",
      "EndTime": "2020-10-30T19:36:41Z",
      "StatusCode": "Successful",
      "Progress": 100,
      "Details": "{\"Subnet ID\": \"subnet-5ea0c127\", \"Availability Zone\":
\"us-west-2b\"}"
    }
  ]
}

```

```
    }  
  ]  
}
```

Jika output termasuk `NextToken` bidang, ada lebih banyak kegiatan. Untuk mendapatkan aktivitas tambahan, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya sebagai berikut.

```
aws autoscaling describe-scaling-activities \  
  --starting-token Z3M3LMPEXAMPLE
```

Untuk selengkapnya, lihat [Memverifikasi aktivitas penskalaan untuk grup Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DescribeScalingActivities](#) di Referensi AWS CLI Perintah.

## describe-scaling-process-types

Contoh kode berikut menunjukkan cara menggunakan `describe-scaling-process-types`.

### AWS CLI

Untuk menggambarkan jenis proses yang tersedia

Contoh ini menjelaskan jenis proses yang tersedia.

```
aws autoscaling describe-scaling-process-types
```

Output:

```
{  
  "Processes": [  
    {  
      "ProcessName": "AZRebalance"  
    },  
    {  
      "ProcessName": "AddToLoadBalancer"  
    },  
    {  
      "ProcessName": "AlarmNotification"  
    },  
  ],  
}
```

```
{
  "ProcessName": "HealthCheck"
},
{
  "ProcessName": "InstanceRefresh"
},
{
  "ProcessName": "Launch"
},
{
  "ProcessName": "ReplaceUnhealthy"
},
{
  "ProcessName": "ScheduledActions"
},
{
  "ProcessName": "Terminate"
}
]
```

Untuk informasi selengkapnya, lihat [Menanggukn dan melanjutkan proses penskalaan di Panduan Pengguna Auto Scaling](#) Amazon EC2.

- Untuk detail API, lihat [DescribeScalingProcessTypes](#) di Referensi AWS CLI Perintah.

## **describe-scheduled-actions**

Contoh kode berikut menunjukkan cara menggunakan `describe-scheduled-actions`.

### AWS CLI

Contoh 1: Untuk menggambarkan semua tindakan terjadwal

Contoh ini menjelaskan semua tindakan terjadwal Anda.

```
aws autoscaling describe-scheduled-actions
```

Output:

```
{
  "ScheduledUpdateGroupActions": [
```

```

    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,
      "MaxSize": 6,
      "DesiredCapacity": 4,
      "TimeZone": "America/New_York"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk menjelaskan tindakan terjadwal untuk grup yang ditentukan

Untuk menjelaskan tindakan terjadwal untuk grup Auto Scaling tertentu, gunakan opsi. `--auto-scaling-group-name`

```

aws autoscaling describe-scheduled-actions \
  --auto-scaling-group-name my-asg

```

Output:

```

{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,

```

```
        "MaxSize": 6,  
        "DesiredCapacity": 4,  
        "TimeZone": "America/New_York"  
    }  
]  
}
```

Untuk informasi selengkapnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 3: Untuk menggambarkan tindakan terjadwal yang ditentukan

Untuk menjelaskan tindakan terjadwal tertentu, gunakan `--scheduled-action-names` opsi.

```
aws autoscaling describe-scheduled-actions \  
  --scheduled-action-names my-recurring-action
```

Output:

```
{  
  "ScheduledUpdateGroupActions": [  
    {  
      "AutoScalingGroupName": "my-asg",  
      "ScheduledActionName": "my-recurring-action",  
      "Recurrence": "30 0 1 1,6,12 *",  
      "ScheduledActionARN": "arn:aws:autoscaling:us-  
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-  
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",  
      "StartTime": "2023-12-01T04:00:00Z",  
      "Time": "2023-12-01T04:00:00Z",  
      "MinSize": 1,  
      "MaxSize": 6,  
      "DesiredCapacity": 4,  
      "TimeZone": "America/New_York"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 4: Untuk menggambarkan tindakan terjadwal dengan waktu mulai yang ditentukan

Untuk menggambarkan tindakan terjadwal yang dimulai pada waktu tertentu, gunakan `--start-time` opsi.

```
aws autoscaling describe-scheduled-actions \  
--start-time "2023-12-01T04:00:00Z"
```

Output:

```
{  
  "ScheduledUpdateGroupActions": [  
    {  
      "AutoScalingGroupName": "my-asg",  
      "ScheduledActionName": "my-recurring-action",  
      "Recurrence": "30 0 1 1,6,12 *",  
      "ScheduledActionARN": "arn:aws:autoscaling:us-  
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-  
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",  
      "StartTime": "2023-12-01T04:00:00Z",  
      "Time": "2023-12-01T04:00:00Z",  
      "MinSize": 1,  
      "MaxSize": 6,  
      "DesiredCapacity": 4,  
      "TimeZone": "America/New_York"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 5: Untuk menggambarkan tindakan terjadwal yang berakhir pada waktu tertentu

Untuk menggambarkan tindakan terjadwal yang berakhir pada waktu tertentu, gunakan `--end-time` opsi.

```
aws autoscaling describe-scheduled-actions \  
--end-time "2023-12-01T04:00:00Z"
```

Output:

```
{
```

```

    "ScheduledUpdateGroupActions": [
      {
        "AutoScalingGroupName": "my-asg",
        "ScheduledActionName": "my-recurring-action",
        "Recurrence": "30 0 1 1,6,12 *",
        "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
        "StartTime": "2023-12-01T04:00:00Z",
        "Time": "2023-12-01T04:00:00Z",
        "MinSize": 1,
        "MaxSize": 6,
        "DesiredCapacity": 4,
        "TimeZone": "America/New_York"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 6: Untuk menggambarkan sejumlah tindakan terjadwal yang ditentukan

Untuk mengembalikan sejumlah tindakan terjadwal tertentu, gunakan `--max-items` opsi.

```

aws autoscaling describe-scheduled-actions \
  --auto-scaling-group-name my-asg \
  --max-items 1

```

Output:

```

{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,

```

```
        "MaxSize": 6,  
        "DesiredCapacity": 4,  
        "TimeZone": "America/New_York"  
    }  
]  
}
```

Jika output menyertakan `NextToken` bidang, ada lebih banyak tindakan terjadwal. Untuk mendapatkan tindakan terjadwal tambahan, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya sebagai berikut.

```
aws autoscaling describe-scheduled-actions \  
  --auto-scaling-group-name my-asg \  
  --starting-token Z3M3LMPEXAMPLE
```

Untuk informasi selengkapnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DescribeScheduledActions](#) di Referensi AWS CLI Perintah.

## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Untuk mendeskripsikan semua tag

Contoh ini menjelaskan semua tag Anda.

```
aws autoscaling describe-tags
```

Output:

```
{  
  "Tags": [  
    {  
      "ResourceType": "auto-scaling-group",  
      "ResourceId": "my-asg",  
      "PropagateAtLaunch": true,  
      "Value": "Research",  
      "Key": "Dept"    }  
  ]  
}
```



```
    },  
    {  
      "ResourceType": "auto-scaling-group",  
      "ResourceId": "my-asg",  
      "PropagateAtLaunch": true,  
      "Value": "WebServer",  
      "Key": "Role"  
    }  
  ]  
}
```

Untuk informasi lebih lanjut, lihat [Menandai instans dan grup Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 2: Untuk mendeskripsikan tag untuk grup tertentu

Untuk menjelaskan tag untuk grup Auto Scaling tertentu, gunakan opsi. `--filters`

```
aws autoscaling describe-tags --filters Name=auto-scaling-group,Values=my-asg
```

Untuk informasi lebih lanjut, lihat [Menandai instans dan grup Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 3: Untuk menggambarkan jumlah tag yang ditentukan

Untuk mengembalikan sejumlah tag tertentu, gunakan `--max-items` opsi.

```
aws autoscaling describe-tags \  
  --max-items 1
```

Jika output termasuk `NextToken` bidang, ada lebih banyak tag. Untuk mendapatkan tag tambahan, gunakan nilai bidang ini dengan `--starting-token` opsi dalam panggilan berikutnya sebagai berikut.

```
aws autoscaling describe-tags \  
  --filters Name=auto-scaling-group,Values=my-asg \  
  --starting-token Z3M3LMPEXAMPLE
```

Untuk informasi lebih lanjut, lihat [Menandai instans dan grup Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## describe-termination-policy-types

Contoh kode berikut menunjukkan cara menggunakan `describe-termination-policy-types`.

### AWS CLI

Untuk menjelaskan jenis kebijakan penghentian yang tersedia

Contoh ini menjelaskan jenis kebijakan penghentian yang tersedia.

```
aws autoscaling describe-termination-policy-types
```

Output:

```
{
  "TerminationPolicyTypes": [
    "AllocationStrategy",
    "ClosestToNextInstanceHour",
    "Default",
    "NewestInstance",
    "OldestInstance",
    "OldestLaunchConfiguration",
    "OldestLaunchTemplate"
  ]
}
```

Untuk informasi selengkapnya, lihat: [Mengendalikan instans Auto Scaling mana yang berakhir selama penurunan skala](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [DescribeTerminationPolicyTypes](#) di Referensi AWS CLI Perintah.

## describe-warm-pool

Contoh kode berikut menunjukkan cara menggunakan `describe-warm-pool`.

### AWS CLI

Untuk menggambarkan kolam yang hangat

Contoh ini menjelaskan kolam hangat untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling describe-warm-pool \
  --auto-scaling-group-name my-asg
```

**Output:**

```
{
  "WarmPoolConfiguration": {
    "MinSize": 2,
    "PoolState": "Stopped"
  },
  "Instances": [
    {
      "InstanceId": "i-070a5bbc7e7f40dc5",
      "InstanceType": "t2.micro",
      "AvailabilityZone": "us-west-2c",
      "LifecycleState": "Warmed:Pending",
      "HealthStatus": "Healthy",
      "LaunchTemplate": {
        "LaunchTemplateId": "lt-00a731f6e9fa48610",
        "LaunchTemplateName": "my-template-for-auto-scaling",
        "Version": "6"
      }
    },
    {
      "InstanceId": "i-0b52f061814d3bd2d",
      "InstanceType": "t2.micro",
      "AvailabilityZone": "us-west-2b",
      "LifecycleState": "Warmed:Pending",
      "HealthStatus": "Healthy",
      "LaunchTemplate": {
        "LaunchTemplateId": "lt-00a731f6e9fa48610",
        "LaunchTemplateName": "my-template-for-auto-scaling",
        "Version": "6"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kolam hangat untuk Penskalaan Otomatis Amazon EC2 di Panduan Pengguna Auto Scaling Amazon EC2](#).

- Untuk detail API, lihat [DescribeWarmPool](#) di Referensi AWS CLI Perintah.

**detach-instances**

Contoh kode berikut menunjukkan cara menggunakan `detach-instances`.

## AWS CLI

Untuk melepaskan instance dari grup Auto Scaling

Contoh ini melepaskan instance tertentu dari grup Auto Scaling yang ditentukan.

```
aws autoscaling detach-instances \  
  --instance-ids i-030017cfa84b20135 \  
  --auto-scaling-group-name my-asg \  
  --should-decrement-desired-capacity
```

Output:

```
{  
  "Activities": [  
    {  
      "ActivityId": "5091cb52-547a-47ce-a236-c9ccbc2cb2c9",  
      "AutoScalingGroupName": "my-asg",  
      "Description": "Detaching EC2 instance: i-030017cfa84b20135",  
      "Cause": "At 2020-10-31T17:35:04Z instance i-030017cfa84b20135 was  
detached in response to a user request, shrinking the capacity from 2 to 1.",  
      "StartTime": "2020-04-12T15:02:16.179Z",  
      "StatusCode": "InProgress",  
      "Progress": 50,  
      "Details": "{\"Subnet ID\": \"subnet-6194ea3b\", \"Availability Zone\":  
\"us-west-2c\"}"  
    }  
  ]  
}
```

- Untuk detail API, lihat [DetachInstances](#) di Referensi AWS CLI Perintah.

## **detach-load-balancer-target-groups**

Contoh kode berikut menunjukkan cara menggunakan `detach-load-balancer-target-groups`.

## AWS CLI

Untuk melepaskan grup target penyeimbang beban dari grup Auto Scaling

Contoh ini melepaskan grup target penyeimbang beban yang ditentukan dari grup Auto Scaling yang ditentukan.

```
aws autoscaling detach-load-balancer-target-groups \  
  --auto-scaling-group-name my-asg \  
  --target-group-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Perintah ini tidak menghasilkan output

Untuk informasi selengkapnya, lihat Memasang load balancer ke grup Auto Scaling Anda dalam Amazon EC2 Auto Scaling User Guide.

- Untuk detail API, lihat [DetachLoadBalancerTargetGroups](#) di Referensi AWS CLI Perintah.

## **detach-load-balancers**

Contoh kode berikut menunjukkan cara menggunakan `detach-load-balancers`.

### AWS CLI

Untuk melepaskan Classic Load Balancer dari grup Auto Scaling

Contoh ini melepaskan Classic Load Balancer yang ditentukan dari grup Auto Scaling yang ditentukan.

```
aws autoscaling detach-load-balancers \  
  --load-balancer-names my-load-balancer \  
  --auto-scaling-group-name my-asg
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat Memasang load balancer ke grup Auto Scaling Anda dalam Amazon EC2 Auto Scaling User Guide.

- Untuk detail API, lihat [DetachLoadBalancers](#) di Referensi AWS CLI Perintah.

## **disable-metrics-collection**

Contoh kode berikut menunjukkan cara menggunakan `disable-metrics-collection`.

### AWS CLI

Untuk menonaktifkan koleksi metrik untuk grup Auto Scaling

Contoh ini menonaktifkan pengumpulan GroupDesiredCapacity metrik untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling disable-metrics-collection \  
  --auto-scaling-group-name my-asg \  
  --metrics GroupDesiredCapacity
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [CloudWatch Metrik pemantauan untuk grup dan instans Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [DisableMetricsCollection](#) di Referensi AWS CLI Perintah.

### **enable-metrics-collection**

Contoh kode berikut menunjukkan cara menggunakan `enable-metrics-collection`.

#### AWS CLI

Contoh 1: Untuk mengaktifkan pengumpulan metrik untuk grup Auto Scaling

Contoh ini memungkinkan pengumpulan data untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling enable-metrics-collection \  
  --auto-scaling-group-name my-asg \  
  --granularity "1Minute"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [CloudWatch Metrik pemantauan untuk grup dan instans Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk mengumpulkan data untuk metrik yang ditentukan untuk grup Auto Scaling

Untuk mengumpulkan data untuk metrik tertentu, gunakan `--metrics` opsi.

```
aws autoscaling enable-metrics-collection \  
  --auto-scaling-group-name my-asg \  
  --metrics GroupDesiredCapacity --granularity "1Minute"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [CloudWatch Metrik pemantauan untuk grup dan instans Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [EnableMetricsCollection](#) di Referensi AWS CLI Perintah.

## enter-standby

Contoh kode berikut menunjukkan cara menggunakan `enter-standby`.

### AWS CLI

Untuk memindahkan instance ke mode siaga

Contoh ini menempatkan instance yang ditentukan ke mode siaga. Ini berguna untuk memperbaiki atau memecahkan masalah instance yang saat ini dalam layanan.

```
aws autoscaling enter-standby \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg \  
  --should-decrement-desired-capacity
```

Output:

```
{  
  "Activities": [  
    {  
      "ActivityId": "ffa056b4-6ed3-41ba-ae7c-249dfae6eba1",  
      "AutoScalingGroupName": "my-asg",  
      "Description": "Moving EC2 instance to Standby: i-061c63c5eb45f0416",  
      "Cause": "At 2020-10-31T20:31:00Z instance i-061c63c5eb45f0416 was moved  
to standby in response to a user request, shrinking the capacity from 1 to 0.",  
      "StartTime": "2020-10-31T20:31:00.949Z",  
      "StatusCode": "InProgress",  
      "Progress": 50,  
      "Details": "{\"Subnet ID\": \"subnet-6194ea3b\", \"Availability Zone\":  
\"us-west-2c\"}"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat Siklus hidup [instans Auto Scaling Amazon EC2](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [EnterStandby](#) di Referensi AWS CLI Perintah.

## execute-policy

Contoh kode berikut menunjukkan cara menggunakan `execute-policy`.

### AWS CLI

Untuk menjalankan kebijakan penskalaan

Contoh ini mengeksekusi kebijakan penskalaan yang dinamai `my-step-scale-out-policy` untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling execute-policy \  
  --auto-scaling-group-name my-asg \  
  --policy-name my-step-scale-out-policy \  
  --metric-value 95 \  
  --breach-threshold 80
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah dan kebijakan penskalaan sederhana](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [ExecutePolicy](#) di Referensi AWS CLI Perintah.

## exit-standby

Contoh kode berikut menunjukkan cara menggunakan `exit-standby`.

### AWS CLI

Untuk memindahkan instance keluar dari mode siaga

Contoh ini memindahkan instance yang ditentukan keluar dari mode siaga.

```
aws autoscaling exit-standby \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg
```

Output:



```
{
  "Activities": [
    {
      "ActivityId": "142928e1-a2dc-453a-9b24-b85ad6735928",
      "AutoScalingGroupName": "my-asg",
      "Description": "Moving EC2 instance out of Standby:
i-061c63c5eb45f0416",
      "Cause": "At 2020-10-31T20:32:50Z instance i-061c63c5eb45f0416 was moved
out of standby in response to a user request, increasing the capacity from 0 to
1.",
      "StartTime": "2020-10-31T20:32:50.222Z",
      "StatusCode": "PreInService",
      "Progress": 30,
      "Details": "{\"Subnet ID\":\"subnet-6194ea3b\",\"Availability Zone\":
\"us-west-2c\"}"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menghapus instans sementara dari grup Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [ExitStandby](#) di Referensi AWS CLI Perintah.

## put-lifecycle-hook

Contoh kode berikut menunjukkan cara menggunakan `put-lifecycle-hook`.

### AWS CLI

Contoh 1: Untuk membuat hook siklus hidup

Contoh ini membuat hook siklus hidup yang akan dipanggil pada setiap instance yang baru diluncurkan, dengan batas waktu 4800 detik. Ini berguna untuk menjaga instance dalam status tunggu hingga skrip data pengguna selesai, atau untuk menjalankan fungsi Lambda AWS menggunakan EventBridge

```
aws autoscaling put-lifecycle-hook \
  --auto-scaling-group-name my-asg \
  --lifecycle-hook-name my-launch-hook \
  --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \
  --heartbeat-timeout 4800
```

Perintah ini tidak menghasilkan output. Jika kait siklus hidup dengan nama yang sama sudah ada, itu akan ditimpa oleh kait siklus hidup baru.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 2: Untuk mengirim pesan email Amazon SNS untuk memberi tahu Anda tentang transisi status instans

Contoh ini membuat hook siklus hidup dengan topik Amazon SNS dan peran IAM yang akan digunakan untuk menerima notifikasi saat peluncuran instans.

```
aws autoscaling put-lifecycle-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-hook-name my-launch-hook \  
  --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \  
  --notification-target-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \  
  --role-arn arn:aws:iam::123456789012:role/my-auto-scaling-role
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 3: Untuk mempublikasikan pesan ke antrian Amazon SQS

Contoh ini membuat hook siklus hidup yang menerbitkan pesan dengan metadata ke antrian Amazon SQS yang ditentukan.

```
aws autoscaling put-lifecycle-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-hook-name my-launch-hook \  
  --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \  
  --notification-target-arn arn:aws:sqs:us-west-2:123456789012:my-sqs-queue \  
  --role-arn arn:aws:iam::123456789012:role/my-notification-role \  
  --notification-metadata "SQS message metadata"
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [PutLifecycleHook](#) di Referensi AWS CLI Perintah.

## put-notification-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-notification-configuration`.

### AWS CLI

Untuk menambahkan notifikasi

Contoh ini menambahkan pemberitahuan yang ditentukan ke grup Auto Scaling yang ditentukan.

```
aws autoscaling put-notification-configuration \  
  --auto-scaling-group-name my-asg \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \  
  --notification-type autoscaling:TEST_NOTIFICATION
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendapatkan notifikasi Amazon SNS ketika grup Auto Scaling Anda menskalakan](#) ke dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [PutNotificationConfiguration](#) di Referensi AWS CLI Perintah.

## put-scaling-policy

Contoh kode berikut menunjukkan cara menggunakan `put-scaling-policy`.

### AWS CLI

Untuk menambahkan kebijakan penskalaan pelacakan target ke grup Auto Scaling

`put-scaling-policy` Contoh berikut menerapkan kebijakan penskalaan pelacakan target ke grup Auto Scaling yang ditentukan. Outputnya berisi ARN dan nama dari dua CloudWatch alarm yang dibuat atas nama Anda. Jika kebijakan penskalaan dengan nama yang sama sudah ada, kebijakan penskalaan akan ditimpa oleh kebijakan penskalaan yang baru.

```
aws autoscaling put-scaling-policy --auto-scaling-group-name my-asg \  
  --policy-name alb1000-target-tracking-scaling-policy \  
  --policy-type TargetTrackingScaling \  
  --target-tracking-configuration file://config.json
```

Isi dari `config.json`:

```
{
```

```

    "TargetValue": 1000.0,
    "PredefinedMetricSpecification": {
      "PredefinedMetricType": "ALBRequestCountPerTarget",
      "ResourceLabel": "app/my-alb/778d41231b141a0f/targetgroup/my-alb-target-
group/943f017f100becff"
    }
  }
}

```

### Output:

```

{
  "PolicyARN": "arn:aws:autoscaling:region:account-id:scalingPolicy:228f02c2-
c665-4bfd-aaac-8b04080bea3c:autoScalingGroupName/my-asg:policyName/alb1000-target-
tracking-scaling-policy",
  "Alarms": [
    {
      "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
my-asg-AlarmHigh-fc0e4183-23ac-497e-9992-691c9980c38e",
      "AlarmName": "TargetTracking-my-asg-AlarmHigh-
fc0e4183-23ac-497e-9992-691c9980c38e"
    },
    {
      "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
my-asg-AlarmLow-61a39305-ed0c-47af-bd9e-471a352ee1a2",
      "AlarmName": "TargetTracking-my-asg-AlarmLow-61a39305-ed0c-47af-
bd9e-471a352ee1a2"
    }
  ]
}

```

Untuk contoh lainnya, lihat [Contoh kebijakan penskalaan untuk Antarmuka Baris AWS Perintah \(AWS CLI\)](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [PutScalingPolicy](#) di Referensi AWS CLI Perintah.

## put-scheduled-update-group-action

Contoh kode berikut menunjukkan cara menggunakan `put-scheduled-update-group-action`.

### AWS CLI

Contoh 1: Untuk menambahkan tindakan terjadwal ke grup Auto Scaling

Contoh ini menambahkan tindakan terjadwal yang ditentukan ke grup Auto Scaling yang ditentukan.

```
aws autoscaling put-scheduled-update-group-action \  
  --auto-scaling-group-name my-asg \  
  --scheduled-action-name my-scheduled-action \  
  --start-time "2023-05-12T08:00:00Z" \  
  --min-size 2 \  
  --max-size 6 \  
  --desired-capacity 4
```

Perintah ini tidak menghasilkan output. Jika tindakan terjadwal dengan nama yang sama sudah ada, itu akan ditimpa oleh tindakan terjadwal yang baru.

Untuk contoh lainnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk menentukan jadwal berulang

Contoh ini membuat tindakan terjadwal untuk skala pada jadwal berulang yang dijadwalkan untuk dieksekusi pada pukul 00:30 pada tanggal 1 Januari, Juni, dan Desember setiap tahun.

```
aws autoscaling put-scheduled-update-group-action \  
  --auto-scaling-group-name my-asg \  
  --scheduled-action-name my-recurring-action \  
  --recurrence "30 0 1 1,6,12 *" \  
  --min-size 2 \  
  --max-size 6 \  
  --desired-capacity 4
```

Perintah ini tidak menghasilkan output. Jika tindakan terjadwal dengan nama yang sama sudah ada, itu akan ditimpa oleh tindakan terjadwal yang baru.

Untuk contoh lainnya, lihat [Penskalaan terjadwal](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [PutScheduledUpdateGroupAction](#) di Referensi AWS CLI Perintah.

## put-warm-pool

Contoh kode berikut menunjukkan cara menggunakan `put-warm-pool`.

## AWS CLI

Untuk membuat kolam hangat

Contoh berikut membuat kolam hangat untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling put-warm-pool \  
  --auto-scaling-group-name my-asg \  
  --min-size 2
```

Perintah ini tidak menghasilkan output. Jika kolam hangat sudah ada, itu akan diperbarui.

Untuk informasi selengkapnya, lihat [Kolam hangat untuk Penskalaan Otomatis Amazon EC2 di Panduan Pengguna Auto Scaling Amazon EC2](#).

- Untuk detail API, lihat [PutWarmPool](#) di Referensi AWS CLI Perintah.

## record-lifecycle-action-heartbeat

Contoh kode berikut menunjukkan cara menggunakan `record-lifecycle-action-heartbeat`.

### AWS CLI

Untuk merekam detak jantung aksi siklus hidup

Contoh ini merekam detak jantung aksi siklus hidup untuk menjaga instance dalam status tertunda.

```
aws autoscaling record-lifecycle-action-heartbeat \  
  --lifecycle-hook-name my-launch-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Kait siklus hidup Amazon EC2 Auto Scaling](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [RecordLifecycleActionHeartbeat](#) di Referensi AWS CLI Perintah.

## resume-processes

Contoh kode berikut menunjukkan cara menggunakan `resume-processes`.

## AWS CLI

Untuk melanjutkan proses yang ditangguhkan

Contoh ini melanjutkan proses penskalaan tersuspensi yang ditentukan untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling resume-processes \  
  --auto-scaling-group-name my-asg \  
  --scaling-processes AlarmNotification
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menangguhkan dan melanjutkan proses penskalaan di Panduan Pengguna Auto Scaling](#) Amazon EC2.

- Untuk detail API, lihat [ResumeProcesses](#) di Referensi AWS CLI Perintah.

## rollback-instance-refresh

Contoh kode berikut menunjukkan cara menggunakan `rollback-instance-refresh`.

### AWS CLI

Untuk memutar kembali penyegaran instance

`rollback-instance-refresh` Contoh berikut mengembalikan penyegaran instance yang sedang berlangsung untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling rollback-instance-refresh \  
  --auto-scaling-group-name my-asg
```

Output:

```
{  
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"  
}
```

Untuk informasi selengkapnya, lihat [Membatalkan perubahan dengan rollback](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [RollbackInstanceRefresh](#) di Referensi AWS CLI Perintah.

## set-desired-capacity

Contoh kode berikut menunjukkan cara menggunakan `set-desired-capacity`.

### AWS CLI

Untuk mengatur kapasitas yang diinginkan untuk grup Auto Scaling

Contoh ini menetapkan kapasitas yang diinginkan untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling set-desired-capacity \  
  --auto-scaling-group-name my-asg \  
  --desired-capacity 2 \  
  --honor-cooldown
```

Perintah ini akan kembali ke prompt jika berhasil.

- Untuk detail API, lihat [SetDesiredCapacity](#) di Referensi AWS CLI Perintah.

## set-instance-health

Contoh kode berikut menunjukkan cara menggunakan `set-instance-health`.

### AWS CLI

Untuk mengatur status kesehatan suatu instans

Contoh ini menetapkan status kesehatan dari instance yang ditentukan ke `Unhealthy`.

```
aws autoscaling set-instance-health \  
  --instance-id i-061c63c5eb45f0416 \  
  --health-status Unhealthy
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetInstanceHealth](#) di Referensi AWS CLI Perintah.

## set-instance-protection

Contoh kode berikut menunjukkan cara menggunakan `set-instance-protection`.



## AWS CLI

Contoh 1: Untuk mengaktifkan pengaturan perlindungan instance untuk sebuah instance

Contoh ini memungkinkan perlindungan instance untuk instance tertentu.

```
aws autoscaling set-instance-protection \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg --protected-from-scale-in
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menonaktifkan pengaturan perlindungan instance untuk sebuah instance

Contoh ini menonaktifkan perlindungan instance untuk instance tertentu.

```
aws autoscaling set-instance-protection \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg \  
  --no-protected-from-scale-in
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetInstanceProtection](#) di Referensi AWS CLI Perintah.

## start-instance-refresh

Contoh kode berikut menunjukkan cara menggunakan `start-instance-refresh`.

## AWS CLI

Contoh 1: Untuk memulai penyegaran instance menggunakan parameter baris perintah

`start-instance-refresh` Contoh berikut memulai penyegaran instance menggunakan argumen baris perintah. Parameter opsional `InstanceWarmup` menentukan 60 detik dan `MinHealthyPercentage` 50 persen.

```
aws autoscaling start-instance-refresh \  
  --auto-scaling-group-name my-asg \  
  --preferences '{"InstanceWarmup": 60, "MinHealthyPercentage": 50}'
```

Output:

```
{
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Untuk informasi selengkapnya, lihat [Memulai penyegaran instans](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Contoh 2: Untuk memulai penyegaran instance menggunakan file JSON

`start-instance-refresh` berikut memulai penyegaran instance menggunakan file JSON. Anda dapat menentukan grup Auto Scaling dan menentukan konfigurasi dan preferensi yang Anda inginkan dalam file JSON, seperti yang ditunjukkan pada contoh berikut.

```
aws autoscaling start-instance-refresh \
  --cli-input-json file://config.json
```

Isi dari `config.json`:

```
{
  "AutoScalingGroupName": "my-asg",
  "DesiredConfiguration": {
    "LaunchTemplate": {
      "LaunchTemplateId": "lt-068f72b729example",
      "Version": "$Default"
    }
  },
  "Preferences": {
    "InstanceWarmup": 60,
    "MinHealthyPercentage": 50,
    "AutoRollback": true,
    "ScaleInProtectedInstances": Ignore,
    "StandbyInstances": Terminate
  }
}
```

Output:

```
{
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Untuk informasi selengkapnya, lihat [Memulai penyegaran instans](#) di Panduan Pengguna Auto Scaling Amazon EC2.

- Untuk detail API, lihat [StartInstanceRefresh](#) di Referensi AWS CLI Perintah.

## **suspend-processes**

Contoh kode berikut menunjukkan cara menggunakan `suspend-processes`.

### AWS CLI

Untuk menangguhkan proses Auto Scaling

Contoh ini menangguhkan proses penskalaan yang ditentukan untuk grup Auto Scaling yang ditentukan.

```
aws autoscaling suspend-processes \  
  --auto-scaling-group-name my-asg \  
  --scaling-processes AlarmNotification
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menangguhkan dan melanjutkan proses penskalaan di Panduan Pengguna Auto Scaling](#) Amazon EC2.

- Untuk detail API, lihat [SuspendProcesses](#) di Referensi AWS CLI Perintah.

## **terminate-instance-in-auto-scaling-group**

Contoh kode berikut menunjukkan cara menggunakan `terminate-instance-in-auto-scaling-group`.

### AWS CLI

Untuk mengakhiri instance dalam grup Auto Scaling

Contoh ini mengakhiri instance yang ditentukan dari grup Auto Scaling yang ditentukan tanpa memperbarui ukuran grup. Auto Scaling Amazon EC2 meluncurkan instance pengganti setelah instans yang ditentukan dihentikan.

```
aws autoscaling terminate-instance-in-auto-scaling-group \  
  --instance-id i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg
```

```
--no-should-decrement-desired-capacity
```

Output:

```
{
  "Activities": [
    {
      "ActivityId": "8c35d601-793c-400c-fcd0-f64a27530df7",
      "AutoScalingGroupName": "my-asg",
      "Description": "Terminating EC2 instance: i-061c63c5eb45f0416",
      "Cause": "",
      "StartTime": "2020-10-31T20:34:25.680Z",
      "StatusCode": "InProgress",
      "Progress": 0,
      "Details": "{\"Subnet ID\": \"subnet-6194ea3b\", \"Availability Zone\": \"us-west-2c\"}"
    }
  ]
}
```

- Untuk detail API, lihat [TerminateInstanceInAutoScalingGroup](#) di Referensi AWS CLI Perintah.

## update-auto-scaling-group

Contoh kode berikut menunjukkan cara menggunakan `update-auto-scaling-group`.

### AWS CLI

Contoh 1: Untuk memperbarui batas ukuran grup Auto Scaling

Contoh ini memperbarui grup Auto Scaling yang ditentukan dengan ukuran minimum 2 dan ukuran maksimum 10.

```
aws autoscaling update-auto-scaling-group \
  --auto-scaling-group-name my-asg \
  --min-size 2 \
  --max-size 10
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyetel batas kapasitas untuk grup Auto Scaling di Panduan Pengguna Auto Scaling Amazon EC2](#).

Contoh 2: Untuk menambahkan pemeriksaan kesehatan Elastic Load Balancing dan menentukan Availability Zone dan subnet mana yang akan digunakan

Contoh ini memperbarui grup Auto Scaling yang ditentukan untuk menambahkan pemeriksaan kesehatan Elastic Load Balancing. Perintah ini juga memperbarui nilai `--vpc-zone-identifier` dengan daftar ID subnet di beberapa Availability Zone.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --health-check-type ELB \  
  --health-check-grace-period 600 \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Elastic Load Balancing dan Amazon EC2 Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 3: Untuk memperbarui grup penempatan dan kebijakan penghentian

Contoh ini memperbarui grup penempatan dan kebijakan penghentian yang akan digunakan.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --placement-group my-placement-group \  
  --termination-policies "OldestInstance"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [grup Auto Scaling di Panduan Pengguna Auto Scaling Amazon EC2](#).

Contoh 4: Untuk menggunakan versi terbaru dari template peluncuran

Contoh ini memperbarui grup Auto Scaling yang ditentukan untuk menggunakan versi terbaru dari template peluncuran yang ditentukan.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12,Version='$Latest'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Templat peluncuran](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 5: Untuk menggunakan versi tertentu dari template peluncuran

Contoh ini memperbarui grup Auto Scaling yang ditentukan untuk menggunakan versi tertentu dari template peluncuran, bukan versi terbaru atau default.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateName=my-template-for-auto-scaling,Version='2'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Templat peluncuran](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

Contoh 6: Untuk menentukan kebijakan instans campuran dan mengaktifkan penyeimbangan kembali kapasitas

Contoh ini memperbarui grup Auto Scaling yang ditentukan untuk menggunakan kebijakan instance campuran dan memungkinkan penyeimbangan kembali kapasitas. Struktur ini memungkinkan Anda menentukan grup dengan kapasitas Spot dan Sesuai Permintaan dan menggunakan templat peluncuran yang berbeda untuk arsitektur yang berbeda.

```
aws autoscaling update-auto-scaling-group \  
  --cli-input-json file://~/config.json
```

Isi dari config.json:

```
{  
  "AutoScalingGroupName": "my-asg",  
  "CapacityRebalance": true,  
  "MixedInstancesPolicy": {  
    "LaunchTemplate": {  
      "LaunchTemplateSpecification": {  
        "LaunchTemplateName": "my-launch-template-for-x86",  
        "Version": "$Latest"  
      },  
    },  
    "Overrides": [  
      {  
        "InstanceType": "t3.xlarge",  
        "CapacityType": "OnDemand",  
        "Subnet": "subnet-12345678",  
        "AvailabilityZone": "us-east-1a",  
        "WeightedCapacity": 100  
      },  
      {  
        "InstanceType": "t3.xlarge",  
        "CapacityType": "Spot",  
        "Subnet": "subnet-12345678",  
        "AvailabilityZone": "us-east-1a",  
        "WeightedCapacity": 50  
      },  
      {  
        "InstanceType": "t3.xlarge",  
        "CapacityType": "OnDemand",  
        "Subnet": "subnet-12345678",  
        "AvailabilityZone": "us-east-1b",  
        "WeightedCapacity": 50  
      }  
    ]  
  }  
}
```

```
        "InstanceType": "c6g.large",
        "LaunchTemplateSpecification": {
            "LaunchTemplateName": "my-launch-template-for-arm",
            "Version": "$Latest"
        }
    },
    {
        "InstanceType": "c5.large"
    },
    {
        "InstanceType": "c5a.large"
    }
]
},
"InstancesDistribution": {
    "OnDemandPercentageAboveBaseCapacity": 50,
    "SpotAllocationStrategy": "capacity-optimized"
}
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Grup Auto Scaling dengan beberapa tipe instans dan opsi pembelian](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Untuk detail API, lihat [UpdateAutoScalingGroup](#) di Referensi AWS CLI Perintah.

## Contoh Paket Auto Scaling menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Paket Auto Scaling AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **create-scaling-plan**

Contoh kode berikut menunjukkan cara menggunakan `create-scaling-plan`.

## AWS CLI

Untuk membuat rencana penskalaan

`create-scaling-plan` Contoh berikut membuat rencana penskalaan bernama `my-scaling-plan` menggunakan file JSON yang sudah dibuat ( bernama `config.json`). Struktur rencana penskalaan mencakup instruksi penskalaan untuk grup Auto Scaling bernama `my-asg` Ini menentukan `TagFilters` properti sebagai sumber aplikasi dan memungkinkan penskalaan prediktif dan penskalaan dinamis.

```
aws autoscaling-plans create-scaling-plan \  
  --scaling-plan-name my-scaling-plan \  
  --cli-input-json file://~/config.json
```

Isi `config.json` file:

```
{  
  "ApplicationSource": {  
    "TagFilters": [  
      {  
        "Key": "purpose",  
        "Values": [  
          "my-application"  
        ]  
      }  
    ]  
  },  
  "ScalingInstructions": [  
    {  
      "ServiceNamespace": "autoscaling",  
      "ResourceId": "autoScalingGroup/my-asg",  
      "ScalableDimension": "autoscaling:autoScalingGroup:DesiredCapacity",
```



```

        "ScheduledActionBufferTime": 300,
        "PredictiveScalingMaxCapacityBehavior":
"SetForecastCapacityToMaxCapacity",
        "PredictiveScalingMode": "ForecastAndScale",
        "PredefinedLoadMetricSpecification": {
            "PredefinedLoadMetricType": "ASGTotalCPUUtilization"
        },
        "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
        "MinCapacity": 1,
        "MaxCapacity": 4,
        "TargetTrackingConfigurations": [
            {
                "PredefinedScalingMetricSpecification": {
                    "PredefinedScalingMetricType": "ASGAverageCPUUtilization"
                },
                "TargetValue": 50
            }
        ]
    ]
}

```

Output:

```

{
  "ScalingPlanVersion": 1
}

```

Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS Auto Scaling](#).

- Untuk detail API, lihat [CreateScalingPlan](#) di Referensi AWS CLI Perintah.

## delete-scaling-plan

Contoh kode berikut menunjukkan cara menggunakan `delete-scaling-plan`.

### AWS CLI

Untuk menghapus rencana penskalaan

`delete-scaling-plan` Contoh berikut menghapus rencana penskalaan yang ditentukan.

```
aws autoscaling-plans delete-scaling-plan \
```

```
--scaling-plan-name my-scaling-plan \  
--scaling-plan-version 1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS Auto Scaling](#).

- Untuk detail API, lihat [DeleteScalingPlan](#) di Referensi AWS CLI Perintah.

## describe-scaling-plan-resources

Contoh kode berikut menunjukkan cara menggunakan `describe-scaling-plan-resources`.

### AWS CLI

Untuk mendeskripsikan sumber daya yang dapat diskalakan untuk rencana penskalaan

`describe-scaling-plan-resources` Contoh berikut menampilkan detail tentang sumber daya tunggal yang dapat diskalakan (grup Auto Scaling) yang terkait dengan rencana penskalaan yang ditentukan.

```
aws autoscaling-plans describe-scaling-plan-resources \  
--scaling-plan-name my-scaling-plan \  
--scaling-plan-version 1
```

### Output:

```
{  
  "ScalingPlanResources": [  
    {  
      "ScalableDimension": "autoscaling:autoScalingGroup:DesiredCapacity",  
      "ScalingPlanVersion": 1,  
      "ResourceId": "autoScalingGroup/my-asg",  
      "ScalingStatusCode": "Active",  
      "ScalingStatusMessage": "Target tracking scaling policies have been  
applied to the resource.",  
      "ScalingPolicies": [  
        {  
          "PolicyName": "AutoScaling-my-asg-b1ab65ae-4be3-4634-bd64-  
c7471662b251",  
          "PolicyType": "TargetTrackingScaling",  
          "TargetTrackingConfiguration": {  
            "PredefinedScalingMetricSpecification": {
```

```

        "PredefinedScalingMetricType":
        "ALBRequestCountPerTarget",
        "ResourceLabel": "app/my-alb/f37c06a68c1748aa/
targetgroup/my-target-group/6d4ea56ca2d6a18d"
        },
        "TargetValue": 40.0
    }
}
],
"ServiceNamespace": "autoscaling",
"ScalingPlanName": "my-scaling-plan"
}
]
}

```

Untuk informasi selengkapnya, lihat [Apa itu AWS Auto Scaling?](#) di Panduan Pengguna AWS Auto Scaling.

- Untuk detail API, lihat [DescribeScalingPlanResources](#) di Referensi AWS CLI Perintah.

## describe-scaling-plans

Contoh kode berikut menunjukkan cara menggunakan `describe-scaling-plans`.

### AWS CLI

Untuk menggambarkan rencana penskalaan

`describe-scaling-plans` Contoh berikut menampilkan rincian rencana penskalaan yang ditentukan.

```
aws autoscaling-plans describe-scaling-plans \
--scaling-plan-names scaling-plan-with-asg-and-ddb
```

Output:

```
{
  "ScalingPlans": [
    {
      "LastMutatingRequestTime": 1565388443.963,
      "ScalingPlanVersion": 1,
      "CreationTime": 1565388443.963,
      "ScalingInstructions": [

```

```

        {
            "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
            "ScalableDimension":
"autoscaling:autoScalingGroup:DesiredCapacity",
            "TargetTrackingConfigurations": [
                {
                    "PredefinedScalingMetricSpecification": {
                        "PredefinedScalingMetricType":
"ASGAverageCPUUtilization"
                    },
                    "TargetValue": 50.0,
                    "EstimatedInstanceWarmup": 300,
                    "DisableScaleIn": false
                }
            ],
            "ResourceId": "autoScalingGroup/my-asg",
            "DisableDynamicScaling": false,
            "MinCapacity": 1,
            "ServiceNamespace": "autoscaling",
            "MaxCapacity": 10
        },
        {
            "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
            "ScalableDimension": "dynamodb:table:ReadCapacityUnits",
            "TargetTrackingConfigurations": [
                {
                    "PredefinedScalingMetricSpecification": {
                        "PredefinedScalingMetricType":
"DynamoDBReadCapacityUtilization"
                    },
                    "TargetValue": 50.0,
                    "ScaleInCooldown": 60,
                    "DisableScaleIn": false,
                    "ScaleOutCooldown": 60
                }
            ],
            "ResourceId": "table/my-table",
            "DisableDynamicScaling": false,
            "MinCapacity": 5,
            "ServiceNamespace": "dynamodb",
            "MaxCapacity": 10000
        },
        {
            "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",

```

```

        "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
        "TargetTrackingConfigurations": [
            {
                "PredefinedScalingMetricSpecification": {
                    "PredefinedScalingMetricType":
"DynamoDBWriteCapacityUtilization"
                },
                "TargetValue": 50.0,
                "ScaleInCooldown": 60,
                "DisableScaleIn": false,
                "ScaleOutCooldown": 60
            }
        ],
        "ResourceId": "table/my-table",
        "DisableDynamicScaling": false,
        "MinCapacity": 5,
        "ServiceNamespace": "dynamodb",
        "MaxCapacity": 10000
    }
],
"ApplicationSource": {
    "TagFilters": [
        {
            "Values": [
                "my-application-id"
            ],
            "Key": "application"
        }
    ]
},
"StatusStartTime": 1565388455.836,
"ScalingPlanName": "scaling-plan-with-asg-and-ddb",
"StatusMessage": "Scaling plan has been created and applied to all
resources.",
"StatusCode": "Active"
}
]
}

```

Untuk informasi selengkapnya, lihat [Apa itu AWS Auto Scaling?](#) di Panduan Pengguna AWS Auto Scaling.

- Untuk detail API, lihat [DescribeScalingPlans](#) di Referensi AWS CLI Perintah.

## get-scaling-plan-resource-forecast-data

Contoh kode berikut menunjukkan cara menggunakan `get-scaling-plan-resource-forecast-data`.

### AWS CLI

Untuk mengambil data perkiraan beban

Contoh ini mengambil data perkiraan beban untuk sumber daya yang dapat diskalakan (grup Auto Scaling) yang terkait dengan rencana penskalaan yang ditentukan.

```
aws autoscaling-plans get-scaling-plan-resource-forecast-data \
  --scaling-plan-name my-scaling-plan \
  --scaling-plan-version 1 \
  --service-namespace "autoscaling" \
  --resource-id autoScalingGroup/my-asg \
  --scalable-dimension "autoscaling:autoScalingGroup:DesiredCapacity" \
  --forecast-data-type "LoadForecast" \
  --start-time "2019-08-30T00:00:00Z" \
  --end-time "2019-09-06T00:00:00Z"
```

Output:

```
{
  "Datapoints": [...]
}
```

Untuk informasi selengkapnya, lihat [Apa Itu AWS Auto Scaling di Panduan Pengguna Auto AWS Scaling](#).

- Untuk detail API, lihat [GetScalingPlanResourceForecastData](#) di Referensi AWS CLI Perintah.

## update-scaling-plan

Contoh kode berikut menunjukkan cara menggunakan `update-scaling-plan`.

### AWS CLI

Untuk memperbarui rencana penskalaan

`update-scaling-plan` Contoh berikut memodifikasi metrik penskalaan untuk grup Auto Scaling dalam rencana penskalaan yang ditentukan.

```
aws autoscaling-plans update-scaling-plan \  
  --scaling-plan-name my-scaling-plan \  
  --scaling-plan-version 1 \  
  --scaling-instructions  
  '{"ScalableDimension":"autoscaling:autoScalingGroup:DesiredCapacity","ResourceId":"autoScal  
my-asg","ServiceNamespace":"autoscaling","TargetTrackingConfigurations":  
[{"PredefinedScalingMetricSpecification":  
  {"PredefinedScalingMetricType":"ALBRequestCountPerTarget","ResourceLabel":"app/my-  
alb/f37c06a68c1748aa/targetgroup/my-target-  
group/6d4ea56ca2d6a18d"},"TargetValue":40.0}],"MinCapacity": 1,"MaxCapacity": 10}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Apa itu AWS Auto Scaling?](#) di Panduan Pengguna AWS Auto Scaling.

- Untuk detail API, lihat [UpdateScalingPlan](#) di Referensi AWS CLI Perintah.

## AWS Backup contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Backup.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-backup-plan**

Contoh kode berikut menunjukkan cara menggunakan `create-backup-plan`.

## AWS CLI

Untuk membuat rencana cadangan

`create-backup-plan` Contoh berikut membuat rencana cadangan yang ditentukan dengan retensi 35 hari.

```
aws backup create-backup-plan \  
--backup-plan "{\"BackupPlanName\": \"Example-Backup-Plan\", \"Rules\": [{\"RuleName\":  
\"DailyBackups\", \"ScheduleExpression\": \"cron(0 5 ? * * *)\", \"StartWindowMinutes  
\":480, \"TargetBackupVaultName\": \"Default\", \"Lifecycle\": {\"DeleteAfterDays  
\":35}}]\""
```

Output:

```
{  
  "BackupPlanId": "1fa3895c-a7f5-484a-a371-2dd6a1a9f729",  
  "BackupPlanArn": "arn:aws:backup:us-west-2:123456789012:backup-plan:1fa3895c-  
a7f5-484a-a371-2dd6a1a9f729",  
  "CreationDate": 1568928754.747,  
  "VersionId": "ZjQ2ZTI5YWQtZDg5Yi00MzYzLWJmZTAzMDE1Mzh1MDhjYjEz"  
}
```

Untuk informasi selengkapnya, lihat [Membuat Rencana Cadangan](#) di Panduan Pengembang AWS Cadangan.

- Untuk detail API, lihat [CreateBackupPlan](#) di Referensi AWS CLI Perintah.

## `create-backup-vault`

Contoh kode berikut menunjukkan cara menggunakan `create-backup-vault`.

## AWS CLI

Untuk membuat brankas cadangan

`create-backup-vault` Contoh berikut membuat brankas cadangan dengan nama yang ditentukan.

```
aws backup create-backup-vault
```



```
--backup-vault-name sample-vault
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "BackupVaultName": "sample-vault",
  "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-vault:sample-
vault",
  "CreationDate": 1568928338.385
}
```

Untuk informasi selengkapnya, lihat [Membuat Brankas Cadangan](#) di Panduan Pengembang AWS Cadangan.

- Untuk detail API, lihat [CreateBackupVault](#) di Referensi AWS CLI Perintah.

## get-backup-plan-from-template

Contoh kode berikut menunjukkan cara menggunakan `get-backup-plan-from-template`.

### AWS CLI

Untuk mendapatkan rencana cadangan yang ada dari template

`get-backup-plan-from-template` Contoh berikut mendapatkan rencana cadangan yang ada dari template yang menentukan cadangan harian dengan retensi 35 hari.

```
aws backup get-backup-plan-from-template \
  --backup-plan-template-id "87c0c1ef-254d-4180-8fef-2e76a2c38aaa"
```

Output:

```
{
  "BackupPlanDocument": {
    "Rules": [
      {
        "RuleName": "DailyBackups",
        "ScheduleExpression": "cron(0 5 ? * * *)",
        "StartWindowMinutes": 480,
        "Lifecycle": {
          "DeleteAfterDays": 35
        }
      }
    ]
  }
}
```

```

    }
  }
]
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Rencana Cadangan](#) di Panduan Pengembang AWS Cadangan.

- Untuk detail API, lihat [GetBackupPlanFromTemplate](#) di Referensi AWS CLI Perintah.

## get-backup-plan

Contoh kode berikut menunjukkan cara menggunakan `get-backup-plan`.

### AWS CLI

Untuk mendapatkan rincian rencana cadangan

`get-backup-plan` Contoh berikut menampilkan rincian rencana cadangan yang ditentukan.

```
aws backup get-backup-plan \
  --backup-plan-id "fcbf5d8f-bd77-4f3a-9c97-f24fb3d373a5"
```

Output:

```
{
  "BackupPlan": {
    "BackupPlanName": "Example-Backup-Plan",
    "Rules": [
      {
        "RuleName": "DailyBackups",
        "TargetBackupVaultName": "Default",
        "ScheduleExpression": "cron(0 5 ? * * *)",
        "StartWindowMinutes": 480,
        "CompletionWindowMinutes": 10080,
        "Lifecycle": {
          "DeleteAfterDays": 35
        },
        "RuleId": "70e0ccdc-e9df-4e83-82ad-c1e5a9471cc3"
      }
    ]
  }
}
```

```
  },
  "BackupPlanId": "fcbf5d8f-bd77-4f3a-9c97-f24fb3d373a5",
  "BackupPlanArn": "arn:aws:backup:us-west-2:123456789012:backup-plan:fcbf5d8f-
bd77-4f3a-9c97-f24fb3d373a5",
  "VersionId": "NjQ2ZTZkODktMGVhNy00MmQ0LWE4YjktZTkWNTQ3OTkyYTcw",
  "CreationDate": 1568926091.57
}
```

Untuk informasi selengkapnya, lihat [Membuat Rencana Cadangan](#) di Panduan Pengembang AWS Cadangan.

- Untuk detail API, lihat [GetBackupPlan](#) di Referensi AWS CLI Perintah.

## list-backup-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-backup-jobs`.

### AWS CLI

Contoh 1: Untuk membuat daftar semua pekerjaan cadangan

`list-backup-jobs` Contoh berikut mengembalikan metadata tentang pekerjaan cadangan Anda di akun Anda AWS .

```
aws backup list-backup-jobs
```

Output:

```
{
  "BackupJobs": [
    {
      "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "BackupVaultName": "Default",
      "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-12345678901234567",
      "CreationDate": 1600721892.929,
      "State": "CREATED",
      "PercentDone": "0.0",
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",

```

```

        "StartBy": 1600725492.929,
        "ResourceType": "EC2"
    },
    {
        "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "BackupVaultName": "Default",
        "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
        "RecoveryPointArn": "arn:aws:backup:us-west-2:123456789012:recovery-
point:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "ResourceArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-12345678",
        "CreationDate": 1600721724.77,
        "CompletionDate": 1600721744.488,
        "State": "COMPLETED",
        "PercentDone": "100.0",
        "BackupSizeInBytes": 71,
        "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
        "StartBy": 1600725324.77,
        "ResourceType": "EFS"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Cadangan](#) di Panduan Pengembang AWS Cadangan.

Contoh 2: Untuk daftar pekerjaan cadangan yang telah selesai

`list-backup-jobs` Contoh berikut mengembalikan metadata tentang pekerjaan pencadangan Anda yang telah selesai di akun Anda AWS .

```

aws backup list-backup-jobs \
  --by-state COMPLETED

```

Output:

```

{
  "BackupJobs": [
    {
      "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",

```

```
        "BackupVaultName": "Default",
        "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
        "RecoveryPointArn": "arn:aws:backup:us-west-2:123456789012:recovery-
point:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "ResourceArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-12345678",
        "CreationDate": 1600721724.77,
        "CompletionDate": 1600721744.488,
        "State": "COMPLETED",
        "PercentDone": "100.0",
        "BackupSizeInBytes": 71,
        "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
        "StartBy": 1600725324.77,
        "ResourceType": "EFS"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Membuat Cadangan](#) di Panduan Pengembang AWS Cadangan.

- Untuk detail API, lihat [ListBackupJobs](#) di Referensi AWS CLI Perintah.

## AWS Batch contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS Batch.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **cancel-job**

Contoh kode berikut menunjukkan cara menggunakan `cancel-job`.

#### AWS CLI

Untuk membatalkan pekerjaan

Contoh ini membatalkan pekerjaan dengan ID pekerjaan yang ditentukan.

Perintah:

```
aws batch cancel-job --job-id bcf0b186-a532-4122-842e-2ccab8d54efb --reason
"Cancelling job."
```

- Untuk detail API, lihat [CancelJob](#) di Referensi AWS CLI Perintah.

### **create-compute-environment**

Contoh kode berikut menunjukkan cara menggunakan `create-compute-environment`.

#### AWS CLI

Untuk membuat lingkungan komputasi terkelola dengan instans On-Demand

Contoh ini menciptakan lingkungan komputasi terkelola dengan jenis instans C4 tertentu yang diluncurkan sesuai permintaan. Lingkungan komputasi disebut OnDemand C4.

Perintah:

```
aws batch create-compute-environment --cli-input-json file://<path_to_json_file>/
C4OnDemand.json
```

Format file JSON:

```
{
  "computeEnvironmentName": "C4OnDemand",
  "type": "MANAGED",
  "state": "ENABLED",
  "computeResources": {
```

```
"type": "EC2",
"minvCpus": 0,
"maxvCpus": 128,
"desiredvCpus": 48,
"instanceTypes": [
  "c4.large",
  "c4.xlarge",
  "c4.2xlarge",
  "c4.4xlarge",
  "c4.8xlarge"
],
"subnets": [
  "subnet-220c0e0a",
  "subnet-1a95556d",
  "subnet-978f6dce"
],
"securityGroupIds": [
  "sg-cf5093b2"
],
"ec2KeyPair": "id_rsa",
"instanceRole": "ecsInstanceRole",
"tags": {
  "Name": "Batch Instance - C4OnDemand"
}
},
"serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole"
}
```

#### Output:

```
{
  "computeEnvironmentName": "C4OnDemand",
  "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/C4OnDemand"
}
```

Untuk membuat lingkungan komputasi terkelola dengan Instans Spot

Contoh ini menciptakan lingkungan komputasi terkelola dengan tipe instans M4 yang diluncurkan saat harga tawaran Spot berada pada atau di bawah 20% dari harga Sesuai Permintaan untuk jenis instans. Lingkungan komputasi disebut M4Spot.

Perintah:

```
aws batch create-compute-environment --cli-input-json file://<path_to_json_file>/M4Spot.json
```

### Format file JSON:

```
{
  "computeEnvironmentName": "M4Spot",
  "type": "MANAGED",
  "state": "ENABLED",
  "computeResources": {
    "type": "SPOT",
    "spotIamFleetRole": "arn:aws:iam::012345678910:role/aws-ec2-spot-fleet-role",
    "minvCpus": 0,
    "maxvCpus": 128,
    "desiredvCpus": 4,
    "instanceTypes": [
      "m4"
    ],
    "bidPercentage": 20,
    "subnets": [
      "subnet-220c0e0a",
      "subnet-1a95556d",
      "subnet-978f6dce"
    ],
    "securityGroupIds": [
      "sg-cf5093b2"
    ],
    "ec2KeyPair": "id_rsa",
    "instanceRole": "ecsInstanceRole",
    "tags": {
      "Name": "Batch Instance - M4Spot"
    }
  },
  "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole"
}
```

### Output:

```
{
  "computeEnvironmentName": "M4Spot",
  "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-environment/M4Spot"
```



```
}
```

- Untuk detail API, lihat [CreateComputeEnvironment](#) di Referensi AWS CLI Perintah.

## create-job-queue

Contoh kode berikut menunjukkan cara menggunakan `create-job-queue`.

### AWS CLI

Untuk membuat antrian pekerjaan prioritas rendah dengan satu lingkungan komputasi

Contoh ini membuat antrian pekerjaan yang disebut `LowPriority` yang menggunakan lingkungan komputasi `M4Spot`.

Perintah:

```
aws batch create-job-queue --cli-input-json file://<path_to_json_file>/  
LowPriority.json
```

Format file JSON:

```
{  
  "jobQueueName": "LowPriority",  
  "state": "ENABLED",  
  "priority": 10,  
  "computeEnvironmentOrder": [  
    {  
      "order": 1,  
      "computeEnvironment": "M4Spot"  
    }  
  ]  
}
```

Output:

```
{  
  "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/LowPriority",  
  "jobQueueName": "LowPriority"  
}
```

Untuk membuat antrian pekerjaan prioritas tinggi dengan dua lingkungan komputasi

Contoh ini membuat antrian pekerjaan HighPriority yang disebut yang menggunakan lingkungan OnDemand komputasi C4 dengan urutan 1 dan lingkungan komputasi M4Spot dengan urutan 2. Penjadwal akan mencoba menempatkan pekerjaan di lingkungan OnDemand komputasi C4 terlebih dahulu.

Perintah:

```
aws batch create-job-queue --cli-input-json file://<path_to_json_file>/HighPriority.json
```

Format file JSON:

```
{
  "jobQueueName": "HighPriority",
  "state": "ENABLED",
  "priority": 1,
  "computeEnvironmentOrder": [
    {
      "order": 1,
      "computeEnvironment": "C4OnDemand"
    },
    {
      "order": 2,
      "computeEnvironment": "M4Spot"
    }
  ]
}
```

Output:

```
{
  "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/HighPriority",
  "jobQueueName": "HighPriority"
}
```

- Untuk detail API, lihat [CreateJobQueue](#) di Referensi AWS CLI Perintah.

## delete-compute-environment

Contoh kode berikut menunjukkan cara menggunakan `delete-compute-environment`.

## AWS CLI

Untuk menghapus lingkungan komputasi

Contoh ini menghapus lingkungan OnDemand komputasi P2.

Perintah:

```
aws batch delete-compute-environment --compute-environment P2OnDemand
```

- Untuk detail API, lihat [DeleteComputeEnvironment](#) di Referensi AWS CLI Perintah.

## **delete-job-queue**

Contoh kode berikut menunjukkan cara menggunakan `delete-job-queue`.

### AWS CLI

Untuk menghapus antrian pekerjaan

Contoh ini menghapus antrian pekerjaan GPGPU.

Perintah:

```
aws batch delete-job-queue --job-queue GPGPU
```

- Untuk detail API, lihat [DeleteJobQueue](#) di Referensi AWS CLI Perintah.

## **deregister-job-definition**

Contoh kode berikut menunjukkan cara menggunakan `deregister-job-definition`.

### AWS CLI

Untuk membatalkan pendaftaran definisi pekerjaan

Contoh ini membatalkan pendaftaran definisi pekerjaan yang disebut `sleep10`.

Perintah:

```
aws batch deregister-job-definition --job-definition sleep10
```

- Untuk detail API, lihat [DeregisterJobDefinition](#) di Referensi AWS CLI Perintah.

## describe-compute-environments

Contoh kode berikut menunjukkan cara menggunakan `describe-compute-environments`.

### AWS CLI

Untuk menggambarkan lingkungan komputasi

Contoh ini menjelaskan lingkungan OnDemand komputasi P2.

Perintah:

```
aws batch describe-compute-environments --compute-environments P2OnDemand
```

Output:

```
{
  "computeEnvironments": [
    {
      "status": "VALID",
      "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole",
      "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-environment/P2OnDemand",
      "computeResources": {
        "subnets": [
          "subnet-220c0e0a",
          "subnet-1a95556d",
          "subnet-978f6dce"
        ],
        "tags": {
          "Name": "Batch Instance - P2OnDemand"
        },
        "desiredvCpus": 48,
        "minvCpus": 0,
        "instanceTypes": [
          "p2"
        ],
        "securityGroupIds": [
          "sg-cf5093b2"
        ],
        "instanceRole": "ecsInstanceRole",
```

```

        "maxvCpus": 128,
        "type": "EC2",
        "ec2KeyPair": "id_rsa"
    },
    "statusReason": "ComputeEnvironment Healthy",
    "ecsClusterArn": "arn:aws:ecs:us-east-1:012345678910:cluster/
P20nDemand_Batch_2c06f29d-d1fe-3a49-879d-42394c86effc",
    "state": "ENABLED",
    "computeEnvironmentName": "P20nDemand",
    "type": "MANAGED"
}
]
}

```

- Untuk detail API, lihat [DescribeComputeEnvironments](#) di Referensi AWS CLI Perintah.

## describe-job-definitions

Contoh kode berikut menunjukkan cara menggunakan `describe-job-definitions`.

### AWS CLI

Untuk menggambarkan definisi pekerjaan aktif

Contoh ini menjelaskan semua definisi pekerjaan aktif Anda.

Perintah:

```
aws batch describe-job-definitions --status ACTIVE
```

Output:

```

{
  "jobDefinitions": [
    {
      "status": "ACTIVE",
      "jobDefinitionArn": "arn:aws:batch:us-east-1:012345678910:job-
definition/sleep60:1",
      "containerProperties": {
        "mountPoints": [],
        "parameters": {},
        "image": "busybox",

```

```
        "environment": {},
        "vcpus": 1,
        "command": [
            "sleep",
            "60"
        ],
        "volumes": [],
        "memory": 128,
        "ulimits": []
    },
    "type": "container",
    "jobDefinitionName": "sleep60",
    "revision": 1
}
]
```

- Untuk detail API, lihat [DescribeJobDefinitions](#) di Referensi AWS CLI Perintah.

## describe-job-queues

Contoh kode berikut menunjukkan cara menggunakan `describe-job-queues`.

### AWS CLI

Untuk menggambarkan antrian pekerjaan

Contoh ini menjelaskan antrian HighPriority pekerjaan.

Perintah:

```
aws batch describe-job-queues --job-queues HighPriority
```

Output:

```
{
  "jobQueues": [
    {
      "status": "VALID",
      "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/HighPriority",
      "computeEnvironmentOrder": [
```

```

        {
            "computeEnvironment": "arn:aws:batch:us-
east-1:012345678910:compute-environment/C4OnDemand",
            "order": 1
        }
    ],
    "statusReason": "JobQueue Healthy",
    "priority": 1,
    "state": "ENABLED",
    "jobQueueName": "HighPriority"
}
]
}

```

- Untuk detail API, lihat [DescribeJobQueues](#) di Referensi AWS CLI Perintah.

## describe-jobs

Contoh kode berikut menunjukkan cara menggunakan `describe-jobs`.

### AWS CLI

Untuk menggambarkan pekerjaan

`describe-jobs` Contoh berikut menjelaskan pekerjaan dengan ID pekerjaan yang ditentukan.

```

aws batch describe-jobs \
  --jobs bcf0b186-a532-4122-842e-2ccab8d54efb

```

Output:

```

{
  "jobs": [
    {
      "status": "SUBMITTED",
      "container": {
        "mountPoints": [],
        "image": "busybox",
        "environment": [],
        "vcpus": 1,
        "command": [
          "sleep",

```

```

        "60"
      ],
      "volumes": [],
      "memory": 128,
      "ulimits": []
    },
    "parameters": {},
    "jobDefinition": "arn:aws:batch:us-east-1:012345678910:job-definition/
sleep60:1",
    "jobQueue": "arn:aws:batch:us-east-1:012345678910:job-queue/
HighPriority",
    "jobId": "bcf0b186-a532-4122-842e-2ccab8d54efb",
    "dependsOn": [],
    "jobName": "example",
    "createdAt": 1480483387803
  }
]
}

```

- Untuk detail API, lihat [DescribeJobs](#) di Referensi AWS CLI Perintah.

## list-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-jobs`.

### AWS CLI

Untuk daftar pekerjaan yang sedang berjalan

Contoh ini mencantumkan pekerjaan yang sedang berjalan dalam antrian HighPriority pekerjaan.

Perintah:

```
aws batch list-jobs --job-queue HighPriority
```

Output:

```

{
  "jobSummaryList": [
    {
      "jobName": "example",
      "jobId": "e66ff5fd-a1ff-4640-b1a2-0b0a142f49bb"
    }
  ]
}

```



```
]
}
```

Untuk daftar pekerjaan yang dikirimkan

Contoh ini mencantumkan pekerjaan dalam antrian HighPriority pekerjaan yang berada dalam status pekerjaan YANG DIKIRIMKAN.

Perintah:

```
aws batch list-jobs --job-queue HighPriority --job-status SUBMITTED
```

Output:

```
{
  "jobSummaryList": [
    {
      "jobName": "example",
      "jobId": "68f0c163-fbd4-44e6-9fd1-25b14a434786"
    }
  ]
}
```

- Untuk detail API, lihat [ListJobs](#) di Referensi AWS CLI Perintah.

## **register-job-definition**

Contoh kode berikut menunjukkan cara menggunakan `register-job-definition`.

AWS CLI

Untuk mendaftarkan definisi pekerjaan

Contoh ini mendaftarkan definisi pekerjaan untuk pekerjaan kontainer sederhana.

Perintah:

```
aws batch register-job-definition --job-definition-name sleep30 --type container --
container-properties '{ "image": "busybox", "vcpus": 1, "memory": 128, "command":
  [ "sleep", "30"]}'
```

Output:

```
{
  "jobDefinitionArn": "arn:aws:batch:us-east-1:012345678910:job-definition/
sleep30:1",
  "jobDefinitionName": "sleep30",
  "revision": 1
}
```

- Untuk detail API, lihat [RegisterJobDefinition](#) di Referensi AWS CLI Perintah.

## submit-job

Contoh kode berikut menunjukkan cara menggunakan `submit-job`.

### AWS CLI

Untuk mengirimkan pekerjaan

Contoh ini mengirimkan pekerjaan kontainer sederhana yang disebut contoh ke antrian HighPriority pekerjaan.

Perintah:

```
aws batch submit-job --job-name example --job-queue HighPriority --job-definition
sleep60
```

Output:

```
{
  "jobName": "example",
  "jobId": "876da822-4198-45f2-a252-6cea32512ea8"
}
```

- Untuk detail API, lihat [SubmitJob](#) di Referensi AWS CLI Perintah.

## terminate-job

Contoh kode berikut menunjukkan cara menggunakan `terminate-job`.

### AWS CLI

Untuk mengakhiri pekerjaan

Contoh ini mengakhiri pekerjaan dengan ID pekerjaan yang ditentukan.

Perintah:

```
aws batch terminate-job --job-id 61e743ed-35e4-48da-b2de-5c8333821c84 --reason
"Terminating job."
```

- Untuk detail API, lihat [TerminateJob](#) di Referensi AWS CLI Perintah.

## update-compute-environment

Contoh kode berikut menunjukkan cara menggunakan `update-compute-environment`.

AWS CLI

Untuk memperbarui lingkungan komputasi

Contoh ini menonaktifkan lingkungan OnDemand komputasi P2 sehingga dapat dihapus.

Perintah:

```
aws batch update-compute-environment --compute-environment P2OnDemand --state
DISABLED
```

Output:

```
{
  "computeEnvironmentName": "P2OnDemand",
  "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/P2OnDemand"
}
```

- Untuk detail API, lihat [UpdateComputeEnvironment](#) di Referensi AWS CLI Perintah.

## update-job-queue

Contoh kode berikut menunjukkan cara menggunakan `update-job-queue`.

AWS CLI

Untuk memperbarui antrian pekerjaan

Contoh ini menonaktifkan antrian pekerjaan sehingga dapat dihapus.

Perintah:

```
aws batch update-job-queue --job-queue GPGPU --state DISABLED
```

Output:

```
{
  "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/GPGPU",
  "jobQueueName": "GPGPU"
}
```

- Untuk detail API, lihat [UpdateJobQueue](#) di Referensi AWS CLI Perintah.

## AWS Budgets contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Budgets.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-budget**

Contoh kode berikut menunjukkan cara menggunakan `create-budget`.

## AWS CLI

Untuk membuat anggaran Biaya dan Penggunaan

`create-budget`Perintah berikut membuat anggaran Biaya dan Penggunaan.

```
aws budgets create-budget \  
  --account-id 111122223333 \  
  --budget file://budget.json \  
  --notifications-with-subscribers file://notifications-with-subscribers.json
```

Isi dari `budget.json`:

```
{  
  "BudgetLimit": {  
    "Amount": "100",  
    "Unit": "USD"  
  },  
  "BudgetName": "Example Tag Budget",  
  "BudgetType": "COST",  
  "CostFilters": {  
    "TagKeyValue": [  
      "user:Key$value1",  
      "user:Key$value2"  
    ]  
  },  
  "CostTypes": {  
    "IncludeCredit": true,  
    "IncludeDiscount": true,  
    "IncludeOtherSubscription": true,  
    "IncludeRecurring": true,  
    "IncludeRefund": true,  
    "IncludeSubscription": true,  
    "IncludeSupport": true,  
    "IncludeTax": true,  
    "IncludeUpfront": true,  
    "UseBlended": false  
  },  
  "TimePeriod": {  
    "Start": 1477958399,  
    "End": 3706473600  
  },  
  "TimeUnit": "MONTHLY"  
}
```

```
}

```

Isi dari `notifications-with-subscribers.json`:

```
[
  {
    "Notification": {
      "ComparisonOperator": "GREATER_THAN",
      "NotificationType": "ACTUAL",
      "Threshold": 80,
      "ThresholdType": "PERCENTAGE"
    },
    "Subscribers": [
      {
        "Address": "example@example.com",
        "SubscriptionType": "EMAIL"
      }
    ]
  }
]
```

- Untuk detail API, lihat [CreateBudget](#) di Referensi AWS CLI Perintah.

## **create-notification**

Contoh kode berikut menunjukkan cara menggunakan `create-notification`.

### AWS CLI

Untuk membuat pemberitahuan untuk anggaran Biaya dan Penggunaan yang ditentukan

Contoh ini membuat pemberitahuan untuk anggaran Biaya dan Penggunaan yang ditentukan.

Perintah:

```
aws budgets create-notification --account-id 111122223333 --budget-name "Example
Budget" --notification
NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTAGE
--subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Untuk detail API, lihat [CreateNotification](#) di Referensi AWS CLI Perintah.

## **create-subscriber**

Contoh kode berikut menunjukkan cara menggunakan `create-subscriber`.

### AWS CLI

Untuk membuat pelanggan untuk pemberitahuan yang terkait dengan anggaran Biaya dan Penggunaan

Contoh ini membuat pelanggan untuk pemberitahuan yang ditentukan.

Perintah:

```
aws budgets create-subscriber --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTAGE --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Untuk detail API, lihat [CreateSubscriber](#) di Referensi AWS CLI Perintah.

## **delete-budget**

Contoh kode berikut menunjukkan cara menggunakan `delete-budget`.

### AWS CLI

Untuk menghapus anggaran Biaya dan Penggunaan

Contoh ini menghapus anggaran Biaya dan Penggunaan yang ditentukan.

Perintah:

```
aws budgets delete-budget --account-id 111122223333 --budget-name "Example Budget"
```

- Untuk detail API, lihat [DeleteBudget](#) di Referensi AWS CLI Perintah.

## **delete-notification**

Contoh kode berikut menunjukkan cara menggunakan `delete-notification`.

### AWS CLI

Untuk menghapus pemberitahuan dari anggaran

Contoh ini menghapus pemberitahuan yang ditentukan dari anggaran yang ditentukan.

Perintah:

```
aws budgets delete-notification --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTA
```

- Untuk detail API, lihat [DeleteNotification](#) di Referensi AWS CLI Perintah.

## **delete-subscriber**

Contoh kode berikut menunjukkan cara menggunakan `delete-subscriber`.

AWS CLI

Untuk menghapus pelanggan dari notifikasi

Contoh ini menghapus pelanggan yang ditentukan dari pemberitahuan yang ditentukan.

Perintah:

```
aws budgets delete-subscriber --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTA --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Untuk detail API, lihat [DeleteSubscriber](#) di Referensi AWS CLI Perintah.

## **describe-budget**

Contoh kode berikut menunjukkan cara menggunakan `describe-budget`.

AWS CLI

Untuk mengambil anggaran yang terkait dengan akun

Contoh ini mengambil anggaran Biaya dan Penggunaan yang ditentukan.

Perintah:

```
aws budgets describe-budget --account-id 111122223333 --budget-name "Example Budget"
```



## Output:

```
{
  "Budget": {
    "CalculatedSpend": {
      "ForecastedSpend": {
        "Amount": "2641.548000000000022919266484677791595458984375",
        "Unit": "USD"
      },
      "ActualSpend": {
        "Amount": "604.45600000000000172803993336856365203857421875",
        "Unit": "USD"
      }
    },
    "BudgetType": "COST",
    "BudgetLimit": {
      "Amount": "100",
      "Unit": "USD"
    },
    "BudgetName": "Example Budget",
    "CostTypes": {
      "IncludeOtherSubscription": true,
      "IncludeUpfront": true,
      "IncludeRefund": true,
      "UseBlended": false,
      "IncludeDiscount": true,
      "UseAmortized": false,
      "IncludeTax": true,
      "IncludeCredit": true,
      "IncludeSupport": true,
      "IncludeRecurring": true,
      "IncludeSubscription": true
    },
    "TimeUnit": "MONTHLY",
    "TimePeriod": {
      "Start": 1477958399.0,
      "End": 3706473600.0
    },
    "CostFilters": {
      "AZ": [
        "us-east-1"
      ]
    }
  }
}
```

```
}
```

- Untuk detail API, lihat [DescribeBudget](#) di Referensi AWS CLI Perintah.

## describe-budgets

Contoh kode berikut menunjukkan cara menggunakan `describe-budgets`.

### AWS CLI

Untuk mengambil anggaran yang terkait dengan akun

Contoh ini mengambil anggaran Biaya dan Penggunaan untuk akun.

Perintah:

```
aws budgets describe-budgets --account-id 111122223333 --max-results 20
```

Output:

```
{
  "Budgets": [
    {
      "CalculatedSpend": {
        "ForecastedSpend": {
          "Amount": "2641.548000000000022919266484677791595458984375",
          "Unit": "USD"
        },
        "ActualSpend": {
          "Amount": "604.45600000000000172803993336856365203857421875",
          "Unit": "USD"
        }
      },
      "BudgetType": "COST",
      "BudgetLimit": {
        "Amount": "100",
        "Unit": "USD"
      },
      "BudgetName": "Example Budget",
      "CostTypes": {
        "IncludeOtherSubscription": true,
        "IncludeUpfront": true,

```

```

        "IncludeRefund": true,
        "UseBlended": false,
        "IncludeDiscount": true,
        "UseAmortized": false,
        "IncludeTax": true,
        "IncludeCredit": true,
        "IncludeSupport": true,
        "IncludeRecurring": true,
        "IncludeSubscription": true
    },
    "TimeUnit": "MONTHLY",
    "TimePeriod": {
        "Start": 1477958399.0,
        "End": 3706473600.0
    },
    "CostFilters": {
        "AZ": [
            "us-east-1"
        ]
    }
}
]
}

```

- Untuk detail API, lihat [DescribeBudgets](#) di Referensi AWS CLI Perintah.

## describe-notifications-for-budget

Contoh kode berikut menunjukkan cara menggunakan `describe-notifications-for-budget`.

### AWS CLI

Untuk mengambil notifikasi untuk anggaran

Contoh ini mengambil notifikasi untuk anggaran Biaya dan Penggunaan.

Perintah:

```
aws budgets describe-notifications-for-budget --account-id 111122223333 --budget-name "Example Budget" --max-results 5
```

Output:

```
{
  "Notifications": [
    {
      "Threshold": 80.0,
      "ComparisonOperator": "GREATER_THAN",
      "NotificationType": "ACTUAL"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeNotificationsForBudget](#) di Referensi AWS CLI Perintah.

## describe-subscribers-for-notification

Contoh kode berikut menunjukkan cara menggunakan `describe-subscribers-for-notification`.

### AWS CLI

Untuk mengambil pelanggan untuk pemberitahuan anggaran

Contoh ini mengambil pelanggan untuk pemberitahuan anggaran Biaya dan Penggunaan.

Perintah:

```
aws budgets describe-subscribers-for-notification --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTAGE --max-results 5
```

Output:

```
{
  "Subscribers": [
    {
      "SubscriptionType": "EMAIL",
      "Address": "example2@example.com"
    },
    {
      "SubscriptionType": "EMAIL",
      "Address": "example@example.com"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [DescribeSubscribersForNotification](#) di Referensi AWS CLI Perintah.

## update-budget

Contoh kode berikut menunjukkan cara menggunakan `update-budget`.

### AWS CLI

Untuk mengganti anggaran untuk anggaran Biaya dan Penggunaan

Contoh ini menggantikan anggaran Biaya dan Penggunaan dengan anggaran baru.

Perintah:

```
aws budgets update-budget --account-id 111122223333 --new-budget file://new-budget.json
```

`new-budget.json`:

```
{
  "BudgetLimit": {
    "Amount": "100",
    "Unit": "USD"
  },
  "BudgetName": "Example Budget",
  "BudgetType": "COST",
  "CostFilters": {
    "AZ" : [ "us-east-1" ]
  },
  "CostTypes": {
    "IncludeCredit": false,
    "IncludeDiscount": true,
    "IncludeOtherSubscription": true,
    "IncludeRecurring": true,
    "IncludeRefund": true,
    "IncludeSubscription": true,
    "IncludeSupport": true,
    "IncludeTax": true,
    "IncludeUpfront": true,
  }
}
```

```
    "UseBlended": false,  
    "UseAmortized": true  
  },  
  "TimePeriod": {  
    "Start": 1477958399,  
    "End": 3706473600  
  },  
  "TimeUnit": "MONTHLY"  
}
```

- Untuk detail API, lihat [UpdateBudget](#) di Referensi AWS CLI Perintah.

## update-notification

Contoh kode berikut menunjukkan cara menggunakan `update-notification`.

### AWS CLI

Untuk mengganti notifikasi untuk anggaran Biaya dan Penggunaan

Contoh ini menggantikan notifikasi 80% untuk anggaran Biaya dan Penggunaan dengan notifikasi 90%.

Perintah:

```
aws budgets update-notification --account-id 111122223333 --budget-name "Example  
Budget" --old-notification  
NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTA  
--new-notification  
NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=90,ThresholdType=PERCENTA
```

- Untuk detail API, lihat [UpdateNotification](#) di Referensi AWS CLI Perintah.

## update-subscriber

Contoh kode berikut menunjukkan cara menggunakan `update-subscriber`.

### AWS CLI

Untuk mengganti pelanggan dengan anggaran Biaya dan Penggunaan

Contoh ini menggantikan pelanggan untuk anggaran Biaya dan Penggunaan.

## Perintah:

```
aws budgets update-subscriber --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTAGE --old-subscriber SubscriptionType=EMAIL,Address=example@example.com --new-subscriber SubscriptionType=EMAIL,Address=example2@example.com
```

- Untuk detail API, lihat [UpdateSubscriber](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Chime menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon Chime.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **associate-phone-number-with-user**

Contoh kode berikut menunjukkan cara menggunakan `associate-phone-number-with-user`.

### AWS CLI

Untuk mengaitkan nomor telepon dengan pengguna

`associate-phone-number-with-user` Contoh berikut mengaitkan nomor telepon yang ditentukan dengan pengguna.

```
aws chime associate-phone-number-with-user \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k \  
  --e164-phone-number "+12065550100"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Nomor Telepon Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [AssociatePhoneNumberWithUser](#) di Referensi AWS CLI Perintah.

## **associate-phone-numbers-with-voice-connector-group**

Contoh kode berikut menunjukkan cara menggunakan `associate-phone-numbers-with-voice-connector-group`.

### AWS CLI

Untuk mengaitkan nomor telepon dengan grup Amazon Chime Voice Connector

`associate-phone-numbers-with-voice-connector-group` Contoh berikut mengaitkan nomor telepon yang ditentukan dengan grup Amazon Chime Voice Connector.

```
aws chime associate-phone-numbers-with-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jkl8901 \  
  --e164-phone-numbers "+12065550100" "+12065550101" \  
  --force-associate
```

Output:

```
{  
  "PhoneNumberErrors": []  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup Konektor Suara Amazon Chime di Panduan](#) Administrasi Amazon Chime.

- Untuk detail API, lihat [AssociatePhoneNumbersWithVoiceConnectorGroup](#) di Referensi AWS CLI Perintah.



## **associate-phone-numbers-with-voice-connector**

Contoh kode berikut menunjukkan cara menggunakan `associate-phone-numbers-with-voice-connector`.

### AWS CLI

Untuk mengaitkan nomor telepon dengan Konektor Suara Amazon Chime

`associate-phone-numbers-with-voice-connector` Contoh berikut mengaitkan nomor telepon yang ditentukan dengan Konektor Suara Amazon Chime.

```
aws chime associate-phone-numbers-with-voice-connector \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --e164-phone-numbers "+12065550100" "+12065550101"
  --force-associate
```

Output:

```
{
  "PhoneNumberErrors": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [AssociatePhoneNumbersWithVoiceConnector](#) di Referensi AWS CLI Perintah.

## **associate-signin-delegate-groups-with-account**

Contoh kode berikut menunjukkan cara menggunakan `associate-signin-delegate-groups-with-account`.

### AWS CLI

Untuk mengaitkan grup delegasi masuk

`associate-signin-delegate-groups-with-account` Contoh berikut mengaitkan grup delegasi login yang ditentukan dengan akun Amazon Chime yang ditentukan.

```
aws chime associate-signin-delegate-groups-with-account \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --signin-delegate-groups GroupName=my_users
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Akses Pengguna dan Izin](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [AssociateSigninDelegateGroupsWithAccount](#) di Referensi AWS CLI Perintah.

## batch-create-room-membership

Contoh kode berikut menunjukkan cara menggunakan `batch-create-room-membership`.

### AWS CLI

Untuk membuat beberapa keanggotaan kamar

`batch-create-room-membership` Contoh berikut menambahkan beberapa pengguna ke ruang obrolan sebagai anggota ruang obrolan. Ini juga memberikan peran administrator dan anggota kepada pengguna.

```
aws chime batch-create-room-membership \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \  
  --membership-item-list "MemberId=1ab2345c-67de-8901-  
f23g-45h678901j2k,Role=Administrator" "MemberId=2ab2345c-67de-8901-  
f23g-45h678901j2k,Role=Member"
```

Output:

```
{  
  "ResponseMetadata": {  
    "RequestId": "169ba401-d886-475f-8b3f-e01eac6fadfb",  
    "HTTPStatusCode": 201,  
    "HTTPHeaders": {  
      "x-amzn-requestid": "169ba401-d886-475f-8b3f-e01eac6fadfb",  
      "content-type": "application/json",  
      "content-length": "13",
```

```

        "date": "Mon, 02 Dec 2019 22:46:58 GMT",
        "connection": "keep-alive"
    },
    "RetryAttempts": 0
},
"Errors": []
}

```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [BatchCreateRoomMembership](#) di Referensi AWS CLI Perintah.

## batch-delete-phone-number

Contoh kode berikut menunjukkan cara menggunakan `batch-delete-phone-number`.

### AWS CLI

Untuk menghapus beberapa nomor telepon

`batch-delete-phone-number` Contoh berikut menghapus semua nomor telepon yang ditentukan.

```

aws chime batch-delete-phone-number \
  --phone-number-ids "%2B12065550100" "%2B12065550101"

```

Perintah ini tidak menghasilkan output. Output:

```

{
  "PhoneNumberErrors": []
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [BatchDeletePhoneNumber](#) di Referensi AWS CLI Perintah.

## batch-suspend-user

Contoh kode berikut menunjukkan cara menggunakan `batch-suspend-user`.

## AWS CLI

Untuk menangguhkan beberapa pengguna

`batch-suspend-user` Contoh berikut menangguhkan pengguna yang terdaftar dari akun Amazon Chime yang ditentukan.

```
aws chime batch-suspend-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```

Output:

```
{  
  "UserErrors": []  
}
```

- Untuk detail API, lihat [BatchSuspendUser](#) di Referensi AWS CLI Perintah.

## `batch-unsuspend-user`

Contoh kode berikut menunjukkan cara menggunakan `batch-unsuspend-user`.

## AWS CLI

Untuk membatalkan penangguhan beberapa pengguna

`batch-unsuspend-user` Contoh berikut menghapus penangguhan sebelumnya untuk pengguna yang terdaftar di akun Amazon Chime yang ditentukan.

```
aws chime batch-unsuspend-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```

Output:

```
{  
  "UserErrors": []  
}
```

```
}

```

- Untuk detail API, lihat [BatchUnsuspendUser](#) di Referensi AWS CLI Perintah.

## batch-update-phone-number

Contoh kode berikut menunjukkan cara menggunakan `batch-update-phone-number`.

### AWS CLI

Untuk memperbarui beberapa jenis produk nomor telepon secara bersamaan

`batch-update-phone-number` Contoh berikut memperbarui jenis produk untuk semua nomor telepon yang ditentukan.

```
aws chime batch-update-phone-number \
  --update-phone-number-request-items PhoneNumberId=
  %2B12065550100,ProductType=BusinessCalling PhoneNumberId=
  %2B12065550101,ProductType=BusinessCalling

```

Output:

```
{
  "PhoneNumberErrors": []
}
```

Untuk memperbarui beberapa nama panggilan nomor telepon secara bersamaan

`batch-update-phone-number` Contoh berikut memperbarui nama panggilan untuk semua nomor telepon yang ditentukan.

```
aws chime batch-update-phone-number \
  --update-phone-number-request-items PhoneNumberId=
  %2B14013143874,CallingName=phonenum1 PhoneNumberId=
  %2B14013144061,CallingName=phonenum2

```

Output:

```
{

```

```
"PhoneNumberErrors": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [BatchUpdatePhoneNumber](#) di Referensi AWS CLI Perintah.

## batch-update-user

Contoh kode berikut menunjukkan cara menggunakan `batch-update-user`.

### AWS CLI

Untuk memperbarui beberapa pengguna dalam satu perintah

`batch-update-user` Contoh berikut memperbarui `LicenseType` untuk setiap pengguna yang terdaftar di akun Amazon Chime yang ditentukan.

```
aws chime batch-update-user \
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
  --update-user-request-items "UserId=a1b2c3d4-5678-90ab-
cdef-22222EXAMPLE,LicenseType=Basic" "UserId=a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE,LicenseType=Basic"
```

Output:

```
{
  "UserErrors": []
}
```

- Untuk detail API, lihat [BatchUpdateUser](#) di Referensi AWS CLI Perintah.

## create-account

Contoh kode berikut menunjukkan cara menggunakan `create-account`.

### AWS CLI

Untuk membuat akun

`create-account` Contoh berikut membuat akun Amazon Chime di bawah akun administrator.

## AWS

```
aws chime create-account \  
  --name MyChimeAccount
```

## Output:

```
{  
  "Account": {  
    "AwsAccountId": "111122223333",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "Name": "MyChimeAccount",  
    "AccountType": "Team",  
    "CreatedTimestamp": "2019-01-04T17:11:22.003Z",  
    "DefaultLicense": "Pro",  
    "SupportedLicenses": [  
      "Basic",  
      "Pro"  
    ],  
    "SigninDelegateGroups": [  
      {  
        "GroupName": "myGroup"  
      },  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memulai](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [CreateAccount](#) di Referensi AWS CLI Perintah.

## `create-bot`

Contoh kode berikut menunjukkan cara menggunakan `create-bot`.

## AWS CLI

Untuk membuat bot Amazon Chime

`create-bot` Contoh berikut membuat bot untuk akun Amazon Chime Enterprise yang ditentukan.

```
aws chime create-bot \  
  --name MyChimeAccount
```

```
--account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
--display-name "myBot" \  
--domain "example.com"
```

### Output:

```
{  
  "Bot": {  
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "DisplayName": "myBot (Bot)",  
    "BotType": "ChatBot",  
    "Disabled": false,  
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "BotEmail": "myBot-chimebot@example.com",  
    "SecurityToken": "wJa1rXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan Bot Obrolan dengan Amazon Chime](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [CreateBot](#) di Referensi AWS CLI Perintah.

## create-phone-number-order

Contoh kode berikut menunjukkan cara menggunakan `create-phone-number-order`.

### AWS CLI

Untuk membuat pesanan nomor telepon

`create-phone-number-order` Contoh berikut membuat urutan nomor telepon untuk nomor telepon yang ditentukan.

```
aws chime create-phone-number-order \  
  --product-type VoiceConnector \  
  --e164-phone-numbers "+12065550100" "+12065550101" "+12065550102"
```

### Output:



```
{
  "PhoneNumberOrder": {
    "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
    "ProductType": "VoiceConnector",
    "Status": "Processing",
    "OrderedPhoneNumbers": [
      {
        "E164PhoneNumber": "+12065550100",
        "Status": "Processing"
      },
      {
        "E164PhoneNumber": "+12065550101",
        "Status": "Processing"
      },
      {
        "E164PhoneNumber": "+12065550102",
        "Status": "Processing"
      }
    ],
    "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
    "UpdatedTimestamp": "2019-08-09T21:35:22.408Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [CreatePhoneNumberOrder](#) di Referensi AWS CLI Perintah.

## create-proxy-session

Contoh kode berikut menunjukkan cara menggunakan `create-proxy-session`.

### AWS CLI

Untuk membuat sesi proxy

`create-proxy-session` Contoh berikut membuat sesi proxy dengan kemampuan suara dan SMS.

```
aws chime create-proxy-session \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
```

```
--participant-phone-numbers "+14015550101" "+12065550100" \  
--capabilities "Voice" "SMS"
```

### Output:

```
{  
  "ProxySession": {  
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",  
    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",  
    "Status": "Open",  
    "ExpiryMinutes": 60,  
    "Capabilities": [  
      "SMS",  
      "Voice"  
    ],  
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",  
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",  
    "Participants": [  
      {  
        "PhoneNumber": "+12065550100",  
        "ProxyPhoneNumber": "+19135550199"  
      },  
      {  
        "PhoneNumber": "+14015550101",  
        "ProxyPhoneNumber": "+19135550199"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [CreateProxySession](#) di Referensi AWS CLI Perintah.

## create-room-membership

Contoh kode berikut menunjukkan cara menggunakan `create-room-membership`.

### AWS CLI

Untuk membuat keanggotaan kamar

`create-room-membership` Contoh berikut menambahkan pengguna yang ditentukan ke ruang obrolan sebagai anggota ruang obrolan.

```
aws chime create-room-membership \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \  
  --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Output:

```
{  
  "RoomMembership": {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Member": {  
      "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",  
      "MemberType": "User",  
      "Email": "janed@example.com",  
      "FullName": "Jane Doe",  
      "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"  
    },  
    "Role": "Member",  
    "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "UpdatedTimestamp": "2019-12-02T22:36:41.969Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [CreateRoomMembership](#) di Referensi AWS CLI Perintah.

## **create-room**

Contoh kode berikut menunjukkan cara menggunakan `create-room`.

### AWS CLI

Untuk membuat ruang obrolan

`create-room` Contoh berikut membuat ruang obrolan untuk akun Amazon Chime yang ditentukan.

```
aws chime create-room \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --name chatRoom
```

Output:

```
{  
  "Room": {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Name": "chatRoom",  
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",  
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",  
    "UpdatedTimestamp": "2019-12-02T22:29:31.549Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [CreateRoom](#) di Referensi AWS CLI Perintah.

## create-user

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

### AWS CLI

Untuk membuat profil pengguna untuk perangkat bersama

`create-user` Contoh berikut membuat profil perangkat bersama untuk alamat email yang ditentukan.

```
aws chime create-user \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --email roomdevice@example.com \  
  --user-type SharedDevice
```

Output:

```
{
```

```

"User": {
  "UserId": "1ab2345c-67de-8901-f23g-45h678901j2k",
  "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
  "PrimaryEmail": "roomdevice@example.com",
  "DisplayName": "Room Device",
  "LicenseType": "Pro",
  "UserType": "SharedDevice",
  "UserRegistrationStatus": "Registered",
  "RegisteredOn": "2020-01-15T22:38:09.806Z",
  "AlexaForBusinessMetadata": {
    "IsAlexaForBusinessEnabled": false
  }
}
}

```

Untuk informasi selengkapnya, lihat [Mempersiapkan Pengaturan](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## create-voice-connector-group

Contoh kode berikut menunjukkan cara menggunakan `create-voice-connector-group`.

### AWS CLI

Untuk membuat grup Amazon Chime Voice Connector

`create-voice-connector-group` Contoh berikut membuat grup Amazon Chime Voice Connector yang menyertakan Konektor Suara Amazon Chime yang ditentukan.

```

aws chime create-voice-connector-group \
  --name myGroup \
  --voice-connector-items VoiceConnectorId=abcdef1ghij2klmno3pqr4,Priority=2

```

Output:

```

{
  "VoiceConnectorGroup": {
    "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jk18901",
    "Name": "myGroup",
    "VoiceConnectorItems": [],
  }
}

```

```
    "CreatedTimestamp": "2019-09-18T16:38:34.734Z",  
    "UpdatedTimestamp": "2019-09-18T16:38:34.734Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [CreateVoiceConnectorGroup](#) di Referensi AWS CLI Perintah.

## **create-voice-connector**

Contoh kode berikut menunjukkan cara menggunakan `create-voice-connector`.

### AWS CLI

Untuk membuat Konektor Suara Amazon Chime

`create-voice-connector` Contoh berikut membuat Konektor Suara Amazon Chime di AWS Wilayah tertentu, dengan enkripsi diaktifkan.

```
aws chime create-voice-connector \  
  --name newVoiceConnector \  
  --aws-region us-west-2 \  
  --require-encryption
```

Output:

```
{  
  "VoiceConnector": {  
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",  
    "AwsRegion": "us-west-2",  
    "Name": "newVoiceConnector",  
    "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",  
    "RequireEncryption": true,  
    "CreatedTimestamp": "2019-09-18T20:34:01.352Z",  
    "UpdatedTimestamp": "2019-09-18T20:34:01.352Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [CreateVoiceConnector](#) di Referensi AWS CLI Perintah.

## **delete-account**

Contoh kode berikut menunjukkan cara menggunakan `delete-account`.

### AWS CLI

Untuk menghapus akun

`delete-account` Contoh berikut menghapus akun yang ditentukan.

```
aws chime delete-account --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Akun Anda](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [DeleteAccount](#) di Referensi AWS CLI Perintah.

## **delete-phone-number**

Contoh kode berikut menunjukkan cara menggunakan `delete-phone-number`.

### AWS CLI

Untuk menghapus nomor telepon

`delete-phone-number` Contoh berikut memindahkan nomor telepon yang ditentukan ke dalam antrian penghapusan.

```
aws chime delete-phone-number \  
  --phone-number-id "+12065550100"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [DeletePhoneNumber](#) di Referensi AWS CLI Perintah.

## delete-proxy-session

Contoh kode berikut menunjukkan cara menggunakan `delete-proxy-session`.

### AWS CLI

Untuk menghapus sesi proxy

`delete-proxy-session` Contoh berikut menghapus sesi proxy yang ditentukan.

```
aws chime delete-proxy-session \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --proxy-session-id 123a4bc5-67d8-901e-2f3g-h4ghjk567891
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [DeleteProxySession](#) di Referensi AWS CLI Perintah.

## delete-room-membership

Contoh kode berikut menunjukkan cara menggunakan `delete-room-membership`.

### AWS CLI

Untuk menghapus pengguna sebagai anggota ruang obrolan

`delete-room-membership` Contoh berikut menghapus anggota yang ditentukan dari ruang obrolan yang ditentukan.

```
aws chime delete-room-membership \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \  
  --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [DeleteRoomMembership](#) di Referensi AWS CLI Perintah.



## delete-room

Contoh kode berikut menunjukkan cara menggunakan `delete-room`.

### AWS CLI

Untuk menghapus ruang obrolan

`delete-room` Contoh berikut menghapus ruang obrolan yang ditentukan dan menghapus keanggotaan ruang obrolan.

```
aws chime delete-room \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [DeleteRoom](#) di Referensi AWS CLI Perintah.

## delete-voice-connector-group

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector-group`.

### AWS CLI

judul

`delete-voice-connector-group` Contoh berikut menghapus grup Amazon Chime Voice Connector yang ditentukan.

```
aws chime delete-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jk18901
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Konektor Suara Amazon Chime di Panduan](#) Administrasi Amazon Chime.

- Untuk detail API, lihat [DeleteVoiceConnectorGroup](#) di Referensi AWS CLI Perintah.

## **delete-voice-connector-origination**

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector-origination`.

### AWS CLI

Untuk menghapus pengaturan originasi

`delete-voice-connector-origination` Contoh berikut menghapus host originasi, port, protokol, prioritas, dan bobot dari Konektor Suara Amazon Chime yang ditentukan.

```
aws chime delete-voice-connector-origination \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [DeleteVoiceConnectorOrigination](#) di Referensi AWS CLI Perintah.

## **delete-voice-connector-proxy**

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector-proxy`.

### AWS CLI

Untuk menghapus konfigurasi proxy

`delete-voice-connector-proxy` Contoh berikut menghapus konfigurasi proxy dari Amazon Chime Voice Connector Anda.

```
aws chime delete-voice-connector-proxy \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [DeleteVoiceConnectorProxy](#) di Referensi AWS CLI Perintah.

## **delete-voice-connector-streaming-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector-streaming-configuration`.

### AWS CLI

Untuk menghapus konfigurasi streaming

`delete-voice-connector-streaming-configuration` Contoh berikut menghapus konfigurasi streaming untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime delete-voice-connector-streaming-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Streaming Data Konektor Suara Amazon Chime ke Kinesis](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [DeleteVoiceConnectorStreamingConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-voice-connector-termination-credentials**

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector-termination-credentials`.

### AWS CLI

Untuk menghapus kredensi penghentian

`delete-voice-connector-termination-credentials` Contoh berikut menghapus kredensial penghentian untuk nama pengguna yang ditentukan dan Amazon Chime Voice Connector.

```
aws chime delete-voice-connector-termination-credentials \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --usernames "jdoe"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [DeleteVoiceConnectorTerminationCredentials](#) di Referensi AWS CLI Perintah.

## **delete-voice-connector-termination**

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector-termination`.

### AWS CLI

Untuk menghapus setelan penghentian

`delete-voice-connector-termination` Contoh berikut menghapus setelan penghentian untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime delete-voice-connector-termination \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [DeleteVoiceConnectorTermination](#) di Referensi AWS CLI Perintah.

## **delete-voice-connector**

Contoh kode berikut menunjukkan cara menggunakan `delete-voice-connector`.

### AWS CLI

Untuk menghapus Konektor Suara Amazon Chime

`delete-voice-connector` Contoh berikut melakukan ini

```
aws chime delete-voice-connector \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [DeleteVoiceConnector](#) di Referensi AWS CLI Perintah.

## **disassociate-phone-number-from-user**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-phone-number-from-user`.

### AWS CLI

Untuk memisahkan nomor telepon dari pengguna

`disassociate-phone-number-from-user` Contoh berikut memisahkan nomor telepon dari pengguna yang ditentukan.

```
aws chime disassociate-phone-number-from-user \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Nomor Telepon Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [DisassociatePhoneNumberFromUser](#) di Referensi AWS CLI Perintah.

## **disassociate-phone-numbers-from-voice-connector-group**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-phone-numbers-from-voice-connector-group`.

### AWS CLI

Untuk memisahkan nomor telepon dari grup Amazon Chime Voice Connector

`disassociate-phone-numbers-from-voice-connector-group` Contoh berikut memisahkan nomor telepon yang ditentukan dari grup Amazon Chime Voice Connector.

```
aws chime disassociate-phone-numbers-from-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jk18901 \  
  --phone-number 1234567890
```

```
--e164-phone-numbers "+12065550100" "+12065550101"
```

Output:

```
{
  "PhoneNumberErrors": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [DisassociatePhoneNumbersFromVoiceConnectorGroup](#) di Referensi AWS CLI Perintah.

### **disassociate-phone-numbers-from-voice-connector**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-phone-numbers-from-voice-connector`.

AWS CLI

Untuk memisahkan nomor telepon dari Amazon Chime Voice Connector

`disassociate-phone-numbers-from-voice-connector` Contoh berikut memisahkan nomor telepon yang ditentukan dari Amazon Chime Voice Connector.

```
aws chime disassociate-phone-numbers-from-voice-connector \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --e164-phone-numbers "+12065550100" "+12065550101"
```

Output:

```
{
  "PhoneNumberErrors": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [DisassociatePhoneNumbersFromVoiceConnector](#) di Referensi AWS CLI Perintah.

## **disassociate-signin-delegate-groups-from-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-signin-delegate-groups-from-account`.

### AWS CLI

Untuk memisahkan grup delegasi masuk

`disassociate-signin-delegate-groups-from-account` Contoh berikut memisahkan grup delegasi login yang ditentukan dari akun Amazon Chime yang ditentukan.

```
aws chime disassociate-signin-delegate-groups-from-account \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --group-names "my_users"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Akses Pengguna dan Izin](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [DisassociateSigninDelegateGroupsFromAccount](#) di Referensi AWS CLI Perintah.

## **get-account-settings**

Contoh kode berikut menunjukkan cara menggunakan `get-account-settings`.

### AWS CLI

Untuk mengambil pengaturan untuk akun

`get-account-settings` Contoh berikut mengambil pengaturan akun untuk akun yang ditentukan.

```
aws chime get-account-settings --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{  
  "AccountSettings": {
```

```

    "DisableRemoteControl": false,
    "EnableDialOut": false
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola Akun Amazon Chime](#) Anda di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetAccountSettings](#) di Referensi AWS CLI Perintah.

## get-account

Contoh kode berikut menunjukkan cara menggunakan `get-account`.

### AWS CLI

Untuk mengambil detail untuk akun

`get-account` Contoh berikut mengambil detail untuk akun Amazon Chime yang ditentukan.

```

aws chime get-account \
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE

```

### Output:

```

{
  "Account": {
    "AwsAccountId": "111122223333",
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "Name": "EnterpriseDirectory",
    "AccountType": "EnterpriseDirectory",
    "CreatedTimestamp": "2018-12-20T18:38:02.181Z",
    "DefaultLicense": "Pro",
    "SupportedLicenses": [
      "Basic",
      "Pro"
    ],
    "SigninDelegateGroups": [
      {
        "GroupName": "myGroup"
      }
    ]
  }
}

```



```
}  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Akun Amazon Chime](#) Anda di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetAccount](#) di Referensi AWS CLI Perintah.

## get-bot

Contoh kode berikut menunjukkan cara menggunakan `get-bot`.

### AWS CLI

Untuk mengambil detail tentang bot

`get-bot` Contoh berikut menampilkan rincian untuk bot yang ditentukan.

```
aws chime get-bot \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k
```

Output:

```
{  
  "Bot": {  
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "DisplayName": "myBot (Bot)",  
    "BotType": "ChatBot",  
    "Disabled": false,  
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "BotEmail": "myBot-chimebot@example.com",  
    "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Bot Obrolan](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [GetBot](#) di Referensi AWS CLI Perintah.

## get-global-settings

Contoh kode berikut menunjukkan cara menggunakan `get-global-settings`.

### AWS CLI

Untuk mendapatkan pengaturan global

`get-global-settings` Contoh berikut mengambil nama bucket S3 yang digunakan untuk menyimpan catatan detail panggilan untuk Amazon Chime Business Calling dan Amazon Chime Voice Connectors yang terkait dengan akun administrator. AWS

```
aws chime get-global-settings
```

Output:

```
{
  "BusinessCalling": {
    "CdrBucket": "s3bucket"
  },
  "VoiceConnector": {
    "CdrBucket": "s3bucket"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengaturan Global](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetGlobalSettings](#) di Referensi AWS CLI Perintah.

## get-phone-number-order

Contoh kode berikut menunjukkan cara menggunakan `get-phone-number-order`.

### AWS CLI

Untuk mendapatkan detail untuk pesanan nomor telepon

`get-phone-number-order` Contoh berikut menampilkan rincian urutan nomor telepon yang ditentukan.

```
aws chime get-phone-number-order \
```

```
--phone-number-order-id abc12345-de67-89f0-123g-h45i678j9012
```

Output:

```
{
  "PhoneNumberOrder": {
    "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
    "ProductType": "VoiceConnector",
    "Status": "Partial",
    "OrderedPhoneNumbers": [
      {
        "E164PhoneNumber": "+12065550100",
        "Status": "Acquired"
      },
      {
        "E164PhoneNumber": "+12065550101",
        "Status": "Acquired"
      },
      {
        "E164PhoneNumber": "+12065550102",
        "Status": "Failed"
      }
    ],
    "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
    "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetPhoneNumberOrder](#) di Referensi AWS CLI Perintah.

## get-phone-number-settings

Contoh kode berikut menunjukkan cara menggunakan `get-phone-number-settings`.

### AWS CLI

Untuk mengambil nama panggilan keluar

`get-phone-number-settings` Contoh berikut mengambil nama panggilan keluar default untuk akun pengguna panggilan. AWS

```
aws chime get-phone-number-settings
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "CallingName": "myName",
  "CallingNameUpdatedTimestamp": "2019-10-28T18:56:42.911Z"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetPhoneNumberSettings](#) di Referensi AWS CLI Perintah.

## get-phone-number

Contoh kode berikut menunjukkan cara menggunakan `get-phone-number`.

### AWS CLI

Untuk mendapatkan detail nomor telepon

`get-phone-number` Contoh berikut menampilkan rincian nomor telepon yang ditentukan.

```
aws chime get-phone-number \
  --phone-number-id +12065550100
```

Output:

```
{
  "PhoneNumber": {
    "PhoneNumberId": "%2B12065550100",
    "E164PhoneNumber": "+12065550100",
    "Type": "Local",
    "ProductType": "VoiceConnector",
    "Status": "Unassigned",
    "Capabilities": {
      "InboundCall": true,
      "OutboundCall": true,
      "InboundSMS": true,
      "OutboundSMS": true,
    }
  }
}
```

```

        "InboundMMS": true,
        "OutboundMMS": true
    },
    "Associations": [
        {
            "Value": "abcdef1ghij2klmno3pqr4",
            "Name": "VoiceConnectorId",
            "AssociatedTimestamp": "2019-10-28T18:40:37.453Z"
        }
    ],
    "CallingNameStatus": "UpdateInProgress",
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
    "UpdatedTimestamp": "2019-08-09T21:35:31.745Z"
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetPhoneNumber](#) di Referensi AWS CLI Perintah.

## get-proxy-session

Contoh kode berikut menunjukkan cara menggunakan `get-proxy-session`.

### AWS CLI

Untuk mendapatkan detail sesi proxy

`get-proxy-session` Contoh berikut mencantumkan rincian sesi proxy yang ditentukan.

```

aws chime get-proxy-session \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --proxy-session-id 123a4bc5-67d8-901e-2f3g-h4ghjk567891

```

Output:

```

{
  "ProxySession": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",
    "Status": "Open",
    "ExpiryMinutes": 60,
  }
}

```

```
    "Capabilities": [
      "SMS",
      "Voice"
    ],
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",
    "Participants": [
      {
        "PhoneNumber": "+12065550100",
        "ProxyPhoneNumber": "+19135550199"
      },
      {
        "PhoneNumber": "+14015550101",
        "ProxyPhoneNumber": "+19135550199"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [GetProxySession](#) di Referensi AWS CLI Perintah.

## get-room

Contoh kode berikut menunjukkan cara menggunakan `get-room`.

### AWS CLI

Untuk mendapatkan detail tentang ruang obrolan

`get-room` Contoh berikut menampilkan rincian tentang ruang obrolan yang ditentukan.

```
aws chime get-room \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j
```

Output:

```
{
  "Room": {
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
```

```
    "Name": "chatRoom",
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
    "UpdatedTimestamp": "2019-12-02T22:29:31.549Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [GetRoom](#) di Referensi AWS CLI Perintah.

## get-user-settings

Contoh kode berikut menunjukkan cara menggunakan `get-user-settings`.

### AWS CLI

Untuk mengambil pengaturan pengguna

`get-user-settings` Contoh berikut menampilkan pengaturan pengguna yang ditentukan.

```
aws chime get-user-settings \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Output:

```
{
  "UserSettings": {
    "Telephony": {
      "InboundCalling": true,
      "OutboundCalling": true,
      "SMS": true
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola Nomor Telepon Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetUserSettings](#) di Referensi AWS CLI Perintah.

## get-user

Contoh kode berikut menunjukkan cara menggunakan `get-user`.

### AWS CLI

Untuk mendapatkan detail tentang pengguna

`get-user` Contoh berikut mengambil rincian untuk pengguna tertentu.

```
aws chime get-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Output:

```
{  
  "User": {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "marthar@example.com",  
    "DisplayName": "Martha Rivera",  
    "LicenseType": "Pro",  
    "UserRegistrationStatus": "Registered",  
    "RegisteredOn": "2018-12-20T18:45:25.231Z",  
    "InvitedOn": "2018-12-20T18:45:25.231Z",  
    "AlexaForBusinessMetadata": {  
      "IsAlexaForBusinessEnabled": False,  
      "AlexaForBusinessRoomArn": "null"  
    },  
    "PersonalPIN": "XXXXXXXXXX"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetUser](#) di Referensi AWS CLI Perintah.

## get-voice-connector-group

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-group`.



## AWS CLI

Untuk mendapatkan detail untuk grup Amazon Chime Voice Connector

`get-voice-connector-group` Contoh berikut menampilkan detail untuk grup Amazon Chime Voice Connector yang ditentukan.

```
aws chime get-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jkl8901
```

Output:

```
{  
  "VoiceConnectorGroup": {  
    "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jkl8901",  
    "Name": "myGroup",  
    "VoiceConnectorItems": [],  
    "CreatedTimestamp": "2019-09-18T16:38:34.734Z",  
    "UpdatedTimestamp": "2019-09-18T16:38:34.734Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [GetVoiceConnectorGroup](#) di Referensi AWS CLI Perintah.

## **get-voice-connector-logging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-logging-configuration`.

## AWS CLI

Untuk mendapatkan detail konfigurasi logging

`get-voice-connector-logging-configuration` Contoh berikut mengembalikan detail konfigurasi logging untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime get-voice-connector-logging-configuration \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jkl8901
```

```
--voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```
{
  "LoggingConfiguration": {
    "EnableSIPLogs": true
  }
}
```

Untuk informasi selengkapnya, lihat [Streaming Media Konektor Suara Amazon Chime ke Kinesis](#) di Panduan Administrasi Lonceng Amazon.

- Untuk detail API, lihat [GetVoiceConnectorLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## get-voice-connector-origination

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-origination`.

AWS CLI

Untuk mengambil pengaturan originasi

`get-voice-connector-origination` Contoh berikut mengambil host originasi, port, protokol, prioritas, dan bobot untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime get-voice-connector-origination \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```
{
  "Origination": {
    "Routes": [
      {
        "Host": "10.24.34.0",
        "Port": 1234,
        "Protocol": "TCP",
        "Priority": 1,
        "Weight": 5
      }
    ]
  }
}
```

```
    }
  ],
  "Disabled": false
}
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [GetVoiceConnectorOrigination](#) di Referensi AWS CLI Perintah.

## get-voice-connector-proxy

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-proxy`.

### AWS CLI

Untuk mendapatkan detail konfigurasi proxy

`get-voice-connector-proxy` Contoh berikut mendapatkan detail konfigurasi proxy untuk Konektor Suara Amazon Chime Anda.

```
aws chime get-voice-connector-proxy \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```
{
  "Proxy": {
    "DefaultSessionExpiryMinutes": 60,
    "Disabled": false,
    "PhoneNumberCountries": [
      "US"
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [GetVoiceConnectorProxy](#) di Referensi AWS CLI Perintah.

## get-voice-connector-streaming-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-streaming-configuration`.

### AWS CLI

Untuk mendapatkan detail konfigurasi streaming

`get-voice-connector-streaming-configuration` Contoh berikut mendapatkan detail konfigurasi streaming untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime get-voice-connector-streaming-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```
{  
  "StreamingConfiguration": {  
    "DataRetentionInHours": 24,  
    "Disabled": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Streaming Data Konektor Suara Amazon Chime ke Kinesis](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [GetVoiceConnectorStreamingConfiguration](#) di Referensi AWS CLI Perintah.

## get-voice-connector-termination-health

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-termination-health`.

### AWS CLI

Untuk mengambil rincian kesehatan penghentian

`get-voice-connector-termination-health` Contoh berikut mengambil detail kesehatan penghentian untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime get-voice-connector-termination-health \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```
{  
  "TerminationHealth": {  
    "Timestamp": "Fri Aug 23 16:45:55 UTC 2019",  
    "Source": "10.24.34.0"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [GetVoiceConnectorTerminationHealth](#) di Referensi AWS CLI Perintah.

## get-voice-connector-termination

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector-termination`.

### AWS CLI

Untuk mengambil pengaturan penghentian

`get-voice-connector-termination` Contoh berikut mengambil setelan penghentian untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime get-voice-connector-termination \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output. Output:

```
{  
  "Termination": {  
    "CpsLimit": 1,  
    "DefaultPhoneNumber": "+12065550100",  
    "CallingRegions": [  
      "US"  
    ],  
    "CidrAllowedList": [  

```

```
        "10.24.34.0/23"
      ],
      "Disabled": false
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [GetVoiceConnectorTermination](#) di Referensi AWS CLI Perintah.

## get-voice-connector

Contoh kode berikut menunjukkan cara menggunakan `get-voice-connector`.

### AWS CLI

Untuk mendapatkan detail untuk Konektor Suara Amazon Chime

`get-voice-connector` Contoh berikut menampilkan rincian Konektor Suara Amazon Chime yang ditentukan.

```
aws chime get-voice-connector \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```
{
  "VoiceConnector": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
    "AwsRegion": "us-west-2",
    "Name": "newVoiceConnector",
    "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",
    "RequireEncryption": true,
    "CreatedTimestamp": "2019-09-18T20:34:01.352Z",
    "UpdatedTimestamp": "2019-09-18T20:34:01.352Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [GetVoiceConnector](#) di Referensi AWS CLI Perintah.

## invite-users

Contoh kode berikut menunjukkan cara menggunakan `invite-users`.

### AWS CLI

Untuk mengundang pengguna untuk bergabung dengan Amazon Chime

`invite-users` Contoh berikut mengirimkan email untuk mengundang pengguna ke akun Amazon Chime yang ditentukan.

```
aws chime invite-users \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-email-list "alejandror@example.com" "janed@example.com"
```

Output:

```
{  
  "Invites": [  
    {  
      "InviteId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
      "Status": "Pending",  
      "EmailAddress": "alejandror@example.com",  
      "EmailStatus": "Sent"  
    },  
    {  
      "InviteId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
      "Status": "Pending",  
      "EmailAddress": "janed@example.com",  
      "EmailStatus": "Sent"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengundang dan Menanggihkan Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [InviteUsers](#) di Referensi AWS CLI Perintah.

## list-accounts

Contoh kode berikut menunjukkan cara menggunakan `list-accounts`.

## AWS CLI

Untuk mendapatkan daftar akun

`list-accounts` Contoh berikut mengambil daftar akun Amazon Chime di akun administrator AWS .

```
aws chime list-accounts
```

Output:

```
{
  "Accounts": [
    {
      "AwsAccountId": "111122223333",
      "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "Name": "First Chime Account",
      "AccountType": "EnterpriseDirectory",
      "CreatedTimestamp": "2018-12-20T18:38:02.181Z",
      "DefaultLicense": "Pro",
      "SupportedLicenses": [
        "Basic",
        "Pro"
      ],
      "SigninDelegateGroups": [
        {
          "GroupName": "myGroup"
        }
      ]
    },
    {
      "AwsAccountId": "111122223333",
      "AccountId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "Name": "Second Chime Account",
      "AccountType": "Team",
      "CreatedTimestamp": "2018-09-04T21:44:22.292Z",
      "DefaultLicense": "Pro",
      "SupportedLicenses": [
        "Basic",
        "Pro"
      ],
      "SigninDelegateGroups": [
        {
```



```

    "GroupName": "myGroup"
  },
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola Akun Amazon Chime](#) Anda di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [ListAccounts](#) di Referensi AWS CLI Perintah.

## list-bots

Contoh kode berikut menunjukkan cara menggunakan `list-bots`.

### AWS CLI

Untuk mengambil daftar bot

`list-bots` Contoh berikut mencantumkan bot yang terkait dengan akun Amazon Chime Enterprise yang ditentukan.

```

aws chime list-bots \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45

```

Output:

```

{
  "Bot": {
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "DisplayName": "myBot (Bot)",
    "BotType": "ChatBot",
    "Disabled": false,
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
    "BotEmail": "myBot-chimebot@example.com",
    "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
  }
}

```

Untuk informasi selengkapnya, lihat [Menggunakan Bot Obrolan dengan Amazon Chime](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [ListBots](#) di Referensi AWS CLI Perintah.

## **list-phone-number-orders**

Contoh kode berikut menunjukkan cara menggunakan `list-phone-number-orders`.

### AWS CLI

Untuk membuat daftar pesanan nomor telepon

`list-phone-number-orders` Contoh berikut mencantumkan urutan nomor telepon yang terkait dengan akun administrator Amazon Chime.

```
aws chime list-phone-number-orders
```

Output:

```
{
  "PhoneNumberOrders": [
    {
      "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
      "ProductType": "VoiceConnector",
      "Status": "Partial",
      "OrderedPhoneNumbers": [
        {
          "E164PhoneNumber": "+12065550100",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550101",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550102",
          "Status": "Failed"
        }
      ]
    },
    {
      "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
      "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
    }
  ]
}
```

```

    }
    {
      "PhoneNumberOrderId": "cba54321-ed76-09f5-321g-h54i876j2109",
      "ProductType": "BusinessCalling",
      "Status": "Partial",
      "OrderedPhoneNumbers": [
        {
          "E164PhoneNumber": "+12065550103",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550104",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550105",
          "Status": "Failed"
        }
      ],
      "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
      "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [ListPhoneNumberOrders](#) di Referensi AWS CLI Perintah.

## list-phone-numbers

Contoh kode berikut menunjukkan cara menggunakan `list-phone-numbers`.

### AWS CLI

Untuk mencantumkan nomor telepon untuk akun Amazon Chime

`list-phone-numbers` Contoh berikut mencantumkan nomor telepon yang terkait dengan akun Amazon Chime administrator.

```
aws chime list-phone-numbers
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "PhoneNumbers": [
    {
      "PhoneNumberId": "%2B12065550100",
      "E164PhoneNumber": "+12065550100",
      "Type": "Local",
      "ProductType": "VoiceConnector",
      "Status": "Assigned",
      "Capabilities": {
        "InboundCall": true,
        "OutboundCall": true,
        "InboundSMS": true,
        "OutboundSMS": true,
        "InboundMMS": true,
        "OutboundMMS": true
      },
      "Associations": [
        {
          "Value": "abcdef1ghij2klmno3pqr4",
          "Name": "VoiceConnectorId",
          "AssociatedTimestamp": "2019-10-28T18:40:37.453Z"
        }
      ],
      "CallingNameStatus": "UpdateInProgress",
      "CreatedTimestamp": "2019-08-12T22:10:20.521Z",
      "UpdatedTimestamp": "2019-10-28T18:42:07.964Z"
    },
    {
      "PhoneNumberId": "%2B12065550101",
      "E164PhoneNumber": "+12065550101",
      "Type": "Local",
      "ProductType": "VoiceConnector",
      "Status": "Assigned",
      "Capabilities": {
        "InboundCall": true,
        "OutboundCall": true,
        "InboundSMS": true,
        "OutboundSMS": true,
        "InboundMMS": true,
        "OutboundMMS": true
      },
      "Associations": [
```

```

        {
            "Value": "abcdef1ghij2klmno3pqr4",
            "Name": "VoiceConnectorId",
            "AssociatedTimestamp": "2019-10-28T18:40:37.511Z"
        }
    ],
    "CallingNameStatus": "UpdateInProgress",
    "CreatedTimestamp": "2019-08-12T22:10:20.521Z",
    "UpdatedTimestamp": "2019-10-28T18:42:07.960Z"
}
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [ListPhoneNumbers](#) di Referensi AWS CLI Perintah.

## list-proxy-sessions

Contoh kode berikut menunjukkan cara menggunakan `list-proxy-sessions`.

### AWS CLI

Untuk membuat daftar sesi proxy

`list-proxy-sessions` Contoh berikut mencantumkan sesi proxy untuk Konektor Suara Amazon Chime Anda.

```
aws chime list-proxy-sessions \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Output:

```

{
  "ProxySession": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",
    "Status": "Open",
    "ExpiryMinutes": 60,
    "Capabilities": [
      "SMS",
      "Voice"
    ]
  }
}

```

```

    ],
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",
    "Participants": [
      {
        "PhoneNumber": "+12065550100",
        "ProxyPhoneNumber": "+19135550199"
      },
      {
        "PhoneNumber": "+14015550101",
        "ProxyPhoneNumber": "+19135550199"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [ListProxySessions](#) di Referensi AWS CLI Perintah.

## list-room-memberships

Contoh kode berikut menunjukkan cara menggunakan `list-room-memberships`.

### AWS CLI

Untuk daftar keanggotaan kamar

`list-room-memberships` Contoh berikut menampilkan daftar rincian keanggotaan untuk chat room yang ditentukan.

```

aws chime list-room-memberships \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j

```

Output:

```

{
  "RoomMemberships": [
    {
      "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
      "Member": {

```

```

        "MemberId": "2ab2345c-67de-8901-f23g-45h678901j2k",
        "MemberType": "User",
        "Email": "zhangw@example.com",
        "FullName": "Zhang Wei",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
    },
    "Role": "Member",
    "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",
    "UpdatedTimestamp": "2019-12-02T22:46:58.532Z"
},
{
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
    "Member": {
        "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",
        "MemberType": "User",
        "Email": "janed@example.com",
        "FullName": "Jane Doe",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
    },
    "Role": "Administrator",
    "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",
    "UpdatedTimestamp": "2019-12-02T22:46:58.532Z"
}
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [ListRoomMemberships](#) di Referensi AWS CLI Perintah.

## list-rooms

Contoh kode berikut menunjukkan cara menggunakan `list-rooms`.

### AWS CLI

Untuk daftar ruang obrolan

`list-rooms` Contoh berikut menampilkan daftar ruang obrolan di akun yang ditentukan. Daftar difilter hanya ke ruang obrolan yang dimiliki anggota tertentu.

```
aws chime list-rooms \
```

```
--account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
--member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

### Output:

```
{  
  "Room": {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Name": "teamRoom",  
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",  
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",  
    "UpdatedTimestamp": "2019-12-02T22:33:19.310Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [ListRooms](#) di Referensi AWS CLI Perintah.

## list-users

Contoh kode berikut menunjukkan cara menggunakan `list-users`.

### AWS CLI

Untuk mencantumkan pengguna di akun

`list-users` Contoh berikut mencantumkan pengguna untuk akun Amazon Chime yang ditentukan.

```
aws chime list-users --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

### Output:

```
{  
  "Users": [  
    {  
      "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
      "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "PrimaryEmail": "mariag@example.com",  
    }  
  ]  
}
```



```
    "DisplayName": "Maria Garcia",
    "LicenseType": "Pro",
    "UserType": "PrivateUser",
    "UserRegistrationStatus": "Registered",
    "RegisteredOn": "2018-12-20T18:45:25.231Z"
    "AlexaForBusinessMetadata": {
      "IsAlexaForBusinessEnabled": false
    }
  },
  {
    "UserId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "PrimaryEmail": "richardr@example.com",
    "DisplayName": "Richard Roe",
    "LicenseType": "Pro",
    "UserType": "PrivateUser",
    "UserRegistrationStatus": "Registered",
    "RegisteredOn": "2018-12-20T18:45:45.415Z"
    "AlexaForBusinessMetadata": {
      "IsAlexaForBusinessEnabled": false
    }
  },
  {
    "UserId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "PrimaryEmail": "saanvis@example.com",
    "DisplayName": "Saanvi Sarkar",
    "LicenseType": "Basic",
    "UserType": "PrivateUser",
    "UserRegistrationStatus": "Registered",
    "RegisteredOn": "2018-12-20T18:46:57.747Z"
    "AlexaForBusinessMetadata": {
      "IsAlexaForBusinessEnabled": false
    }
  },
  {
    "UserId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE",
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "PrimaryEmail": "wxiulan@example.com",
    "DisplayName": "Wang Xiulan",
    "LicenseType": "Basic",
    "UserType": "PrivateUser",
    "UserRegistrationStatus": "Registered",
    "RegisteredOn": "2018-12-20T18:47:15.390Z"
```

```
        "AlexaForBusinessMetadata": {
            "IsAlexaForBusinessEnabled": false
        }
    ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [ListUsers](#) di Referensi AWS CLI Perintah.

## list-voice-connector-groups

Contoh kode berikut menunjukkan cara menggunakan `list-voice-connector-groups`.

### AWS CLI

Untuk mencantumkan grup Amazon Chime Voice Connector untuk akun Amazon Chime

`list-voice-connector-groups` Contoh berikut mencantumkan grup Amazon Chime Voice Connector yang terkait dengan akun Amazon Chime administrator.

```
aws chime list-voice-connector-groups
```

Output:

```
{
  "VoiceConnectorGroups": [
    {
      "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jk18901",
      "Name": "myGroup",
      "VoiceConnectorItems": [],
      "CreatedTimestamp": "2019-09-18T16:38:34.734Z",
      "UpdatedTimestamp": "2019-09-18T16:38:34.734Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup Konektor Suara Amazon Chime di Panduan](#) Administrasi Amazon Chime.

- Untuk detail API, lihat [ListVoiceConnectorGroups](#) di Referensi AWS CLI Perintah.

## **list-voice-connector-termination-credentials**

Contoh kode berikut menunjukkan cara menggunakan `list-voice-connector-termination-credentials`.

### AWS CLI

Untuk mengambil daftar kredensi penghentian

`list-voice-connector-termination-credentials` Contoh berikut mengambil daftar kredensial penghentian untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime list-voice-connector-termination-credentials \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "Usernames": [
    "jdoe"
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [ListVoiceConnectorTerminationCredentials](#) di Referensi AWS CLI Perintah.

## **list-voice-connectors**

Contoh kode berikut menunjukkan cara menggunakan `list-voice-connectors`.

### AWS CLI

Untuk mencantumkan Konektor Suara Amazon Chime untuk sebuah akun

`list-voice-connectors` Contoh berikut mencantumkan Konektor Suara Amazon Chime yang terkait dengan akun pemanggil.

```
aws chime list-voice-connectors
```

**Output:**

```
{
  "VoiceConnectors": [
    {
      "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
      "AwsRegion": "us-east-1",
      "Name": "MyVoiceConnector",
      "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",
      "RequireEncryption": true,
      "CreatedTimestamp": "2019-06-04T18:46:56.508Z",
      "UpdatedTimestamp": "2019-09-18T16:33:00.806Z"
    },
    {
      "VoiceConnectorId": "cbadef1ghij2klmno3pqr5",
      "AwsRegion": "us-west-2",
      "Name": "newVoiceConnector",
      "OutboundHostName": "cbadef1ghij2klmno3pqr5.voiceconnector.chime.aws",
      "RequireEncryption": true,
      "CreatedTimestamp": "2019-09-18T20:34:01.352Z",
      "UpdatedTimestamp": "2019-09-18T20:34:01.352Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [ListVoiceConnectors](#) di Referensi AWS CLI Perintah.

**logout-user**

Contoh kode berikut menunjukkan cara menggunakan `logout-user`.

**AWS CLI**

Untuk keluar dari pengguna

`logout-user` Contoh berikut log out pengguna tertentu.

```
aws chime logout-user \
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
```

```
--user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [LogoutUser](#) di Referensi AWS CLI Perintah.

## **put-voice-connector-logging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-voice-connector-logging-configuration`.

### AWS CLI

Untuk menambahkan konfigurasi logging untuk Amazon Chime Voice Connector

`put-voice-connector-logging-configuration` Contoh berikut mengaktifkan konfigurasi logging SIP untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime put-voice-connector-logging-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --logging-configuration EnableSIPLogs=true
```

Output:

```
{  
  "LoggingConfiguration": {  
    "EnableSIPLogs": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Streaming Media Konektor Suara Amazon Chime ke Kinesis](#) di Panduan Administrasi Lonceng Amazon.

- Untuk detail API, lihat [PutVoiceConnectorLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## **put-voice-connector-origination**

Contoh kode berikut menunjukkan cara menggunakan `put-voice-connector-origination`.

### AWS CLI

Untuk mengatur pengaturan originasi

`put-voice-connector-origination` Contoh berikut mengatur host originasi, port, protokol, prioritas, dan bobot untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime put-voice-connector-origination \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --origination
Routes=[{"Host":"10.24.34.0",Port=1234,Protocol="TCP",Priority=1,Weight=5}],Disabled=false
```

Output:

```
{
  "Origination": {
    "Routes": [
      {
        "Host": "10.24.34.0",
        "Port": 1234,
        "Protocol": "TCP",
        "Priority": 1,
        "Weight": 5
      }
    ],
    "Disabled": false
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [PutVoiceConnectorOrigination](#) di Referensi AWS CLI Perintah.

## **put-voice-connector-proxy**

Contoh kode berikut menunjukkan cara menggunakan `put-voice-connector-proxy`.

### AWS CLI

Untuk menempatkan konfigurasi proxy

`put-voice-connector-proxy` Contoh berikut menetapkan konfigurasi proxy ke Amazon Chime Voice Connector Anda.

```
aws chime put-voice-connector-proxy \
```

```
--voice-connector-id abcdef1ghij2klmno3pqr4 \  
--default-session-expiry-minutes 60 \  
--phone-number-pool-countries "US"
```

Output:

```
{  
  "Proxy": {  
    "DefaultSessionExpiryMinutes": 60,  
    "Disabled": false,  
    "PhoneNumberCountries": [  
      "US"  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [PutVoiceConnectorProxy](#) di Referensi AWS CLI Perintah.

## put-voice-connector-streaming-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-voice-connector-streaming-configuration`.

AWS CLI

Untuk membuat konfigurasi streaming

`put-voice-connector-streaming-configuration` Contoh berikut membuat konfigurasi streaming untuk Konektor Suara Amazon Chime yang ditentukan. Ini memungkinkan streaming media dari Amazon Chime Voice Connector ke Amazon Kinesis, dan menetapkan periode retensi data menjadi 24 jam.

```
aws chime put-voice-connector-streaming-configuration \  
--voice-connector-id abcdef1ghij2klmno3pqr4 \  
--streaming-configuration DataRetentionInHours=24,Disabled=false
```

Output:

```
{
  "StreamingConfiguration": {
    "DataRetentionInHours": 24,
    "Disabled": false
  }
}
```

Untuk informasi selengkapnya, lihat [Streaming Data Konektor Suara Amazon Chime ke Kinesis](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [PutVoiceConnectorStreamingConfiguration](#) di Referensi AWS CLI Perintah.

## **put-voice-connector-termination-credentials**

Contoh kode berikut menunjukkan cara menggunakan `put-voice-connector-termination-credentials`.

AWS CLI

Untuk mengatur kredensi penghentian

`put-voice-connector-termination-credentials` Contoh berikut menetapkan kredensial penghentian untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime put-voice-connector-termination-credentials \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --credentials Username="jdoe",Password="XXXXXXXX"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [PutVoiceConnectorTerminationCredentials](#) di Referensi AWS CLI Perintah.

## **put-voice-connector-termination**

Contoh kode berikut menunjukkan cara menggunakan `put-voice-connector-termination`.



## AWS CLI

Untuk mengatur pengaturan penghentian

`put-voice-connector-termination` Contoh berikut menetapkan wilayah panggilan dan mengizinkan pengaturan penghentian host IP untuk Konektor Suara Amazon Chime yang ditentukan.

```
aws chime put-voice-connector-termination \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --termination CallingRegions="US",CidrAllowedList="10.24.34.0/23",Disabled=false
```

Output:

```
{  
  "Termination": {  
    "CpsLimit": 0,  
    "CallingRegions": [  
      "US"  
    ],  
    "CidrAllowedList": [  
      "10.24.34.0/23"  
    ],  
    "Disabled": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [PutVoiceConnectorTermination](#) di Referensi AWS CLI Perintah.

## **regenerate-security-token**

Contoh kode berikut menunjukkan cara menggunakan `regenerate-security-token`.

## AWS CLI

Untuk membuat ulang token keamanan

`regenerate-security-token` Contoh berikut meregenerasi token keamanan untuk bot yang ditentukan.

```
aws chime regenerate-security-token \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k
```

Output:

```
{  
  "Bot": {  
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "DisplayName": "myBot (Bot)",  
    "BotType": "ChatBot",  
    "Disabled": false,  
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "BotEmail": "myBot-chimebot@example.com",  
    "SecurityToken": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Permintaan Bot Obrolan](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [RegenerateSecurityToken](#) di Referensi AWS CLI Perintah.

## reset-personal-pin

Contoh kode berikut menunjukkan cara menggunakan `reset-personal-pin`.

### AWS CLI

Untuk mengatur ulang PIN rapat pribadi pengguna

`reset-personal-pin` Contoh berikut mengatur ulang PIN rapat pribadi pengguna yang ditentukan.

```
aws chime reset-personal-pin \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Output:

```
{
  "User": {
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "PrimaryEmail": "mateo@example.com",
    "DisplayName": "Mateo Jackson",
    "LicenseType": "Pro",
    "UserType": "PrivateUser",
    "UserRegistrationStatus": "Registered",
    "RegisteredOn": "2018-12-20T18:45:25.231Z",
    "AlexaForBusinessMetadata": {
      "IsAlexaForBusinessEnabled": False,
      "AlexaForBusinessRoomArn": "null"
    },
    "PersonalPIN": "XXXXXXXXXX"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengubah PIN Rapat Pribadi](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [ResetPersonalPin](#) di Referensi AWS CLI Perintah.

## restore-phone-number

Contoh kode berikut menunjukkan cara menggunakan `restore-phone-number`.

### AWS CLI

Untuk mengembalikan nomor telepon

`restore-phone-number` Contoh berikut mengembalikan nomor telepon yang ditentukan dari antrian penghapusan.

```
aws chime restore-phone-number \
  --phone-number-id "+12065550100"
```

Output:

```
{
  "PhoneNumber": {
    "PhoneNumberId": "%2B12065550100",
```

```

    "E164PhoneNumber": "+12065550100",
    "Type": "Local",
    "ProductType": "BusinessCalling",
    "Status": "Unassigned",
    "Capabilities": {
      "InboundCall": true,
      "OutboundCall": true,
      "InboundSMS": true,
      "OutboundSMS": true,
      "InboundMMS": true,
      "OutboundMMS": true
    },
    "Associations": [],
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
    "UpdatedTimestamp": "2019-08-12T22:06:36.355Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [RestorePhoneNumber](#) di Referensi AWS CLI Perintah.

## search-available-phone-numbers

Contoh kode berikut menunjukkan cara menggunakan `search-available-phone-numbers`.

### AWS CLI

Untuk mencari nomor telepon yang tersedia

`search-available-phone-numbers` Contoh berikut mencari nomor telepon yang tersedia berdasarkan kode area.

```
aws chime search-available-phone-numbers \
  --area-code "206"
```

Output:

```
{
  "E164PhoneNumbers": [
    "+12065550100",

```

```
    "+12065550101",  
    "+12065550102",  
    "+12065550103",  
    "+12065550104",  
    "+12065550105",  
    "+12065550106",  
    "+12065550107",  
    "+12065550108",  
    "+12065550109",  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [SearchAvailablePhoneNumbers](#) di Referensi AWS CLI Perintah.

## update-account-settings

Contoh kode berikut menunjukkan cara menggunakan `update-account-settings`.

### AWS CLI

Untuk memperbarui pengaturan untuk akun Anda

`update-account-settings` Contoh berikut menonaktifkan remote control layar bersama untuk akun Amazon Chime yang ditentukan.

```
aws chime update-account-settings \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --account-settings DisableRemoteControl=true
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateAccountSettings](#) di Referensi AWS CLI Perintah.

## update-account

Contoh kode berikut menunjukkan cara menggunakan `update-account`.

### AWS CLI

Untuk memperbarui akun

`update-account` Contoh berikut memperbarui nama akun yang ditentukan.

```
aws chime update-account \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --name MyAccountName
```

Output:

```
{  
  "Account": {  
    "AwsAccountId": "111122223333",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "Name": "MyAccountName",  
    "AccountType": "Team",  
    "CreatedTimestamp": "2018-09-04T21:44:22.292Z",  
    "DefaultLicense": "Pro",  
    "SupportedLicenses": [  
      "Basic",  
      "Pro"  
    ],  
    "SigninDelegateGroups": [  
      {  
        "GroupName": "myGroup"  
      },  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengganti Nama Akun Anda](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [UpdateAccount](#) di Referensi AWS CLI Perintah.

## update-bot

Contoh kode berikut menunjukkan cara menggunakan `update-bot`.

### AWS CLI

Untuk memperbarui bot

`update-bot` Contoh berikut memperbarui status bot yang ditentukan untuk menghentikannya berjalan.

```
aws chime update-bot \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k \
  --disabled
```

### Output:

```
{
  "Bot": {
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "DisplayName": "myBot (Bot)",
    "BotType": "ChatBot",
    "Disabled": true,
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
    "BotEmail": "myBot-chimebot@example.com",
    "SecurityToken": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Bot Obrolan](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [UpdateBot](#) di Referensi AWS CLI Perintah.

## update-global-settings

Contoh kode berikut menunjukkan cara menggunakan `update-global-settings`.

### AWS CLI

Untuk memperbarui pengaturan global

`update-global-settings` Contoh berikut memperbarui bucket S3 yang digunakan untuk menyimpan catatan detail panggilan untuk Amazon Chime Business Calling dan Amazon Chime Voice Connectors yang terkait dengan akun administrator. AWS

```
aws chime update-global-settings \
  --business-calling CdrBucket="s3bucket" \
  --voice-connector CdrBucket="s3bucket"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Pengaturan Global](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [UpdateGlobalSettings](#) di Referensi AWS CLI Perintah.

## **update-phone-number-settings**

Contoh kode berikut menunjukkan cara menggunakan `update-phone-number-settings`.

### AWS CLI

Untuk memperbarui nama panggilan keluar

`update-phone-number-settings` Contoh berikut memperbarui nama panggilan keluar default untuk AWS akun administrator.

```
aws chime update-phone-number-settings \  
  --calling-name "myName"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [UpdatePhoneNumberSettings](#) di Referensi AWS CLI Perintah.

## **update-phone-number**

Contoh kode berikut menunjukkan cara menggunakan `update-phone-number`.

### AWS CLI

Contoh 1: Untuk memperbarui jenis produk untuk nomor telepon

`update-phone-number` Contoh berikut memperbarui jenis produk nomor telepon yang ditentukan.

```
aws chime update-phone-number \  
  --product-type "myProductType"
```



```
--phone-number-id "+12065550100" \  
--product-type "BusinessCalling"
```

**Output:**

```
{  
  "PhoneNumber": {  
    "PhoneNumberId": "%2B12065550100",  
    "E164PhoneNumber": "+12065550100",  
    "Type": "Local",  
    "ProductType": "BusinessCalling",  
    "Status": "Unassigned",  
    "Capabilities": {  
      "InboundCall": true,  
      "OutboundCall": true,  
      "InboundSMS": true,  
      "OutboundSMS": true,  
      "InboundMMS": true,  
      "OutboundMMS": true  
    },  
    "Associations": [],  
    "CallingName": "phonenumber1",  
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",  
    "UpdatedTimestamp": "2019-08-12T21:44:07.591Z"  
  }  
}
```

Contoh 2: Untuk memperbarui nama panggilan keluar untuk nomor telepon

update-phone-number Contoh berikut memperbarui nama panggilan keluar untuk nomor telepon yang ditentukan.

```
aws chime update-phone-number -- phone-number-id "+12065550100" --calling-name  
"phonenumber2"
```

**Output:**

```
{  
  "PhoneNumber": {  
    "PhoneNumberId": "%2B12065550100",  
    "E164PhoneNumber": "+12065550100",  
    "Type": "Local",
```

```

    "ProductType": "BusinessCalling",
    "Status": "Unassigned",
    "Capabilities": {
      "InboundCall": true,
      "OutboundCall": true,
      "InboundSMS": true,
      "OutboundSMS": true,
      "InboundMMS": true,
      "OutboundMMS": true
    },
    "Associations": [],
    "CallingName": "phonenumber2",
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
    "UpdatedTimestamp": "2019-08-12T21:44:07.591Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Nomor Telepon](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [UpdatePhoneNumber](#) di Referensi AWS CLI Perintah.

## update-proxy-session

Contoh kode berikut menunjukkan cara menggunakan `update-proxy-session`.

### AWS CLI

Untuk memperbarui sesi proxy

`update-proxy-session` Contoh berikut memperbarui kemampuan sesi proxy.

```

aws chime update-proxy-session \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --proxy-session-id 123a4bc5-67d8-901e-2f3g-h4ghjk567891 \
  --capabilities "Voice"

```

Output:

```

{
  "ProxySession": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",

```

```

    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",
    "Status": "Open",
    "ExpiryMinutes": 60,
    "Capabilities": [
      "Voice"
    ],
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",
    "Participants": [
      {
        "PhoneNumber": "+12065550100",
        "ProxyPhoneNumber": "+19135550199"
      },
      {
        "PhoneNumber": "+14015550101",
        "ProxyPhoneNumber": "+19135550199"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Sesi Telepon Proxy](#) di Panduan Pengembang Amazon Chime.

- Untuk detail API, lihat [UpdateProxySession](#) di Referensi AWS CLI Perintah.

## update-room-membership

Contoh kode berikut menunjukkan cara menggunakan `update-room-membership`.

### AWS CLI

Untuk memperbarui keanggotaan kamar

`update-room-membership` Contoh berikut memodifikasi peran anggota ruang obrolan yang ditentukan untuk Administrator.

```

aws chime update-room-membership \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
  --member-id 1ab2345c-67de-8901-f23g-45h678901j2k \
  --role Administrator

```

**Output:**

```
{
  "RoomMembership": {
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
    "Member": {
      "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",
      "MemberType": "User",
      "Email": "sofiamartinez@example.com",
      "FullName": "Sofia Martinez",
      "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
    },
    "Role": "Administrator",
    "InvitedBy": "arn:aws:iam::111122223333:user/admin",
    "UpdatedTimestamp": "2019-12-02T22:40:22.931Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [UpdateRoomMembership](#) di Referensi AWS CLI Perintah.

**update-room**

Contoh kode berikut menunjukkan cara menggunakan `update-room`.

**AWS CLI**

Untuk memperbarui ruang obrolan

`update-room` Contoh berikut memodifikasi nama ruang obrolan yang ditentukan.

```
aws chime update-room \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
  --name teamRoom
```

**Output:**

```
{
```

```

    "Room": {
      "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
      "Name": "teamRoom",
      "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
      "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
      "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
      "UpdatedTimestamp": "2019-12-02T22:33:19.310Z"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat Ruang Obrolan](#) di Panduan Pengguna Amazon Chime.

- Untuk detail API, lihat [UpdateRoom](#) di Referensi AWS CLI Perintah.

## update-user-settings

Contoh kode berikut menunjukkan cara menggunakan `update-user-settings`.

### AWS CLI

Untuk memperbarui pengaturan pengguna

`update-user-settings` Contoh berikut memungkinkan pengguna yang ditentukan untuk melakukan panggilan masuk dan keluar dan mengirim dan menerima pesan SMS.

```

aws chime update-user-settings \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k \
  --user-settings "Telephony={InboundCalling=true,OutboundCalling=true,SMS=true}"

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Nomor Telepon Pengguna](#) di Panduan Administrasi Amazon Chime.

- Untuk detail API, lihat [UpdateUserSettings](#) di Referensi AWS CLI Perintah.

## update-user

Contoh kode berikut menunjukkan cara menggunakan `update-user`.

## AWS CLI

Untuk memperbarui detail pengguna

Contoh ini memperbarui detail yang ditentukan untuk pengguna yang ditentukan.

Perintah:

```
aws chime update-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE \  
  --license-type "Basic"
```

Output:

```
{  
  "User": {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"  
  }  
}
```

- Untuk detail API, lihat [UpdateUser](#) di Referensi AWS CLI Perintah.

## update-voice-connector-group

Contoh kode berikut menunjukkan cara menggunakan `update-voice-connector-group`.

### AWS CLI

Untuk memperbarui detail untuk grup Amazon Chime Voice Connector

Contoh berikut memperbarui detail grup Amazon Chime Voice Connector yang ditentukan.

```
aws chime update-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jk18901 \  
  --name "newGroupName" \  
  --voice-connector-items VoiceConnectorId=abcdef1ghij2klmno3pqr4,Priority=1
```

Output:

```
{
  "VoiceConnectorGroup": {
    "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jkl8901",
    "Name": "newGroupName",
    "VoiceConnectorItems": [
      {
        "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
        "Priority": 1
      }
    ],
    "CreatedTimestamp": "2019-09-18T16:38:34.734Z",
    "UpdatedTimestamp": "2019-10-28T19:00:57.081Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [UpdateVoiceConnectorGroup](#) di Referensi AWS CLI Perintah.

## update-voice-connector

Contoh kode berikut menunjukkan cara menggunakan `update-voice-connector`.

### AWS CLI

Untuk memperbarui detail untuk Konektor Suara Amazon Chime

`update-voice-connector` Contoh berikut memperbarui nama Konektor Suara Amazon Chime yang ditentukan.

```
aws chime update-voice-connector \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --name newName \
  --require-encryption
```

Output:

```
{
  "VoiceConnector": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
```

```
    "AwsRegion": "us-west-2",
    "Name": "newName",
    "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",
    "RequireEncryption": true,
    "CreatedTimestamp": "2019-09-18T20:34:01.352Z",
    "UpdatedTimestamp": "2019-09-18T20:40:52.895Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Konektor Suara Amazon Chime di Panduan Administrasi Amazon Chime](#).

- Untuk detail API, lihat [UpdateVoiceConnector](#) di Referensi AWS CLI Perintah.

## Contoh Cloud Control API menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum menggunakan AWS Command Line Interface with Cloud Control API.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-resource**

Contoh kode berikut menunjukkan cara menggunakan `create-resource`.

AWS CLI

Untuk membuat sumber daya



`create-resource` Contoh berikut membuat sumber daya AWS: `:Kinesis: :Stream`, bernama `ResourceExample`, dengan periode retensi 168 jam dan hitungan pecahan tiga.

```
aws cloudcontrol create-resource \  
  --type-name AWS::Kinesis::Stream \  
  --desired-state "{\"Name\": \"ResourceExample\", \"RetentionPeriodHours\":168, \  
  \"ShardCount\":3}"
```

Output:

```
{  
  "ProgressEvent": {  
    "EventTime": 1632506656.706,  
    "TypeName": "AWS::Kinesis::Stream",  
    "OperationStatus": "IN_PROGRESS",  
    "Operation": "CREATE",  
    "Identifier": "ResourceExample",  
    "RequestToken": "20999d87-e304-4725-ad84-832dcbfd7fc5"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat sumber daya](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [CreateResource](#) di Referensi AWS CLI Perintah.

## **delete-resource**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource`.

### AWS CLI

Untuk menghapus sumber daya

`delete-resource` Contoh berikut menghapus sumber daya AWS: `:Kinesis: :Stream` dengan pengenal `ResourceExample` dari akun Anda. AWS

```
aws cloudcontrol delete-resource \  
  --type-name AWS::Kinesis::Stream \  
  --identifier ResourceExample
```

**Output:**

```
{
  "ProgressEvent": {
    "TypeName": "AWS::Kinesis::Stream",
    "Identifier": "ResourceExample",
    "RequestToken": "e48f26ff-d0f9-4ab8-a878-120db1edf111",
    "Operation": "DELETE",
    "OperationStatus": "IN_PROGRESS",
    "EventTime": 1632950300.14
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus sumber daya](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [DeleteResource](#) di Referensi AWS CLI Perintah.

**get-resource-request-status**

Contoh kode berikut menunjukkan cara menggunakan `get-resource-request-status`.

**AWS CLI**

Untuk mendapatkan informasi status permintaan sumber daya

`get-resource-request-status` Contoh berikut mengembalikan informasi status tentang permintaan sumber daya tertentu.

```
aws cloudcontrol get-resource-request-status \
  --request-token "e1a6b86e-46bd-41ac-bfba-001234567890"
```

**Output:**

```
{
  "ProgressEvent": {
    "TypeName": "AWS::Kinesis::Stream",
    "Identifier": "Demo",
    "RequestToken": "e1a6b86e-46bd-41ac-bfba-001234567890",
    "Operation": "CREATE",
    "OperationStatus": "FAILED",
  }
}
```

```

    "EventTime": 1632950268.481,
    "StatusMessage": "Resource of type 'AWS::Kinesis::Stream' with identifier
'Demo' already exists.",
    "ErrorCode": "AlreadyExists"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola permintaan pengoperasian sumber daya](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [GetResourceRequestStatus](#) di Referensi AWS CLI Perintah.

## get-resource

Contoh kode berikut menunjukkan cara menggunakan `get-resource`.

### AWS CLI

Untuk mendapatkan status sumber daya saat ini

`get-resource` Contoh berikut mengembalikan keadaan saat ini dari `AWS::Kinesis::Stream` sumber daya bernama `ResourceExample`

```

aws cloudcontrol get-resource \
  --type-name AWS::Kinesis::Stream \
  --identifier ResourceExample

```

Output:

```

{
  "TypeName": "AWS::Kinesis::Stream",
  "ResourceDescription": {
    "Identifier": "ResourceExample",
    "Properties": "{\"Arn\":\"arn:aws:kinesis:us-west-2:099908667365:stream/ResourceExample\", \"RetentionPeriodHours\":168, \"Name\":\"ResourceExample\", \"ShardCount\":3}"
  }
}

```

Untuk informasi selengkapnya, lihat [Membaca status sumber daya saat ini](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [GetResource](#) di Referensi AWS CLI Perintah.

## list-resource-requests

Contoh kode berikut menunjukkan cara menggunakan `list-resource-requests`.

### AWS CLI

Untuk membuat daftar permintaan operasi sumber daya aktif

`list-resource-requests` Contoh berikut mencantumkan permintaan sumber daya untuk operasi CREATE dan UPDATE yang gagal di AWS akun Anda.

```
aws cloudcontrol list-resource-requests \
  --resource-request-status-filter Operations=CREATE,OperationStatuses=FAILED
```

### Output:

```
{
  "ResourceRequestStatusSummaries": [
    {
      "TypeName": "AWS::Kinesis::Stream",
      "Identifier": "Demo",
      "RequestToken": "e1a6b86e-46bd-41ac-bfba-633abcdfdbd7",
      "Operation": "CREATE",
      "OperationStatus": "FAILED",
      "EventTime": 1632950268.481,
      "StatusMessage": "Resource of type 'AWS::Kinesis::Stream' with
identifier 'Demo' already exists.",
      "ErrorCode": "AlreadyExists"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola permintaan pengoperasian sumber daya](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [ListResourceRequests](#) di Referensi AWS CLI Perintah.

## list-resources

Contoh kode berikut menunjukkan cara menggunakan `list-resources`.

## AWS CLI

Untuk membuat daftar sumber daya dari jenis tertentu

`list-resources` Contoh berikut mencantumkan sumber daya AWS: `:Kinesis: :Stream` yang disediakan di akun Anda. AWS

```
aws cloudcontrol list-resources \
  --type-name AWS::Kinesis::Stream
```

Output:

```
{
  "TypeName": "AWS::Kinesis::Stream",
  "ResourceDescriptions": [
    {
      "Identifier": "MyKinesisStream",
      "Properties": "{\"Name\":\"MyKinesisStream\"}"
    },
    {
      "Identifier": "AnotherStream",
      "Properties": "{\"Name\":\"AnotherStream\"}"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menemukan sumber daya](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [ListResources](#) di Referensi AWS CLI Perintah.

## update-resource

Contoh kode berikut menunjukkan cara menggunakan `update-resource`.

## AWS CLI

Untuk memperbarui properti sumber daya yang ada

`update-resource` Contoh berikut memperbarui kebijakan retensi LogGroup sumber daya AWS: `:Logs: :` yang diberi nama `ExampleLogGroup` menjadi 90 hari.

```
aws cloudcontrol update-resource \  
  --type-name AWS::Logs::LogGroup \  
  --identifier ExampleLogGroup \  
  --patch-document "[{"op":"replace","path":"/RetentionInDays","value":90}]"
```

Output:

```
{  
  "ProgressEvent": {  
    "EventTime": "2021-08-09T18:17:15.219Z",  
    "TypeName": "AWS::Logs::LogGroup",  
    "OperationStatus": "IN_PROGRESS",  
    "Operation": "UPDATE",  
    "Identifier": "ExampleLogGroup",  
    "RequestToken": "5f40c577-3534-4b20-9599-0b0123456789"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui sumber daya](#) di Panduan Pengguna Cloud Control API.

- Untuk detail API, lihat [UpdateResource](#) di Referensi AWS CLI Perintah.

## AWS Cloud Map contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS Cloud Map.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

## Tindakan

### **create-private-dns-namespace**

Contoh kode berikut menunjukkan cara menggunakan `create-private-dns-namespace`.

#### AWS CLI

Untuk membuat namespace DNS pribadi

`create-private-dns-namespace` Contoh berikut membuat namespace DNS pribadi.

```
aws servicediscovery create-private-dns-namespace \  
  --name example.com \  
  --vpc vpc-1c56417b
```

Output:

```
{  
  "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd"  
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya dengan `get-operation`. Untuk informasi selengkapnya, lihat [get-operation](#).

Untuk informasi selengkapnya, lihat [Membuat ruang nama](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [CreatePrivateDnsNamespace](#) di Referensi AWS CLI Perintah.

### **create-service**

Contoh kode berikut menunjukkan cara menggunakan `create-service`.

#### AWS CLI

Untuk membuat layanan

`create-service` Contoh berikut menciptakan layanan.

```
aws servicediscovery create-service \  
  --name example.com \  
  --vpc vpc-1c56417b
```

```
--name myservice \
--namespace-id ns-ylexjili4cdxy3xm \
--dns-config "NamespaceId=ns-
ylexjili4cdxy3xm,RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=60}]"
```

Output:

```
{
  "Service": {
    "Id": "srv-p5zdwlg5uvvzjita",
    "Arn": "arn:aws:servicediscovery:us-west-2:803642222207:service/srv-
p5zdwlg5uvvzjita",
    "Name": "myservice",
    "NamespaceId": "ns-ylexjili4cdxy3xm",
    "DnsConfig": {
      "NamespaceId": "ns-ylexjili4cdxy3xm",
      "RoutingPolicy": "MULTIVALUE",
      "DnsRecords": [
        {
          "Type": "A",
          "TTL": 60
        }
      ]
    },
    "CreateDate": 1587081768.334,
    "CreatorRequestId": "567c1193-6b00-4308-bd57-ad38a8822d25"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat layanan](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [CreateService](#) di Referensi AWS CLI Perintah.

## delete-namespace

Contoh kode berikut menunjukkan cara menggunakan `delete-namespace`.

### AWS CLI

Untuk menghapus namespace

`delete-namespace` Contoh berikut menghapus namespace.



```
aws servicediscovery delete-namespace \  
  --id ns-ylexjili4cdxy3xm
```

Output:

```
{  
  "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k98y6drk"  
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation`. Untuk informasi selengkapnya, lihat [get-operation](#).

Untuk informasi selengkapnya, lihat [Menghapus ruang nama di Panduan](#) Pengembang AWS Cloud Map.

- Untuk detail API, lihat [DeleteNamespaced](#) di Referensi AWS CLI Perintah.

## **delete-service**

Contoh kode berikut menunjukkan cara menggunakan `delete-service`.

AWS CLI

Untuk menghapus layanan

`delete-service` Contoh berikut menghapus layanan.

```
aws servicediscovery delete-service \  
  --id srv-p5zdwlg5uvvzjita
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus layanan](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [DeleteService](#) di Referensi AWS CLI Perintah.

## **deregister-instance**

Contoh kode berikut menunjukkan cara menggunakan `deregister-instance`.

## AWS CLI

Untuk membatalkan pendaftaran instance layanan

`deregister-instance` Contoh berikut membatalkan pendaftaran instance layanan.

```
aws servicediscovery deregister-instance \  
  --service-id srv-p5zdwlg5uvvzjita \  
  --instance-id myservice-53
```

Output:

```
{  
  "OperationId": "4yejorelbukcjpnr6t1mrghsjwpngf4-k98rnaiq"  
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation`. Untuk informasi selengkapnya, lihat [get-operation](#).

Untuk informasi selengkapnya, lihat [membatalkan pendaftaran instance layanan](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [DeregisterInstance](#) di Referensi AWS CLI Perintah.

## `discover-instances`

Contoh kode berikut menunjukkan cara menggunakan `discover-instances`.

## AWS CLI

Untuk menemukan instans terdaftar

`discover-instances` Contoh berikut menemukan contoh terdaftar.

```
aws servicediscovery discover-instances \  
  --namespace-name example.com \  
  --service-name myservice \  
  --max-results 10 \  
  --health-status ALL
```

Output:

```
{
  "Instances": [
    {
      "InstanceId": "myservice-53",
      "NamespaceName": "example.com",
      "ServiceName": "myservice",
      "HealthStatus": "UNKNOWN",
      "Attributes": {
        "AWS_INSTANCE_IPV4": "172.2.1.3",
        "AWS_INSTANCE_PORT": "808"
      }
    }
  ]
}
```

- Untuk detail API, lihat [DiscoverInstances](#) di Referensi AWS CLI Perintah.

## get-operation

Contoh kode berikut menunjukkan cara menggunakan `get-operation`.

### AWS CLI

Untuk mendapatkan hasil operasi

`get-operation` Contoh berikut mendapatkan hasil operasi.

```
aws servicediscovery get-operation \
  --operation-id gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd
```

Output:

```
{
  "Operation": {
    "Id": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd",
    "Type": "CREATE_NAMESPACE",
    "Status": "SUCCESS",
    "CreateDate": 1587055860.121,
    "UpdateDate": 1587055900.469,
    "Targets": {
      "NAMESPACE": "ns-ylexjili4cdxy3xm"
    }
  }
}
```

```

    }
  }
}

```

- Untuk detail API, lihat [GetOperation](#) di Referensi AWS CLI Perintah.

## list-instances

Contoh kode berikut menunjukkan cara menggunakan `list-instances`.

### AWS CLI

Untuk membuat daftar instance layanan

`list-instances` Contoh berikut mencantumkan instance layanan.

```

aws servicediscovery list-instances \
  --service-id srv-qzpwvt2tfqcegapy

```

Output:

```

{
  "Instances": [
    {
      "Id": "i-06bdabbae60f65a4e",
      "Attributes": {
        "AWS_INSTANCE_IPV4": "172.2.1.3",
        "AWS_INSTANCE_PORT": "808"
      }
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat daftar instance layanan](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [ListInstances](#) di Referensi AWS CLI Perintah.

## list-namespaces

Contoh kode berikut menunjukkan cara menggunakan `list-namespaces`.

## AWS CLI

Untuk daftar ruang nama

`list-namespaces` Contoh berikut mencantumkan ruang nama.

```
aws servicediscovery list-namespaces
```

Output:

```
{
  "Namespaces": [
    {
      "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-a3ccy2e7e3a7rile",
      "CreateDate": 1585354387.357,
      "Id": "ns-a3ccy2e7e3a7rile",
      "Name": "local",
      "Properties": {
        "DnsProperties": {
          "HostedZoneId": "Z06752353VBUDTC32S84S"
        },
        "HttpProperties": {
          "HttpName": "local"
        }
      },
      "Type": "DNS_PRIVATE"
    },
    {
      "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-pocfyjtrsmwtvcxx",
      "CreateDate": 1586468974.698,
      "Description": "My second namespace",
      "Id": "ns-pocfyjtrsmwtvcxx",
      "Name": "My-second-namespace",
      "Properties": {
        "DnsProperties": {},
        "HttpProperties": {
          "HttpName": "My-second-namespace"
        }
      },
      "Type": "HTTP"
    }
  ],
}
```

```

    {
      "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
ylexjili4cdxy3xm",
      "CreateDate": 1587055896.798,
      "Id": "ns-ylexjili4cdxy3xm",
      "Name": "example.com",
      "Properties": {
        "DnsProperties": {
          "HostedZoneId": "Z09983722P0QME1B3KC8I"
        },
        "HttpProperties": {
          "HttpName": "example.com"
        }
      },
      "Type": "DNS_PRIVATE"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat daftar ruang nama di Panduan](#) Pengembang AWS Cloud Map.

- Untuk detail API, lihat [ListNamespaces](#) di Referensi AWS CLI Perintah.

## list-services

Contoh kode berikut menunjukkan cara menggunakan `list-services`.

### AWS CLI

Untuk daftar layanan

`list-services` Contoh berikut mencantumkan layanan.

```
aws servicediscovery list-services
```

Output:

```

{
  "Services": [
    {
      "Id": "srv-p5zdwlg5uvvzjita",

```

```

    "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
p5zdwlg5uvvzjita",
    "Name": "myservice",
    "DnsConfig": {
      "RoutingPolicy": "MULTIVALUE",
      "DnsRecords": [
        {
          "Type": "A",
          "TTL": 60
        }
      ]
    },
    "CreateDate": 1587081768.334
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat daftar layanan](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [ListServices](#) di Referensi AWS CLI Perintah.

## register-instance

Contoh kode berikut menunjukkan cara menggunakan `register-instance`.

### AWS CLI

Untuk mendaftarkan instance layanan

`register-instance` Contoh berikut mendaftarkan instance layanan.

```

aws servicediscovery register-instance \
  --service-id srv-p5zdwlg5uvvzjita \
  --instance-id myservice-53 \
  --attributes=AWS_INSTANCE_IPV4=172.2.1.3,AWS_INSTANCE_PORT=808

```

Output:

```

{
  "OperationId": "4yejorelbukcjpnr6t1mrghsjwpngf4-k95yg2u7"
}

```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation`. Untuk informasi selengkapnya, lihat [get-operation](#).

Untuk informasi selengkapnya, lihat [Mendaftarkan instans](#) di Panduan Pengembang AWS Cloud Map.

- Untuk detail API, lihat [RegisterInstance](#) di Referensi AWS CLI Perintah.

## AWS Cloud9 contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Cloud9.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-environment-ec2**

Contoh kode berikut menunjukkan cara menggunakan `create-environment-ec2`.

### AWS CLI

Untuk membuat lingkungan pengembangan AWS Cloud9 EC2

`create-environment-ec2` Contoh berikut ini membuat lingkungan pengembangan AWS Cloud9 dengan pengaturan yang ditentukan, meluncurkan instance Amazon Elastic Compute Cloud (Amazon EC2), dan kemudian menghubungkan dari instance ke lingkungan.

```
aws cloud9 create-environment-ec2 \
```



```
--name my-demo-env \  
--description "My demonstration development environment." \  
--instance-type t2.micro --image-id amazonlinux-2023-x86_64 \  
--subnet-id subnet-1fab8aEX \  
--automatic-stop-time-minutes 60 \  
--owner-arn arn:aws:iam::123456789012:user/MyDemoUser
```

Output:

```
{  
  "environmentId": "8a34f51ce1e04a08882f1e811bd706EX"  
}
```

Untuk informasi selengkapnya, lihat [Membuat Lingkungan EC2 di Panduan Pengguna AWS Cloud9](#).

- Untuk detail API, lihat [CreateEnvironmentEc2](#) di Referensi AWS CLI Perintah.

## **create-environment-membership**

Contoh kode berikut menunjukkan cara menggunakan `create-environment-membership`.

### AWS CLI

Untuk menambahkan anggota lingkungan ke lingkungan pengembangan AWS Cloud9

Contoh ini menambahkan anggota lingkungan yang ditentukan ke lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 create-environment-membership --environment-id  
8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/  
AnotherDemoUser --permissions read-write
```

Output:

```
{  
  "membership": {  
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",  
    "userId": "AIDAJ3LOROMOUXTBSUGEX",  
    "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
```

```
    "permissions": "read-write"  
  }  
}
```

- Untuk detail API, lihat [CreateEnvironmentMembership](#) di Referensi AWS CLI Perintah.

## **delete-environment-membership**

Contoh kode berikut menunjukkan cara menggunakan `delete-environment-membership`.

### AWS CLI

Untuk menghapus anggota lingkungan dari lingkungan pengembangan AWS Cloud9

Contoh ini menghapus anggota lingkungan yang ditentukan dari lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 delete-environment-membership --environment-id  
8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/  
AnotherDemoUser
```

Output:

```
None.
```

- Untuk detail API, lihat [DeleteEnvironmentMembership](#) di Referensi AWS CLI Perintah.

## **delete-environment**

Contoh kode berikut menunjukkan cara menggunakan `delete-environment`.

### AWS CLI

Untuk menghapus lingkungan pengembangan AWS Cloud9

Contoh ini menghapus lingkungan pengembangan AWS Cloud9 yang ditentukan. Jika instans Amazon EC2 terhubung ke lingkungan, juga menghentikan instans.

Perintah:

```
aws cloud9 delete-environment --environment-id 8a34f51ce1e04a08882f1e811bd706EX
```

Output:

```
None.
```

- Untuk detail API, lihat [DeleteEnvironment](#) di Referensi AWS CLI Perintah.

## describe-environment-memberships

Contoh kode berikut menunjukkan cara menggunakan `describe-environment-memberships`.

### AWS CLI

Untuk mendapatkan informasi tentang anggota lingkungan untuk lingkungan pengembangan AWS Cloud9

Contoh ini mendapatkan informasi tentang anggota lingkungan untuk lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 describe-environment-memberships --environment-id
8a34f51ce1e04a08882f1e811bd706EX
```

Output:

```
{
  "memberships": [
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJ3LOROMOUCTBSU6EX",
      "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
      "permissions": "read-write"
    },
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

```
]
}
```

Untuk mendapatkan informasi tentang pemilik lingkungan pengembangan AWS Cloud9

Contoh ini mendapatkan informasi tentang pemilik lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 describe-environment-memberships --environment-id
8a34f51ce1e04a08882f1e811bd706EX --permissions owner
```

Output:

```
{
  "memberships": [
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

Untuk mendapatkan informasi tentang anggota lingkungan untuk beberapa lingkungan pengembangan AWS Cloud9

Contoh ini mendapatkan informasi tentang anggota lingkungan yang ditentukan untuk beberapa lingkungan pengembangan AWS Cloud9.

Perintah:

```
aws cloud9 describe-environment-memberships --user-arn
arn:aws:iam::123456789012:user/MyDemoUser
```

Output:

```
{
  "memberships": [
    {
```

```
"environmentId": "10a75714bd494714929e7f5ec4125aEX",
"lastAccess": 1516213427.0,
"userId": "AIDAJNUEDQAQWFELJDLEX",
"userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
"permissions": "owner"
},
{
"environmentId": "1980b80e5f584920801c09086667f0EX",
"lastAccess": 1516144884.0,
"userId": "AIDAJNUEDQAQWFELJDLEX",
"userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
"permissions": "owner"
}
]
}
```

- Untuk detail API, lihat [DescribeEnvironmentMemberships](#) di Referensi AWS CLI Perintah.

## describe-environment-status

Contoh kode berikut menunjukkan cara menggunakan `describe-environment-status`.

### AWS CLI

Untuk mendapatkan informasi status untuk lingkungan pengembangan AWS Cloud9

Contoh ini mendapatkan informasi status untuk lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 describe-environment-status --environment-id
685f892f431b45c2b28cb69eadcdb0EX
```

Output:

```
{
"status": "ready",
"message": "Environment is ready to use"
}
```

- Untuk detail API, lihat [DescribeEnvironmentStatus](#) di Referensi AWS CLI Perintah.

## describe-environments

Contoh kode berikut menunjukkan cara menggunakan `describe-environments`.

### AWS CLI

Untuk mendapatkan informasi tentang lingkungan pengembangan AWS Cloud9

Contoh ini mendapatkan informasi tentang lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 describe-environments --environment-ids 685f892f431b45c2b28cb69eadcdb0EX
349c86d4579e4e7298d500ff57a6b2EX
```

Output:

```
{
  "environments": [
    {
      "id": "685f892f431b45c2b28cb69eadcdb0EX",
      "name": "my-demo-ec2-env",
      "description": "Created from CodeStar.",
      "type": "ec2",
      "arn": "arn:aws:cloud9:us-
east-1:123456789012:environment:685f892f431b45c2b28cb69eadcdb0EX",
      "ownerArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "lifecycle": {
        "status": "CREATED"
      }
    },
    {
      "id": "349c86d4579e4e7298d500ff57a6b2EX",
      "name": "my-demo-ssh-env",
      "description": "",
      "type": "ssh",
      "arn": "arn:aws:cloud9:us-
east-1:123456789012:environment:349c86d4579e4e7298d500ff57a6b2EX",
      "ownerArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "lifecycle": {
        "status": "CREATED"
      }
    }
  ]
}
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeEnvironments](#) di Referensi AWS CLI Perintah.

## list-environments

Contoh kode berikut menunjukkan cara menggunakan `list-environments`.

### AWS CLI

Untuk mendapatkan daftar pengidentifikasi lingkungan pengembangan AWS Cloud9 yang tersedia

Contoh ini mendapatkan daftar pengidentifikasi lingkungan pengembangan AWS Cloud9 yang tersedia.

Perintah:

```
aws cloud9 list-environments
```

Output:

```
{  
  "environmentIds": [  
    "685f892f431b45c2b28cb69eadcdb0EX",  
    "1980b80e5f584920801c09086667f0EX"  
  ]  
}
```

- Untuk detail API, lihat [ListEnvironments](#) di Referensi AWS CLI Perintah.

## update-environment-membership

Contoh kode berikut menunjukkan cara menggunakan `update-environment-membership`.

### AWS CLI

Untuk mengubah pengaturan anggota lingkungan yang ada untuk lingkungan pengembangan AWS Cloud9

Contoh ini mengubah pengaturan anggota lingkungan yang ada yang ditentukan untuk lingkungan pengembangan AWS Cloud9 yang ditentukan.

Perintah:

```
aws cloud9 update-environment-membership --environment-id
8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/
AnotherDemoUser --permissions read-only
```

Output:

```
{
  "membership": {
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
    "userId": "AIDAJ3LOROMOUCTBSU6EX",
    "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
    "permissions": "read-only"
  }
}
```

- Untuk detail API, lihat [UpdateEnvironmentMembership](#) di Referensi AWS CLI Perintah.

## update-environment

Contoh kode berikut menunjukkan cara menggunakan `update-environment`.

### AWS CLI

Untuk mengubah pengaturan lingkungan pengembangan AWS Cloud9 yang ada

Contoh ini mengubah setelan yang ditentukan dari lingkungan pengembangan AWS Cloud9 yang sudah ditentukan.

Perintah:

```
aws cloud9 update-environment --environment-id 8a34f51ce1e04a08882f1e811bd706EX
--name my-changed-demo-env --description "My changed demonstration development
environment."
```

Output:



None .

- Untuk detail API, lihat [UpdateEnvironment](#) di Referensi AWS CLI Perintah.

## AWS CloudFormation contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS CloudFormation.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **activate-type**

Contoh kode berikut menunjukkan cara menggunakan `activate-type`.

AWS CLI

Untuk mengaktifkan tipe

`activate-type` Contoh berikut mengaktifkan ekstensi pihak ketiga publik, membuatnya tersedia untuk digunakan dalam template tumpukan.

```
aws cloudformation activate-type \  
  --region us-west-2 \  
  --type RESOURCE \  
  --type-name Example::Test::1234567890abcdef0 \  
  --type-name-alias Example::Test::Alias
```

**Output:**

```
{
  "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/Example-Test-Alias"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [ActivateType](#) di Referensi AWS CLI Perintah.

**batch-describe-type-configurations**

Contoh kode berikut menunjukkan cara menggunakan `batch-describe-type-configurations`.

**AWS CLI**

Untuk menggambarkan konfigurasi tipe secara batch

`batch-describe-type-configurations` Contoh berikut mengkonfigurasi data untuk tipe.

```
aws cloudformation batch-describe-type-configurations \
  --region us-west-2 \
  --type-configuration-identifiers TypeArn="arn:aws:cloudformation:us-
west-2:123456789012:type/resource/Example-Test-
Type,TypeConfigurationAlias=MyConfiguration"
```

**Output:**

```
{
  "Errors": [],
  "UnprocessedTypeConfigurations": [],
  "TypeConfigurations": [
    {
      "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/Example-Test-Type",
      "Alias": "MyConfiguration",
      "Configuration": "{\n  \"Example\": {\n    \"ApiKey\": \"examplekey\", \n    \"ApplicationKey\": \"examplekey1\", \n    \"ApiURL\": \"exampleurl\" \n  }\n}"
```

```
        "LastUpdated": "2021-10-01T15:25:46.210000+00:00",
        "TypeArn": "arn:aws:cloudformation:us-east-1:123456789012:type/resource/
Example-Test-Type"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [BatchDescribeTypeConfigurations](#) di Referensi AWS CLI Perintah.

## cancel-update-stack

Contoh kode berikut menunjukkan cara menggunakan `cancel-update-stack`.

### AWS CLI

Untuk membatalkan pembaruan tumpukan yang sedang berlangsung

`cancel-update-stack` Perintah berikut membatalkan pembaruan tumpukan pada `myteststack` tumpukan:

```
aws cloudformation cancel-update-stack --stack-name myteststack
```

- Untuk detail API, lihat [CancelUpdateStack](#) di Referensi AWS CLI Perintah.

## continue-update-rollback

Contoh kode berikut menunjukkan cara menggunakan `continue-update-rollback`.

### AWS CLI

Untuk mencoba kembali rollback pembaruan

`continue-update-rollback` Contoh berikut melanjutkan operasi rollback dari pembaruan tumpukan yang sebelumnya gagal.

```
aws cloudformation continue-update-rollback \  
  --stack-name my-stack
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [ContinueUpdateRollback](#) di Referensi AWS CLI Perintah.

## create-change-set

Contoh kode berikut menunjukkan cara menggunakan `create-change-set`.

### AWS CLI

Untuk membuat set perubahan

`create-change-set` Contoh berikut membuat set perubahan dengan `CAPABILITY_IAM` kemampuan. File `template.yaml` adalah AWS CloudFormation template di folder saat ini yang mendefinisikan tumpukan yang menyertakan sumber daya IAM.

```
aws cloudformation create-change-set \  
  --stack-name my-application \  
  --change-set-name my-change-set \  
  --template-body file://template.yaml \  
  --capabilities CAPABILITY_IAM
```

Output:

```
{  
  "Id": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-change-set/  
bc9555ba-a949-xmpl-bfb8-f41d04ec5784",  
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-application/  
d0a825a0-e4cd-xmpl-b9fb-061c69e99204"  
}
```

- Untuk detail API, lihat [CreateChangeSet](#) di Referensi AWS CLI Perintah.

## create-stack-instances

Contoh kode berikut menunjukkan cara menggunakan `create-stack-instances`.

### AWS CLI

Untuk membuat instance tumpukan

`create-stack-instances` Contoh berikut membuat instance dari tumpukan yang disetel dalam dua akun dan di empat wilayah. Pengaturan toleransi kesalahan memastikan bahwa pembaruan dicoba di semua akun dan wilayah, bahkan jika beberapa tumpukan tidak dapat dibuat.

```
aws cloudformation create-stack-instances \  
  --stack-set-name my-stack-set \  
  --accounts 123456789012 223456789012 \  
  --regions us-east-1 us-east-2 us-west-1 us-west-2 \  
  --operation-preferences FailureToleranceCount=7
```

Output:

```
{  
  "OperationId": "d7995c31-83c2-xmpl-a3d4-e9ca2811563f"  
}
```

Untuk membuat set tumpukan, gunakan `create-stack-set` perintah.

- Untuk detail API, lihat [CreateStackInstances](#) di Referensi AWS CLI Perintah.

## **create-stack-set**

Contoh kode berikut menunjukkan cara menggunakan `create-stack-set`.

### AWS CLI

Untuk membuat set tumpukan

`create-stack-set` Contoh berikut membuat kumpulan tumpukan menggunakan template file YAMAL yang ditentukan. `template.yaml` adalah AWS CloudFormation template di folder saat ini yang mendefinisikan tumpukan.

```
aws cloudformation create-stack-set \  
  --stack-set-name my-stack-set \  
  --template-body file://template.yaml \  
  --description "SNS topic"
```

Output:

```
{
```

```
"StackSetId": "my-stack-set:8d0f160b-d157-xmpl-a8e6-c0ce8e5d8cc1"
}
```

Untuk menambahkan instance tumpukan ke set tumpukan, gunakan `create-stack-instances` perintah.

- Untuk detail API, lihat [CreateStackSet](#) di Referensi AWS CLI Perintah.

## **create-stack**

Contoh kode berikut menunjukkan cara menggunakan `create-stack`.

### AWS CLI

Untuk membuat AWS CloudFormation tumpukan

`create-stack` Perintah berikut membuat tumpukan dengan nama `myteststack` menggunakan `sampletemplate.json` template:

```
aws cloudformation create-stack --stack-name myteststack --template-body file://
sampletemplate.json --parameters ParameterKey=KeyPairName,ParameterValue=TestKey
ParameterKey=SubnetIDs,ParameterValue=SubnetID1\\,SubnetID2
```

Output:

```
{
  "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896"
}
```

Untuk informasi selengkapnya, lihat Tumpukan di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [CreateStack](#) di Referensi AWS CLI Perintah.

## **deactivate-type**

Contoh kode berikut menunjukkan cara menggunakan `deactivate-type`.

### AWS CLI

Untuk menonaktifkan tipe

`deactivate-type` Contoh berikut menonaktifkan ekstensi publik yang sebelumnya diaktifkan di akun dan Wilayah ini.

```
aws cloudformation deactivate-type \  
  --region us-west-2 \  
  --type MODULE \  
  --type-name Example::Test::Type::MODULE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DeactivateType](#) di Referensi AWS CLI Perintah.

## **delete-change-set**

Contoh kode berikut menunjukkan cara menggunakan `delete-change-set`.

### AWS CLI

Untuk menghapus set perubahan

`delete-change-set` Contoh berikut menghapus set perubahan dengan menentukan nama set perubahan dan nama tumpukan.

```
aws cloudformation delete-change-set \  
  --stack-name my-stack \  
  --change-set-name my-change-set
```

Perintah ini tidak menghasilkan output.

`delete-change-set` Contoh berikut menghapus set perubahan dengan menentukan ARN lengkap dari set perubahan.

```
aws cloudformation delete-change-set \  
  --change-set-name arn:aws:cloudformation:us-east-2:123456789012:changeSet/my-  
change-set/4eca1a01-e285-xmpl-8026-9a1967bfb4b0
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteChangeSet](#) di Referensi AWS CLI Perintah.

## delete-stack-instances

Contoh kode berikut menunjukkan cara menggunakan `delete-stack-instances`.

### AWS CLI

Untuk menghapus instance tumpukan

`delete-stack-instances` Contoh berikut menghapus instance tumpukan yang disetel dalam dua akun di dua wilayah dan mengakhiri tumpukan.

```
aws cloudformation delete-stack-instances \  
  --stack-set-name my-stack-set \  
  --accounts 123456789012 567890123456 \  
  --regions us-east-1 us-west-1 \  
  --no-retain-stacks
```

Output:

```
{  
  "OperationId": "ad49f10c-fd1d-413f-a20a-8de6e2fa8f27"  
}
```

Untuk menghapus kumpulan tumpukan kosong, gunakan `delete-stack-set` perintah.

- Untuk detail API, lihat [DeleteStackInstances](#) di Referensi AWS CLI Perintah.

## delete-stack-set

Contoh kode berikut menunjukkan cara menggunakan `delete-stack-set`.

### AWS CLI

Untuk menghapus set tumpukan

Perintah berikut menghapus set tumpukan kosong yang ditentukan. Set tumpukan harus kosong.

```
aws cloudformation delete-stack-set \  
  --stack-set-name my-stack-set
```

Perintah ini tidak menghasilkan output.



Untuk menghapus instance dari kumpulan tumpukan, gunakan `delete-stack-instances` perintah.

- Untuk detail API, lihat [DeleteStackSet](#) di Referensi AWS CLI Perintah.

## **delete-stack**

Contoh kode berikut menunjukkan cara menggunakan `delete-stack`.

### AWS CLI

Untuk menghapus tumpukan

`delete-stack` Contoh berikut menghapus tumpukan yang ditentukan.

```
aws cloudformation delete-stack \  
  --stack-name my-stack
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteStack](#) di Referensi AWS CLI Perintah.

## **deploy**

Contoh kode berikut menunjukkan cara menggunakan `deploy`.

### AWS CLI

Berikut perintah menyebarkan template bernama `template.json` ke tumpukan bernama `my-new-stack`:

```
aws cloudformation deploy --template-file /path_to_template/template.json --stack-name my-new-stack --parameter-overrides Key1=Value1 Key2=Value2 --tags Key1=Value1 Key2=Value2
```

- Untuk detail API, lihat [Menerapkan](#) di Referensi AWS CLI Perintah.

## **deregister-type**

Contoh kode berikut menunjukkan cara menggunakan `deregister-type`.

## AWS CLI

Untuk membatalkan pendaftaran versi tipe

`deregister-type` Contoh berikut menghapus versi tipe yang ditentukan dari penggunaan aktif dalam CloudFormation registri, sehingga tidak dapat lagi digunakan dalam CloudFormation operasi.

```
aws cloudformation deregister-type \  
  --type RESOURCE \  
  --type-name My::Logs::LogGroup \  
  --version-id 00000002
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan CloudFormation Registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DeregisterType](#) di Referensi AWS CLI Perintah.

## `describe-account-limits`

Contoh kode berikut menunjukkan cara menggunakan `describe-account-limits`.

## AWS CLI

Untuk mendapatkan informasi tentang batas akun Anda

Perintah berikut mengambil daftar batas regional untuk akun saat ini.

```
aws cloudformation describe-account-limits
```

Output:

```
{  
  "AccountLimits": [  
    {  
      "Name": "StackLimit",  
      "Value": 200  
    },  
    {
```

```

        "Name": "StackOutputsLimit",
        "Value": 60
    },
    {
        "Name": "ConcurrentResourcesLimit",
        "Value": 2500
    }
]
}

```

- Untuk detail API, lihat [DescribeAccountLimits](#) di Referensi AWS CLI Perintah.

## describe-change-set

Contoh kode berikut menunjukkan cara menggunakan `describe-change-set`.

### AWS CLI

Untuk mendapatkan informasi tentang set perubahan

`describe-change-set` Contoh berikut menampilkan rincian dari set perubahan yang ditentukan oleh nama set perubahan dan nama stack.

```

aws cloudformation describe-change-set \
  --change-set-name my-change-set \
  --stack-name my-stack

```

`describe-change-set` Contoh berikut menampilkan rincian set perubahan yang ditentukan oleh ARN lengkap dari set perubahan:

```

aws cloudformation describe-change-set \
  --change-set-name arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
change-set/bc9555ba-a949-xmpl-bfb8-f41d04ec5784

```

Output:

```

{
  "Changes": [
    {
      "Type": "Resource",

```

```

    "ResourceChange": {
      "Action": "Modify",
      "LogicalResourceId": "function",
      "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
      "ResourceType": "AWS::Lambda::Function",
      "Replacement": "False",
      "Scope": [
        "Properties"
      ],
      "Details": [
        {
          "Target": {
            "Attribute": "Properties",
            "Name": "Timeout",
            "RequiresRecreation": "Never"
          },
          "Evaluation": "Static",
          "ChangeSource": "DirectModification"
        }
      ]
    }
  ],
  "ChangeSetName": "my-change-set",
  "ChangeSetId": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
change-set/4eca1a01-e285-xmpl-8026-9a1967bfb4b0",
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
  "StackName": "my-stack",
  "Description": null,
  "Parameters": null,
  "CreationTime": "2019-10-02T05:20:56.651Z",
  "ExecutionStatus": "AVAILABLE",
  "Status": "CREATE_COMPLETE",
  "StatusReason": null,
  "NotificationARNs": [],
  "RollbackConfiguration": {},
  "Capabilities": [
    "CAPABILITY_IAM"
  ],
  "Tags": null
}

```

- Untuk detail API, lihat [DescribeChangeSet](#) di Referensi AWS CLI Perintah.

## describe-publisher

Contoh kode berikut menunjukkan cara menggunakan `describe-publisher`.

### AWS CLI

Untuk menggambarkan penerbit

`describe-publisher` Contoh berikut mengkonfigurasi informasi untuk penerbit.

```
aws cloudformation describe-publisher \  
  --region us-west-2 \  
  --publisher-id 000q6TfUovXsEMmgKowxDZLLwqr2QUsh
```

Output:

```
{  
  "PublisherId": "000q6TfUovXsEMmgKowxDZLLwqr2QUshd2e75c8c",  
  "PublisherStatus": "VERIFIED",  
  "IdentityProvider": "AWS_Marketplace",  
  "PublisherProfile": "https://aws.amazon.com/marketplace/seller-profile?  
id=2c5dc1f0-17cd-4259-8e46-822a83dtegd"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DescribePublisher](#) di Referensi AWS CLI Perintah.

## describe-stack-drift-detection-status

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-drift-detection-status`.

### AWS CLI

Untuk memeriksa status operasi deteksi drift

`describe-stack-drift-detection-status` Contoh berikut menampilkan status operasi deteksi drift. Dapatkan ID by menjalankan `detect-stack-drift` perintah.

```
aws cloudformation describe-stack-drift-detection-status \  
  --region us-west-2 \  
  --stack-name my-stack
```

```
--stack-drift-detection-id 1a229160-e4d9-xmpl-ab67-0a4f93df83d4
```

### Output:

```
{
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
  "StackDriftDetectionId": "1a229160-e4d9-xmpl-ab67-0a4f93df83d4",
  "StackDriftStatus": "DRIFTED",
  "DetectionStatus": "DETECTION_COMPLETE",
  "DriftedStackResourceCount": 1,
  "Timestamp": "2019-10-02T05:54:30.902Z"
}
```

- Untuk detail API, lihat [DescribeStackDriftDetectionStatus](#) di Referensi AWS CLI Perintah.

## describe-stack-events

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-events`.

### AWS CLI

Untuk menggambarkan peristiwa tumpukan

`describe-stack-events` Contoh berikut menampilkan 2 peristiwa terbaru untuk tumpukan tertentu.

```
aws cloudformation describe-stack-events \
  --stack-name my-stack \
  --max-items 2

{
  "StackEvents": [
    {
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "EventId": "4e1516d0-e4d6-xmpl-b94f-0a51958a168c",
      "StackName": "my-stack",
      "LogicalResourceId": "my-stack",
      "PhysicalResourceId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "ResourceType": "AWS::CloudFormation::Stack",

```

```

        "Timestamp": "2019-10-02T05:34:29.556Z",
        "ResourceStatus": "UPDATE_COMPLETE"
    },
    {
        "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
        "EventId": "4dd3c810-e4d6-xmpl-bade-0aaf8b31ab7a",
        "StackName": "my-stack",
        "LogicalResourceId": "my-stack",
        "PhysicalResourceId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
        "ResourceType": "AWS::CloudFormation::Stack",
        "Timestamp": "2019-10-02T05:34:29.127Z",
        "ResourceStatus": "UPDATE_COMPLETE_CLEANUP_IN_PROGRESS"
    }
],
"NextToken": "eyJ0ZXh0VG9XMPLiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

- Untuk detail API, lihat [DescribeStackEvents](#) di Referensi AWS CLI Perintah.

## describe-stack-instance

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-instance`.

### AWS CLI

Untuk menggambarkan instance tumpukan

Perintah berikut menjelaskan sebuah instance dari set stack yang ditentukan dalam akun dan Region yang ditentukan. Kumpulan tumpukan ada di wilayah dan akun saat ini, dan instance ada di `us-west-2` wilayah di akun `123456789012`.

```

aws cloudformation describe-stack-instance \
  --stack-set-name my-stack-set \
  --stack-instance-account 123456789012 \
  --stack-instance-region us-west-2

```

Output:

```
{
```

```

"StackInstance": {
  "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
  "Region": "us-west-2",
  "Account": "123456789012",
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/
StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532/4287f9a0-e615-
xmpl-894a-12b31d3117be",
  "ParameterOverrides": [],
  "Status": "OUTDATED",
  "StatusReason": "ResourceLogicalId:ConfigBucket,
ResourceType:AWS::S3::Bucket, ResourceStatusReason:You have attempted to create
more buckets than allowed (Service: Amazon S3; Status Code: 400; Error Code:
TooManyBuckets; Request ID: F7F21CXMPL580224; S3 Extended Request ID: egd/
Fdt89BXMPLyiqbMNLjVk55Yqqvi3NYW2nKLUVWhUGEHnFcmZdyj967lhriaG/dWMobS040o=)."
}
}

```

- Untuk detail API, lihat [DescribeStackInstances](#) di Referensi AWS CLI Perintah.

## describe-stack-resource-drifts

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-resource-drifts`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber daya yang melayang dari definisi tumpukan

Perintah berikut menampilkan informasi tentang sumber daya hanyut untuk tumpukan yang ditentukan. Untuk memulai deteksi drift, gunakan perintah `detect-stack-drift`:

```

aws cloudformation describe-stack-resource-drifts \
  --stack-name my-stack

```

Output menunjukkan fungsi AWS Lambda yang telah dimodifikasi: out-of-band

```

{
  "StackResourceDrifts": [
    {
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "function",

```



```

        "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
        "ResourceType": "AWS::Lambda::Function",
        "ExpectedProperties": "{\"Description\": \"Write a file to S3.\",
\\\"Environment\\\": {\\\"Variables\\\": {\\\"bucket\\\": \\\"my-stack-bucket-1vc62xmplgguf
\\\"}}, \\\"Handler\\\": \\\"index.handler\\\", \\\"MemorySize\\\": 128, \\\"Role\\\":
\\\"arn:aws:iam:123456789012:role/my-functionRole-HIZXMPLE0M9E\\\", \\\"Runtime\\\":
\\\"nodejs10.x\\\", \\\"Tags\\\": [{\\\"Key\\\": \\\"lambda:createdBy\\\", \\\"Value\\\": \\\"SAM\\\"}], \\\"Timeout
\\\": 900, \\\"TracingConfig\\\": {\\\"Mode\\\": \\\"Active\\\"}}\",
        \"ActualProperties\": \"{\\\"Description\\\": \\\"Write a file to S3.\",
\\\"Environment\\\": {\\\"Variables\\\": {\\\"bucket\\\": \\\"my-stack-bucket-1vc62xmplgguf
\\\"}}, \\\"Handler\\\": \\\"index.handler\\\", \\\"MemorySize\\\": 256, \\\"Role\\\":
\\\"arn:aws:iam:123456789012:role/my-functionRole-HIZXMPLE0M9E\\\", \\\"Runtime\\\":
\\\"nodejs10.x\\\", \\\"Tags\\\": [{\\\"Key\\\": \\\"lambda:createdBy\\\", \\\"Value\\\": \\\"SAM\\\"}], \\\"Timeout
\\\": 22, \\\"TracingConfig\\\": {\\\"Mode\\\": \\\"Active\\\"}}\",
        \"PropertyDifferences\": [
            {
                \"PropertyPath\": \"/MemorySize\",
                \"ExpectedValue\": \"128\",
                \"ActualValue\": \"256\",
                \"DifferenceType\": \"NOT_EQUAL\"
            },
            {
                \"PropertyPath\": \"/Timeout\",
                \"ExpectedValue\": \"900\",
                \"ActualValue\": \"22\",
                \"DifferenceType\": \"NOT_EQUAL\"
            }
        ],
        \"StackResourceDriftStatus\": \"MODIFIED\",
        \"Timestamp\": \"2019-10-02T05:54:44.064Z\"
    }
]
}

```

- Untuk detail API, lihat [DescribeStackResourceDrifts](#) di Referensi AWS CLI Perintah.

## describe-stack-resource

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-resource`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber daya tumpukan

`describe-stack-resource` Contoh berikut menampilkan rincian untuk sumber daya bernama `MyFunction` dalam tumpukan tertentu.

```
aws cloudformation describe-stack-resource \  
  --stack-name MyStack \  
  --logical-resource-id MyFunction
```

Output:

```
{  
  "StackResourceDetail": {  
    "StackName": "MyStack",  
    "StackId": "arn:aws:cloudformation:us-east-2:123456789012:stack/MyStack/  
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",  
    "LogicalResourceId": "MyFunction",  
    "PhysicalResourceId": "my-function-SEZV4XMPL4S5",  
    "ResourceType": "AWS::Lambda::Function",  
    "LastUpdatedTimestamp": "2019-10-02T05:34:27.989Z",  
    "ResourceStatus": "UPDATE_COMPLETE",  
    "Metadata": "{}",  
    "DriftInformation": {  
      "StackResourceDriftStatus": "IN_SYNC"  
    }  
  }  
}
```

- Untuk detail API, lihat [DescribeStackResource](#) di Referensi AWS CLI Perintah.

## `describe-stack-resources`

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-resources`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber daya tumpukan

`describe-stack-resources` Contoh berikut menampilkan rincian untuk sumber daya dalam tumpukan tertentu.

```
aws cloudformation describe-stack-resources \  
  --stack-name my-stack
```

## Output:

```
{
  "StackResources": [
    {
      "StackName": "my-stack",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "bucket",
      "PhysicalResourceId": "my-stack-bucket-1vc62xmplgguf",
      "ResourceType": "AWS::S3::Bucket",
      "Timestamp": "2019-10-02T04:34:11.345Z",
      "ResourceStatus": "CREATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    },
    {
      "StackName": "my-stack",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "function",
      "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
      "ResourceType": "AWS::Lambda::Function",
      "Timestamp": "2019-10-02T05:34:27.989Z",
      "ResourceStatus": "UPDATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    },
    {
      "StackName": "my-stack",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "functionRole",
      "PhysicalResourceId": "my-functionRole-HIZXMPLE0M9E",
      "ResourceType": "AWS::IAM::Role",
      "Timestamp": "2019-10-02T04:34:06.350Z",
      "ResourceStatus": "CREATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [DescribeStackResources](#) di Referensi AWS CLI Perintah.

## describe-stack-set-operation

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-set-operation`.

### AWS CLI

Untuk mendapatkan informasi tentang operasi stack set

Contoh `describe-stack-set-operation` berikut menampilkan detail untuk operasi pembaruan pada set tumpukan yang ditentukan.

```
aws cloudformation describe-stack-set-operation \  
  --stack-set-name enable-config \  
  --operation-id 35d45ebc-ed88-xmpl-ab59-0197a1fc83a0
```

### Output:

```
{  
  "StackSetOperation": {  
    "OperationId": "35d45ebc-ed88-xmpl-ab59-0197a1fc83a0",  
    "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",  
    "Action": "UPDATE",  
    "Status": "SUCCEEDED",  
    "OperationPreferences": {  
      "RegionOrder": [  
        "us-east-1",  
        "us-west-2",  
        "eu-west-1",  
        "us-west-1"  
      ],  
      "FailureToleranceCount": 7,  
      "MaxConcurrentCount": 2  
    },  
    "AdministrationRoleARN": "arn:aws:iam::123456789012:role/  
AWSCloudFormationStackSetAdministrationRole",  
    "ExecutionRoleName": "AWSCloudFormationStackSetExecutionRole",  
    "CreationTimestamp": "2019-10-03T16:28:44.377Z",  
  },  
}
```

```

      "EndTimeStamp": "2019-10-03T16:42:08.607Z"
    }
  }
}

```

- Untuk detail API, lihat [DescribeStackSetOperation](#) di Referensi AWS CLI Perintah.

## describe-stack-set

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-set`.

### AWS CLI

Untuk mendapatkan informasi tentang kumpulan tumpukan

Contoh `describe-stack-set` berikut menampilkan rincian tentang set stack yang ditentukan.

```

aws cloudformation describe-stack-set \
  --stack-set-name my-stack-set

```

Output:

```

{
  "StackSet": {
    "StackSetName": "my-stack-set",
    "StackSetId": "my-stack-set:296a3360-xmpl-40af-be78-9341e95bf743",
    "Description": "Create an Amazon SNS topic",
    "Status": "ACTIVE",
    "TemplateBody": "AWSTemplateFormatVersion: '2010-09-09'\nDescription: An AWS\nSNS topic\nResources:\n  topic:\n    Type: AWS::SNS::Topic",
    "Parameters": [],
    "Capabilities": [],
    "Tags": [],
    "StackSetARN": "arn:aws:cloudformation:us-west-2:123456789012:stackset/\nenable-config:296a3360-xmpl-40af-be78-9341e95bf743",
    "AdministrationRoleARN": "arn:aws:iam::123456789012:role/\nAWSCloudFormationStackSetAdministrationRole",
    "ExecutionRoleName": "AWSCloudFormationStackSetExecutionRole"
  }
}

```

- Untuk detail API, lihat [DescribeStackSet](#) di Referensi AWS CLI Perintah.

## describe-stacks

Contoh kode berikut menunjukkan cara menggunakan `describe-stacks`.

### AWS CLI

Untuk menggambarkan AWS CloudFormation tumpukan

`describe-stacks` Perintah berikut menunjukkan informasi ringkasan untuk `myteststack` tumpukan:

```
aws cloudformation describe-stacks --stack-name myteststack
```

Output:

```
{
  "Stacks": [
    {
      "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896",
      "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample template showing how to create a publicly accessible S3 bucket. **WARNING** This template creates an S3 bucket. You will be billed for the AWS resources used if you create a stack from this template.",
      "Tags": [],
      "Outputs": [
        {
          "Description": "Name of S3 bucket to hold website content",
          "OutputKey": "BucketName",
          "OutputValue": "myteststack-s3bucket-jssofilzie2w"
        }
      ],
      "StackStatusReason": null,
      "CreationTime": "2013-08-23T01:02:15.422Z",
      "Capabilities": [],
      "StackName": "myteststack",
      "StackStatus": "CREATE_COMPLETE",
      "DisableRollback": false
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Tumpukan di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DescribeStacks](#) di Referensi AWS CLI Perintah.

## describe-type-registration

Contoh kode berikut menunjukkan cara menggunakan `describe-type-registration`.

### AWS CLI

Untuk menampilkan informasi pendaftaran jenis

`describe-type-registration` Contoh berikut menampilkan informasi tentang pendaftaran jenis yang ditentukan, termasuk status, jenis, dan versi tipe saat ini.

```
aws cloudformation describe-type-registration \  
  --registration-token a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "ProgressStatus": "COMPLETE",  
  "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-  
LogGroup",  
  "Description": "Deployment is currently in DEPLOY_STAGE of status COMPLETED; ",  
  "TypeVersionArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/  
My-Logs-LogGroup/00000001"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan CloudFormation Registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DescribeTypeRegistration](#) di Referensi AWS CLI Perintah.

## describe-type

Contoh kode berikut menunjukkan cara menggunakan `describe-type`.

### AWS CLI

Untuk menampilkan informasi jenis

`describe-type` Contoh berikut menampilkan informasi untuk jenis yang ditentukan.

```
aws cloudformation describe-type \  
  --type-name My::Logs::LogGroup \  
  --type RESOURCE
```

Output:

```
{  
  "SourceUrl": "https://github.com/aws-cloudformation/aws-cloudformation-resource-providers-logs.git",  
  "Description": "Customized resource derived from AWS::Logs::LogGroup",  
  "TimeCreated": "2019-12-03T23:29:33.321Z",  
  "Visibility": "PRIVATE",  
  "TypeName": "My::Logs::LogGroup",  
  "LastUpdated": "2019-12-03T23:29:33.321Z",  
  "DeprecatedStatus": "LIVE",  
  "ProvisioningType": "FULLY_MUTABLE",  
  "Type": "RESOURCE",  
  "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup/00000001",  
  "Schema": "[details omitted]"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan CloudFormation Registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DescribeType](#) di Referensi AWS CLI Perintah.

## detect-stack-drift

Contoh kode berikut menunjukkan cara menggunakan `detect-stack-drift`.

AWS CLI

Untuk mendeteksi sumber daya yang melayang

`detect-stack-drift` Contoh berikut memulai deteksi drift untuk tumpukan tertentu.

```
aws cloudformation detect-stack-drift \  
  --stack-name my-stack
```

Output:



```
{
  "StackDriftDetectionId": "1a229160-e4d9-xmpl-ab67-0a4f93df83d4"
}
```

Anda kemudian dapat menggunakan ID ini dengan `describe-stack-resource-drifts` perintah untuk menggambarkan sumber daya yang hanyut.

- Untuk detail API, lihat [DetectStackDrift](#) di Referensi AWS CLI Perintah.

## detect-stack-resource-drift

Contoh kode berikut menunjukkan cara menggunakan `detect-stack-resource-drift`.

### AWS CLI

Untuk mendeteksi penyimpangan untuk sumber daya

`detect-stack-resource-drift` Contoh berikut memeriksa sumber daya bernama `MyFunction` dalam tumpukan bernama `MyStack` drift:

```
aws cloudformation detect-stack-resource-drift \
  --stack-name MyStack \
  --logical-resource-id MyFunction
```

Output menunjukkan fungsi AWS Lambda yang telah dimodifikasi: `out-of-band`

```
{
  "StackResourceDrift": {
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/MyStack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
    "LogicalResourceId": "MyFunction",
    "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
    "ResourceType": "AWS::Lambda::Function",
    "ExpectedProperties": "{\"Description\":\"Write a file to S3.\",
  \"Environment\":{\"Variables\":{\"bucket\":\"my-stack-bucket-1vc62xmplgguf\"}},
  \"Handler\":\"index.handler\", \"MemorySize\":128, \"Role\":
  \"arn:aws:iam:123456789012:role/my-functionRole-HIZXMPLE0M9E\", \"Runtime\":
  \"nodejs10.x\", \"Tags\":[{\"Key\":\"lambda:createdBy\", \"Value\":\"SAM\"}],
  \"Timeout\":900, \"TracingConfig\":{\"Mode\":\"Active\"}}",
    "ActualProperties": "{\"Description\":\"Write a file to S3.\", \"Environment
  \":{\"Variables\":{\"bucket\":\"my-stack-bucket-1vc62xmplgguf\"}}, \"Handler\":
```

```

{"index.handler","\MemorySize\:256","\Role\":"arn:aws:iam::123456789012:role/my-functionRole-HIZXMPLE0M9E","\Runtime\":"nodejs10.x","\Tags\":[{"Key\":"lambda:createdBy","\Value\":"SAM"}],\Timeout\:22,\TracingConfig\":{"Mode\":"Active\"}},
  "PropertyDifferences": [
    {
      "PropertyPath": "/MemorySize",
      "ExpectedValue": "128",
      "ActualValue": "256",
      "DifferenceType": "NOT_EQUAL"
    },
    {
      "PropertyPath": "/Timeout",
      "ExpectedValue": "900",
      "ActualValue": "22",
      "DifferenceType": "NOT_EQUAL"
    }
  ],
  "StackResourceDriftStatus": "MODIFIED",
  "Timestamp": "2019-10-02T05:58:47.433Z"
}
}

```

- Untuk detail API, lihat [DetectStackResourceDrift](#) di Referensi AWS CLI Perintah.

## detect-stack-set-drift

Contoh kode berikut menunjukkan cara menggunakan `detect-stack-set-drift`.

### AWS CLI

Untuk mendeteksi penyimpangan pada kumpulan tumpukan dan semua instance tumpukan terkait

`detect-stack-set-drift` Contoh berikut memulai operasi deteksi drift pada set tumpukan yang ditentukan, termasuk semua instance tumpukan yang terkait dengan kumpulan tumpukan itu, dan mengembalikan ID operasi yang dapat digunakan untuk melacak status operasi drift.

```
aws cloudformation detect-stack-set-drift \
  --stack-set-name stack-set-drift-example
```

Output:

```
{
  "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perubahan Konfigurasi Tidak Terkelola dalam Kumpulan Tumpukan](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DetectStackSetDrift](#) di Referensi AWS CLI Perintah.

## **estimate-template-cost**

Contoh kode berikut menunjukkan cara menggunakan `estimate-template-cost`.

### AWS CLI

Untuk memperkirakan biaya template

`estimate-template-cost` Contoh berikut menghasilkan perkiraan biaya untuk template bernama `template.yaml` dalam folder saat ini.

```
aws cloudformation estimate-template-cost \
  --template-body file://template.yaml
```

Output:

```
{
  "Url": "http://calculator.s3.amazonaws.com/calc5.html?
key=cloudformation/7870825a-xmpl-4def-92e7-c4f8dd360cca"
}
```

- Untuk detail API, lihat [EstimateTemplateCost](#) di Referensi AWS CLI Perintah.

## **execute-change-set**

Contoh kode berikut menunjukkan cara menggunakan `execute-change-set`.

### AWS CLI

Untuk menjalankan set perubahan

`execute-change-set` Contoh berikut mengeksekusi set perubahan yang ditentukan oleh nama set perubahan dan nama stack.

```
aws cloudformation execute-change-set \
  --change-set-name my-change-set \
  --stack-name my-stack
```

`execute-change-set` Contoh berikut mengeksekusi set perubahan yang ditentukan oleh ARN penuh dari set perubahan.

```
aws cloudformation execute-change-set \
  --change-set-name arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
  change-set/bc9555ba-a949-xmpl-bfb8-f41d04ec5784
```

- Untuk detail API, lihat [ExecuteChangeSet](#) di Referensi AWS CLI Perintah.

## get-stack-policy

Contoh kode berikut menunjukkan cara menggunakan `get-stack-policy`.

### AWS CLI

Untuk melihat kebijakan tumpukan

`get-stack-policy` Contoh berikut menampilkan kebijakan tumpukan untuk tumpukan tertentu. Untuk melampirkan kebijakan ke tumpukan, gunakan `set-stack-policy` perintah.

```
aws cloudformation get-stack-policy \
  --stack-name my-stack
```

Output:

```
{
  "StackPolicyBody": "{\n  \"Statement\" : [\n    {\n      \"Effect\" :\n  \"Allow\",\n      \"Action\" : \"Update:*\",\n      \"Principal\" : \"*\",\n      \"Resource\" : \"*\"\n    },\n    {\n      \"Effect\" : \"Deny\",\n      \"Action\" : \"Update:*\",\n      \"Principal\" : \"*\",\n      \"Resource\" :\n  \"LogicalResourceId/bucket\"\n    }\n  ]\n}"
```

- Untuk detail API, lihat [GetStackPolicy](#) di Referensi AWS CLI Perintah.

## get-template-summary

Contoh kode berikut menunjukkan cara menggunakan `get-template-summary`.

### AWS CLI

Untuk menampilkan ringkasan template

Perintah berikut menampilkan informasi ringkasan tentang sumber daya dan metadata untuk file template yang ditentukan.

```
aws cloudformation get-template-summary \  
  --template-body file://template.yaml
```

Output:

```
{  
  "Parameters": [],  
  "Description": "A VPC and subnets.",  
  "ResourceTypes": [  
    "AWS::EC2::VPC",  
    "AWS::EC2::Subnet",  
    "AWS::EC2::Subnet",  
    "AWS::EC2::RouteTable",  
    "AWS::EC2::VPCEndpoint",  
    "AWS::EC2::SubnetRouteTableAssociation",  
    "AWS::EC2::SubnetRouteTableAssociation",  
    "AWS::EC2::VPCEndpoint"  
  ],  
  "Version": "2010-09-09"  
}
```

- Untuk detail API, lihat [GetTemplateSummary](#) di Referensi AWS CLI Perintah.

## get-template

Contoh kode berikut menunjukkan cara menggunakan `get-template`.

### AWS CLI

Untuk melihat badan template untuk AWS CloudFormation tumpukan

`get-template` Perintah berikut menunjukkan template untuk `myteststack` tumpukan:

```
aws cloudformation get-template --stack-name myteststack
```

## Output:

```
{
  "TemplateBody": {
    "AWSTemplateFormatVersion": "2010-09-09",
    "Outputs": {
      "BucketName": {
        "Description": "Name of S3 bucket to hold website content",
        "Value": {
          "Ref": "S3Bucket"
        }
      }
    },
    "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample
template showing how to create a publicly accessible S3 bucket. **WARNING** This
template creates an S3 bucket. You will be billed for the AWS resources used if you
create a stack from this template.",
    "Resources": {
      "S3Bucket": {
        "Type": "AWS::S3::Bucket",
        "Properties": {
          "AccessControl": "PublicRead"
        }
      }
    }
  }
}
```

- Untuk detail API, lihat [GetTemplate](#) di Referensi AWS CLI Perintah.

## list-change-sets

Contoh kode berikut menunjukkan cara menggunakan `list-change-sets`.

### AWS CLI

Untuk membuat daftar set perubahan

`list-change-sets` Contoh berikut menampilkan daftar set perubahan yang tertunda untuk tumpukan tertentu.

```
aws cloudformation list-change-sets \  
  --stack-name my-stack
```

### Output:

```
{  
  "Summaries": [  
    {  
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-  
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",  
      "StackName": "my-stack",  
      "ChangeSetId": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/  
my-change-set/70160340-7914-xmpl-bcbf-128a1fa78b5d",  
      "ChangeSetName": "my-change-set",  
      "ExecutionStatus": "AVAILABLE",  
      "Status": "CREATE_COMPLETE",  
      "CreationTime": "2019-10-02T05:38:54.297Z"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListChangeSets](#) di Referensi AWS CLI Perintah.

## list-exports

Contoh kode berikut menunjukkan cara menggunakan `list-exports`.

### AWS CLI

Untuk daftar ekspor

`list-exports` Contoh berikut menampilkan daftar ekspor dari tumpukan di wilayah saat ini.

```
aws cloudformation list-exports
```

### Output:

```
{  
  "Exports": [  
    {
```

```

      "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
      "Name": "private-vpc-subnet-a",
      "Value": "subnet-07b410xmplddcfa03"
    },
    {
      "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
      "Name": "private-vpc-subnet-b",
      "Value": "subnet-075ed3xmpllebd2fb1"
    },
    {
      "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
      "Name": "private-vpc-vpcid",
      "Value": "vpc-011d7xmpl1100e9841"
    }
  ]
}

```

- Untuk detail API, lihat [ListExports](#) di Referensi AWS CLI Perintah.

## list-imports

Contoh kode berikut menunjukkan cara menggunakan `list-imports`.

### AWS CLI

Untuk daftar impor

`list-imports` Contoh berikut mencantumkan tumpukan yang mengimpor ekspor yang ditentukan. Untuk mendapatkan daftar ekspor yang tersedia, gunakan `list-exports` perintah.

```

aws cloudformation list-imports \
  --export-name private-vpc-vpcid

```

Output:

```

{
  "Imports": [
    "my-database-stack"
  ]
}

```



```
}

```

- Untuk detail API, lihat [ListImports](#) di Referensi AWS CLI Perintah.

## list-stack-instances

Contoh kode berikut menunjukkan cara menggunakan `list-stack-instances`.

### AWS CLI

Untuk membuat daftar instance untuk tumpukan

`list-stack-instances` Contoh berikut mencantumkan instance yang dibuat dari set stack yang ditentukan.

```
aws cloudformation list-stack-instances \
  --stack-set-name enable-config

```

Output contoh mencakup detail tentang tumpukan yang gagal diperbarui karena kesalahan:

```
{
  "Summaries": [
    {
      "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
      "Region": "us-west-2",
      "Account": "123456789012",
      "StackId": "arn:aws:cloudformation:ap-northeast-1:123456789012:stack/StackSet-enable-config-35a6ac50-d9f8-4084-86e4-7da34d5de4c4/a1631cd0-e5fb-xmpl-b474-0aa20f14f06e",
      "Status": "CURRENT"
    },
    {
      "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
      "Region": "us-west-2",
      "Account": "123456789012",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532/eab53680-e5fa-xmpl-ba14-0a522351f81e",
      "Status": "OUTDATED",
      "StatusReason": "ResourceLogicalId:ConfigDeliveryChannel, ResourceType:AWS::Config::DeliveryChannel, ResourceStatusReason:Failed to put delivery channel 'StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532-"
    }
  ]
}
```

```
ConfigDeliveryChannel-10JWJ7XD59WR0' because the maximum number of delivery
channels: 1 is reached. (Service: AmazonConfig; Status Code: 400; Error Code:
MaxNumberOfDeliveryChannelsExceededException; Request ID: d14b34a0-ef7c-xmpl-
acf8-8a864370ae56)."
```

```
    }
  ]
}
```

- Untuk detail API, lihat [ListStackInstances](#) di Referensi AWS CLI Perintah.

## list-stack-resources

Contoh kode berikut menunjukkan cara menggunakan `list-stack-resources`.

### AWS CLI

Untuk membuat daftar sumber daya dalam tumpukan

Perintah berikut menampilkan daftar sumber daya dalam tumpukan yang ditentukan.

```
aws cloudformation list-stack-resources \
  --stack-name my-stack
```

Output:

```
{
  "StackResourceSummaries": [
    {
      "LogicalResourceId": "bucket",
      "PhysicalResourceId": "my-stack-bucket-1vc62xmplgguf",
      "ResourceType": "AWS::S3::Bucket",
      "LastUpdatedTimestamp": "2019-10-02T04:34:11.345Z",
      "ResourceStatus": "CREATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    },
    {
      "LogicalResourceId": "function",
      "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
      "ResourceType": "AWS::Lambda::Function",
      "LastUpdatedTimestamp": "2019-10-02T05:34:27.989Z",
```

```

    "ResourceStatus": "UPDATE_COMPLETE",
    "DriftInformation": {
      "StackResourceDriftStatus": "IN_SYNC"
    }
  },
  {
    "LogicalResourceId": "functionRole",
    "PhysicalResourceId": "my-functionRole-HIZXMPLE0M9E",
    "ResourceType": "AWS::IAM::Role",
    "LastUpdatedTimestamp": "2019-10-02T04:34:06.350Z",
    "ResourceStatus": "CREATE_COMPLETE",
    "DriftInformation": {
      "StackResourceDriftStatus": "IN_SYNC"
    }
  }
]
}

```

- Untuk detail API, lihat [ListStackResources](#) di Referensi AWS CLI Perintah.

## list-stack-set-operation-results

Contoh kode berikut menunjukkan cara menggunakan `list-stack-set-operation-results`.

### AWS CLI

Untuk mencantumkan hasil operasi set tumpukan

Perintah berikut menampilkan hasil operasi pembaruan pada instance dalam kumpulan tumpukan yang ditentukan.

```

aws cloudformation list-stack-set-operation-results \
  --stack-set-name enable-config \
  --operation-id 35d45ebc-ed88-xmpl-ab59-0197a1fc83a0

```

### Output:

```

{
  "Summaries": [
    {
      "Account": "223456789012",
      "Region": "us-west-2",

```

```

    "Status": "SUCCEEDED",
    "AccountGateResult": {
      "Status": "SKIPPED",
      "StatusReason": "Function not found: arn:aws:lambda:eu-
west-1:223456789012:function:AWSCloudFormationStackSetAccountGate"
    }
  },
  {
    "Account": "223456789012",
    "Region": "ap-south-1",
    "Status": "CANCELLED",
    "StatusReason": "Cancelled since failure tolerance has exceeded"
  }
]
}

```

Catatan: SKIPPED Status untuk AccountGateResult diharapkan untuk operasi yang berhasil kecuali Anda membuat fungsi gerbang akun.

- Untuk detail API, lihat [ListStackSetOperationResults](#) di Referensi AWS CLI Perintah.

## list-stack-set-operations

Contoh kode berikut menunjukkan cara menggunakan `list-stack-set-operations`.

### AWS CLI

Untuk membuat daftar operasi set tumpukan

`list-stack-set-operations` Contoh berikut menampilkan daftar operasi terbaru pada set stack yang ditentukan.

```
aws cloudformation list-stack-set-operations \
  --stack-set-name my-stack-set
```

Output:

```
{
  "Summaries": [
    {
      "OperationId": "35d45ebc-ed88-xmpl-ab59-0197a1fc83a0",
      "Action": "UPDATE",

```

```

    "Status": "SUCCEEDED",
    "CreationTimestamp": "2019-10-03T16:28:44.377Z",
    "EndTimestamp": "2019-10-03T16:42:08.607Z"
  },
  {
    "OperationId": "891aa98f-7118-xmpl-00b2-00954d1dd0d6",
    "Action": "UPDATE",
    "Status": "FAILED",
    "CreationTimestamp": "2019-10-03T15:43:53.916Z",
    "EndTimestamp": "2019-10-03T15:45:58.925Z"
  }
]
}

```

- Untuk detail API, lihat [ListStackSetOperations](#) di Referensi AWS CLI Perintah.

## list-stack-sets

Contoh kode berikut menunjukkan cara menggunakan `list-stack-sets`.

### AWS CLI

Untuk membuat daftar set tumpukan

`list-stack-sets` Contoh berikut menampilkan daftar set tumpukan di wilayah dan akun saat ini.

```
aws cloudformation list-stack-sets
```

Output:

```

{
  "Summaries": [
    {
      "StackSetName": "enable-config",
      "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
      "Description": "Enable AWS Config",
      "Status": "ACTIVE"
    }
  ]
}

```

- Untuk detail API, lihat [ListStackSets](#) di Referensi AWS CLI Perintah.

## list-stacks

Contoh kode berikut menunjukkan cara menggunakan `list-stacks`.

### AWS CLI

Untuk daftar AWS CloudFormation tumpukan

`list-stacks` Perintah berikut menunjukkan ringkasan semua tumpukan yang memiliki status `CREATE_COMPLETE`:

```
aws cloudformation list-stacks --stack-status-filter CREATE_COMPLETE
```

Output:

```
[
  {
    "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896",
    "TemplateDescription": "AWS CloudFormation Sample Template S3_Bucket: Sample
template showing how to create a publicly accessible S3 bucket. **WARNING** This
template creates an S3 bucket. You will be billed for the AWS resources used if you
create a stack from this template.",
    "StackStatusReason": null,
    "CreationTime": "2013-08-26T03:27:10.190Z",
    "StackName": "myteststack",
    "StackStatus": "CREATE_COMPLETE"
  }
]
```

- Untuk detail API, lihat [ListStacks](#) di Referensi AWS CLI Perintah.

## list-type-registrations

Contoh kode berikut menunjukkan cara menggunakan `list-type-registrations`.

### AWS CLI

Untuk membuat daftar pendaftaran lengkap dari suatu jenis

`list-type-registrations` Contoh berikut menampilkan daftar pendaftaran tipe lengkap untuk jenis yang ditentukan.

```
aws cloudformation list-type-registrations \  
  --type RESOURCE \  
  --type-name My::Logs::LogGroup \  
  --registration-status-filter COMPLETE
```

Output:

```
{  
  "RegistrationTokenList": [  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan CloudFormation Registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [ListTypeRegistrations](#) di Referensi AWS CLI Perintah.

## list-type-versions

Contoh kode berikut menunjukkan cara menggunakan `list-type-versions`.

AWS CLI

Untuk membuat daftar versi ekstensi

`list-type-versions` Contoh berikut mengembalikan informasi ringkasan tentang versi ekstensi.

```
aws cloudformation list-type-versions \  
  --endpoint https://example.com \  
  --region us-west-2 \  
  --type RESOURCE \  
  --type-name My::Resource::Example \  
  --publisher-id 123456789012
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [ListTypeVersions](#) di Referensi AWS CLI Perintah.

## list-types

Contoh kode berikut menunjukkan cara menggunakan `list-types`.

### AWS CLI

Untuk mencantumkan jenis sumber daya pribadi di akun

`list-types` Contoh berikut menampilkan daftar jenis sumber daya pribadi yang saat ini terdaftar di AWS akun saat ini.

```
aws cloudformation list-types
```

Output:

```
{
  "TypeSummaries": [
    {
      "Description": "WordPress blog resource for internal use",
      "LastUpdated": "2019-12-04T18:28:15.059Z",
      "TypeName": "My::WordPress::BlogExample",
      "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-WordPress-BlogExample",
      "DefaultVersionId": "00000005",
      "Type": "RESOURCE"
    },
    {
      "Description": "Customized resource derived from AWS::Logs::LogGroup",
      "LastUpdated": "2019-12-04T18:28:15.059Z",
      "TypeName": "My::Logs::LogGroup",
      "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup",
      "DefaultVersionId": "00000003",
      "Type": "RESOURCE"
    }
  ]
}
```



```
]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan CloudFormation Registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [ListTypes](#) di Referensi AWS CLI Perintah.

## package

Contoh kode berikut menunjukkan cara menggunakan `package`.

### AWS CLI

Perintah berikut mengekspor template bernama `template.json` dengan mengunggah artefak lokal ke bucket S3 `bucket-name` dan menulis template yang diekspor ke: `packaged-template.json`

```
aws cloudformation package --template-file /path_to_template/template.json --s3-bucket bucket-name --output-template-file packaged-template.json --use-json
```

- Untuk detail API, lihat [Package](#) in AWS CLI Command Reference.

## publish-type

Contoh kode berikut menunjukkan cara menggunakan `publish-type`.

### AWS CLI

Untuk mempublikasikan ekstensi

`publish-type` Contoh berikut menerbitkan ekstensi yang ditentukan ke CloudFormation registri sebagai ekstensi publik di Wilayah ini.

```
aws cloudformation publish-type \  
  --region us-west-2 \  
  --type RESOURCE \  
  --type-name Example::Test::1234567890abcdef0
```

Output:

```
{
  "PublicTypeArn": "arn:aws:cloudformation:us-west-2::type/
resource/000q6TfUovXsEMmgKowxDZLLwqr2QUshd2e75c8c/Example-
Test-1234567890abcdef0/1.0.0"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [PublishType](#) di Referensi AWS CLI Perintah.

## register-publisher

Contoh kode berikut menunjukkan cara menggunakan `register-publisher`.

### AWS CLI

Untuk mendaftarkan penerbit

`register-publisher` Contoh berikut mendaftarkan penerbit dan menerima parameter syarat dan kondisi.

```
aws cloudformation register-publisher \
  --region us-west-2 \
  --accept-terms-and-conditions
```

Output:

```
{
  "PublisherId": "000q6TfUovXsEMmgKowxDZLLwqr2QUshd2e75c8c"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [RegisterPublisher](#) di Referensi AWS CLI Perintah.

## register-type

Contoh kode berikut menunjukkan cara menggunakan `register-type`.

## AWS CLI

Untuk mendaftarkan jenis sumber daya

`register-type` Contoh berikut mendaftarkan jenis sumber daya tertentu sebagai jenis sumber daya pribadi di akun pengguna.

```
aws cloudformation register-type \  
  --type-name My::Organization::ResourceName \  
  --schema-handler-package s3://bucket_name/my-organization-resource_name.zip \  
  --type RESOURCE
```

Output:

```
{  
  "RegistrationToken": "f5525280-104e-4d35-bef5-8f1f1example"  
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Penyedia Sumber Daya](#) di Panduan Pengguna Antarmuka Baris CloudFormation Perintah untuk Pengembangan Jenis.

- Untuk detail API, lihat [RegisterType](#) di Referensi AWS CLI Perintah.

## set-stack-policy

Contoh kode berikut menunjukkan cara menggunakan `set-stack-policy`.

## AWS CLI

Untuk menerapkan kebijakan tumpukan

`set-stack-policy` Contoh berikut menonaktifkan pembaruan untuk sumber daya yang ditentukan dalam tumpukan yang ditentukan. `stack-policy.json` adalah dokumen JSON yang mendefinisikan operasi yang diizinkan pada sumber daya di tumpukan.

```
aws cloudformation set-stack-policy \  
  --stack-name my-stack \  
  --stack-policy-body file://stack-policy.json
```

Output:

```
{
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : "Update:*",
      "Principal": "*",
      "Resource" : "*"
    },
    {
      "Effect" : "Deny",
      "Action" : "Update:*",
      "Principal": "*",
      "Resource" : "LogicalResourceId/bucket"
    }
  ]
}
```

- Untuk detail API, lihat [SetStackPolicy](#) di Referensi AWS CLI Perintah.

## set-type-configuration

Contoh kode berikut menunjukkan cara menggunakan `set-type-configuration`.

### AWS CLI

Untuk mengkonfigurasi data

`set-type-configuration` Contoh berikut menentukan data konfigurasi untuk CloudFormation ekstensi terdaftar, di akun yang diberikan dan Wilayah.

```
aws cloudformation set-type-configuration \
  --region us-west-2 \
  --type RESOURCE \
  --type-name Example::Test::Type \
  --configuration-alias default \
  --configuration "{\"CredentialKey\": \"testUserCredential\"}"
```

Output:

```
{
  "ConfigurationArn": "arn:aws:cloudformation:us-west-2:123456789012:type-configuration/resource/Example-Test-Type/default"
```

```
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [SetTypeConfiguration](#) di Referensi AWS CLI Perintah.

## set-type-default-version

Contoh kode berikut menunjukkan cara menggunakan `set-type-default-version`.

### AWS CLI

Untuk menyetel versi default tipe

`set-type-default-version` Contoh berikut menetapkan versi tipe tertentu yang akan digunakan sebagai default untuk jenis ini.

```
aws cloudformation set-type-default-version \  
  --type RESOURCE \  
  --type-name My::Logs::LogGroup \  
  --version-id 00000003
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan CloudFormation Registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [SetTypeDefaultVersion](#) di Referensi AWS CLI Perintah.

## signal-resource

Contoh kode berikut menunjukkan cara menggunakan `signal-resource`.

### AWS CLI

Untuk memberi sinyal sumber daya

`signal-resource` Contoh berikut sinyal success untuk memenuhi kondisi tunggu bernama `MyWaitCondition` dalam tumpukan bernama `my-stack`.

```
aws cloudformation signal-resource \  
  --stack-name my-stack \  
  --logical-resource-id MyWaitCondition \  
  --unique-id 1234 \  
  --status SUCCESS
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SignalResource](#) di Referensi AWS CLI Perintah.

## stop-stack-set-operation

Contoh kode berikut menunjukkan cara menggunakan `stop-stack-set-operation`.

### AWS CLI

Untuk menghentikan operasi set tumpukan

`stop-stack-set-operation` Contoh berikut menghentikan operasi pembaruan dalam progress pada set tumpukan yang ditentukan.

```
aws cloudformation stop-stack-set-operation \  
  --stack-set-name my-stack-set \  
  --operation-id 1261cd27-490b-xmpl-ab42-793a896c69e6
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [StopStackSetOperation](#) di Referensi AWS CLI Perintah.

## test-type

Contoh kode berikut menunjukkan cara menggunakan `test-type`.

### AWS CLI

Untuk menguji ekstensi

`test-type` Contoh berikut menguji ekstensi terdaftar untuk memastikannya memenuhi semua persyaratan yang diperlukan untuk diterbitkan dalam CloudFormation registri.

```
aws cloudformation test-type \  
  --stack-name my-stack
```

```
--arn arn:aws:cloudformation:us-west-2:123456789012:type/resource/Sample-Test-Resource123/00000001
```

Output:

```
{
  "TypeVersionArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/Sample-Test-Resource123/00000001"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS CloudFormation registri](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [TestType](#) di Referensi AWS CLI Perintah.

## update-stack-instances

Contoh kode berikut menunjukkan cara menggunakan `update-stack-instances`.

### AWS CLI

Untuk memperbarui instance tumpukan

`update-stack-instances` Contoh berikut mencoba ulang pembaruan pada instance tumpukan di dua akun di dua wilayah dengan pengaturan terbaru. Pengaturan toleransi kesalahan yang ditentukan memastikan bahwa pembaruan dicoba di semua akun dan wilayah, bahkan jika beberapa tumpukan tidak dapat diperbarui.

```
aws cloudformation update-stack-instances \
  --stack-set-name my-stack-set \
  --accounts 123456789012 567890123456 \
  --regions us-east-1 us-west-2 \
  --operation-preferences FailureToleranceCount=3
```

Output:

```
{
  "OperationId": "103ebdf2-21ea-xmpl-8892-de5e30733132"
}
```

- Untuk detail API, lihat [UpdateStackInstances](#) di Referensi AWS CLI Perintah.

## update-stack-set

Contoh kode berikut menunjukkan cara menggunakan `update-stack-set`.

### AWS CLI

Untuk memperbarui set tumpukan

`update-stack-set` Contoh berikut menambahkan tag dengan nama kunci `Owner` dan nilai `IT` untuk instance stack dalam set stack yang ditentukan.

```
aws cloudformation update-stack-set \  
  --stack-set-name my-stack-set \  
  --use-previous-template \  
  --tags Key=Owner,Value=IT
```

Output:

```
{  
  "OperationId": "e2b60321-6cab-xmpl-bde7-530c6f47950e"  
}
```

- Untuk detail API, lihat [UpdateStackSet](#) di Referensi AWS CLI Perintah.

## update-stack

Contoh kode berikut menunjukkan cara menggunakan `update-stack`.

### AWS CLI

Untuk memperbarui AWS CloudFormation tumpukan

`update-stack` Perintah berikut memperbarui template dan parameter input untuk `mystack` tumpukan:

```
aws cloudformation update-stack --stack-name mystack --  
  template-url https://s3.amazonaws.com/sample/updated.template --
```



```
parameters ParameterKey=KeyPairName,ParameterValue=SampleKeyPair
ParameterKey=SubnetIDs,ParameterValue=SampleSubnetID1\\,SampleSubnetID2
```

`update-stack` Perintah berikut memperbarui hanya nilai `SubnetIDs` parameter untuk `mystack` tumpukan. Jika Anda tidak menentukan nilai parameter, nilai default yang ditentukan dalam template digunakan:

```
aws cloudformation update-stack --stack-name mystack --
template-url https://s3.amazonaws.com/sample/updated.template
--parameters ParameterKey=KeyPairName,UsePreviousValue=true
ParameterKey=SubnetIDs,ParameterValue=SampleSubnetID1\\,UpdatedSampleSubnetID2
```

`update-stack` Perintah berikut menambahkan dua topik notifikasi `mystack` tumpukan ke tumpukan:

```
aws cloudformation update-stack --stack-name mystack --use-previous-template --
notification-arns "arn:aws:sns:use-east-1:123456789012:mytopic1" "arn:aws:sns:us-
east-1:123456789012:mytopic2"
```

Untuk informasi selengkapnya, lihat [AWS CloudFormation tumpukan pembaruan](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [UpdateStack](#) di Referensi AWS CLI Perintah.

## update-termination-protection

Contoh kode berikut menunjukkan cara menggunakan `update-termination-protection`.

### AWS CLI

Untuk mengaktifkan perlindungan terminasi

`update-termination-protection` Contoh berikut memungkinkan perlindungan terminasi pada tumpukan yang ditentukan.

```
aws cloudformation update-termination-protection \
--stack-name my-stack \
--enable-termination-protection
```

Output:

```
{
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204"
}
```

- Untuk detail API, lihat [UpdateTerminationProtection](#) di Referensi AWS CLI Perintah.

## validate-template

Contoh kode berikut menunjukkan cara menggunakan `validate-template`.

### AWS CLI

Untuk memvalidasi template AWS CloudFormation

`validate-template` Perintah berikut memvalidasi `sampletemplate.json` template:

```
aws cloudformation validate-template --template-body file://sampletemplate.json
```

Output:

```
{
  "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample template
showing how to create a publicly accessible S3 bucket. **WARNING** This template
creates an S3 bucket. You will be billed for the AWS resources used if you create a
stack from this template.",
  "Parameters": [],
  "Capabilities": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS CloudFormation Template](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [ValidateTemplate](#) di Referensi AWS CLI Perintah.

## CloudFront contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CloudFront.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-cloud-front-origin-access-identity**

Contoh kode berikut menunjukkan cara menggunakan `create-cloud-front-origin-access-identity`.

AWS CLI

Untuk membuat identitas akses CloudFront asal

Contoh berikut membuat identitas akses CloudFront asal (OAI) dengan menyediakan konfigurasi OAI sebagai argumen baris perintah:

```
aws cloudfront create-cloud-front-origin-access-identity \  
  --cloud-front-origin-access-identity-config \  
    CallerReference="cli-example",Comment="Example OAI"
```

Anda dapat mencapai hal yang sama dengan menyediakan konfigurasi OAI dalam file JSON, seperti yang ditunjukkan pada contoh berikut:

```
aws cloudfront create-cloud-front-origin-access-identity \  
  --cloud-front-origin-access-identity-config file://OAI-config.json
```

File tersebut `OAI-config.json` adalah dokumen JSON di direktori saat ini yang berisi berikut ini:

```
{
  "CallerReference": "cli-example",
  "Comment": "Example OAI"
}
```

Apakah Anda memberikan konfigurasi OAI dengan argumen baris perintah atau file JSON, outputnya sama:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/origin-access-identity/cloudfront/E74FTE3AEXAMPLE",
  "ETag": "E2QWRUHEXAMPLE",
  "CloudFrontOriginAccessIdentity": {
    "Id": "E74FTE3AEXAMPLE",
    "S3CanonicalUserId":
    "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example OAI"
    }
  }
}
```

- Untuk detail API, lihat [CreateCloudFrontOriginAccessIdentity](#) di Referensi AWS CLI Perintah.

## create-distribution-with-tags

Contoh kode berikut menunjukkan cara menggunakan `create-distribution-with-tags`.

### AWS CLI

Untuk membuat CloudFront distribusi dengan tag

Contoh berikut membuat distribusi dengan dua tag dengan menyediakan konfigurasi distribusi dan tag dalam file JSON bernama `dist-config-with-tags.json`:

```
aws cloudfront create-distribution-with-tags \
  --distribution-config-with-tags file://dist-config-with-tags.json
```

File tersebut `dist-config-with-tags.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini. Perhatikan Tags objek di bagian atas file, yang berisi dua tag:

Name = ExampleDistributionProject = ExampleProject

```
{
  "Tags": {
    "Items": [
      {
        "Key": "Name",
        "Value": "ExampleDistribution"
      },
      {
        "Key": "Project",
        "Value": "ExampleProject"
      }
    ]
  },
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
          "DomainName": "awsexamplebucket.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          },
          "S3OriginConfig": {
            "OriginAccessIdentity": ""
          }
        }
      ]
    },
    "OriginGroups": {
      "Quantity": 0
    },
    "DefaultCacheBehavior": {
      "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
      "ForwardedValues": {
        "QueryString": false,

```

```
    "Cookies": {
      "Forward": "none"
    },
    "Headers": {
      "Quantity": 0
    },
    "QueryStringCacheKeys": {
      "Quantity": 0
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ],
    "CachedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ]
    }
  },
  "SmoothStreaming": false,
  "DefaultTTL": 86400,
  "MaxTTL": 31536000,
  "Compress": false,
  "LambdaFunctionAssociations": {
    "Quantity": 0
  },
  "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
  "Quantity": 0
},
"CustomErrorResponses": {
  "Quantity": 0
```

```

    },
    "Comment": "",
    "Logging": {
      "Enabled": false,
      "IncludeCookies": false,
      "Bucket": "",
      "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}

```

### Output:

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/EDFDVBD6EXAMPLE",
  "ETag": "E2QWRUHEXAMPLE",
  "Distribution": {
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-12-04T23:35:41.433Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d1111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    }
  }
}

```

```
},
"DistributionConfig": {
  "CallerReference": "cli-example",
  "Aliases": {
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
        "S3OriginConfig": {
          "OriginAccessIdentity": ""
        }
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    }
  },
}
```



```
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponses": {
    "Quantity": 0
  },
  "Comment": "",
  "Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
  },
  "PriceClass": "PriceClass_All",
  "Enabled": true,
  "ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
```

```

    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}

```

- Untuk detail API, lihat [CreateDistributionWithTags](#) di Referensi AWS CLI Perintah.

## create-distribution

Contoh kode berikut menunjukkan cara menggunakan `create-distribution`.

### AWS CLI

Untuk membuat CloudFront distribusi

Contoh berikut membuat distribusi untuk bucket S3 bernama `awsexamplebucket`, dan juga menentukan `index.html` sebagai objek root default, menggunakan argumen baris perintah:

```

aws cloudfront create-distribution \
  --origin-domain-name awsexamplebucket.s3.amazonaws.com \
  --default-root-object index.html

```

Alih-alih menggunakan argumen baris perintah, Anda dapat memberikan konfigurasi distribusi dalam file JSON, seperti yang ditunjukkan pada contoh berikut:

```

aws cloudfront create-distribution \
  --distribution-config file://dist-config.json

```

File tersebut `dist-config.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini:

```

{
  "CallerReference": "cli-example",
  "Aliases": {

```

```
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
        "S3OriginConfig": {
          "OriginAccessIdentity": ""
        }
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
```

```
    "Items": [
      "HEAD",
      "GET"
    ],
    "CachedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ]
    }
  },
  "SmoothStreaming": false,
  "DefaultTTL": 86400,
  "MaxTTL": 31536000,
  "Compress": false,
  "LambdaFunctionAssociations": {
    "Quantity": 0
  },
  "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
  "Quantity": 0
},
"CustomErrorResponses": {
  "Quantity": 0
},
"Comment": "",
"Logging": {
  "Enabled": false,
  "IncludeCookies": false,
  "Bucket": "",
  "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
  "CloudFrontDefaultCertificate": true,
  "MinimumProtocolVersion": "TLSv1",
  "CertificateSource": "cloudfront"
},
"Restrictions": {
  "GeoRestriction": {
    "RestrictionType": "none",
```

```

        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}

```

Apakah Anda memberikan informasi distribusi dengan argumen baris perintah atau file JSON, outputnya sama:

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EMLARXS9EXAMPLE",
  "ETag": "E9LHASXEXAMPLE",
  "Distribution": {
    "Id": "EMLARXS9EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-11-22T00:55:15.705Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
  },
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
      },
    ],
  },
  "S3OriginConfig": {

```

```
        "OriginAccessIdentity": ""
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
```

```

        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
    }
}

```

- Untuk detail API, lihat [CreateDistribution](#) di Referensi AWS CLI Perintah.

## create-field-level-encryption-config

Contoh kode berikut menunjukkan cara menggunakan `create-field-level-encryption-config`.

### AWS CLI

Untuk membuat konfigurasi CloudFront enkripsi tingkat lapangan

Contoh berikut membuat konfigurasi enkripsi tingkat lapangan dengan menyediakan parameter konfigurasi dalam file JSON bernama `fle-config.json`. Sebelum Anda dapat membuat konfigurasi enkripsi tingkat lapangan, Anda harus memiliki profil enkripsi tingkat lapangan. Untuk membuat profil, lihat perintah `create-field-level-encryption-profile`.

Untuk informasi selengkapnya tentang enkripsi CloudFront tingkat lapangan, lihat [Menggunakan Enkripsi Tingkat Bidang untuk Membantu Melindungi Data Sensitif di Panduan Pengembang Amazon. CloudFront](#)

```
aws cloudfront create-field-level-encryption-config \  
  --field-level-encryption-config file://fle-config.json
```

File tersebut `fle-config.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini:

```
{  
  "CallerReference": "cli-example",  
  "Comment": "Example FLE configuration",  
  "QueryArgProfileConfig": {  
    "ForwardWhenQueryArgProfileIsUnknown": true,  
    "QueryArgProfiles": {  
      "Quantity": 0  
    }  
  },  
  "ContentTypeProfileConfig": {  
    "ForwardWhenContentTypeIsUnknown": true,  
    "ContentTypeProfiles": {  
      "Quantity": 1,  
      "Items": [  
        {  
          "Format": "URLEncoded",  
          "ProfileId": "P280MFCLSY0CVU",  
          "ContentType": "application/x-www-form-urlencoded"  
        }  
      ]  
    }  
  }  
}
```



```

    }
  ]
}

```

### Output:

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/field-level-encryption/
C3KM2WVD605UAY",
  "ETag": "E2P4Z4VU7TY5SG",
  "FieldLevelEncryption": {
    "Id": "C3KM2WVD605UAY",
    "LastModifiedTime": "2019-12-10T21:30:18.974Z",
    "FieldLevelEncryptionConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example FLE configuration",
      "QueryArgProfileConfig": {
        "ForwardWhenQueryArgProfileIsUnknown": true,
        "QueryArgProfiles": {
          "Quantity": 0,
          "Items": []
        }
      },
      "ContentTypeProfileConfig": {
        "ForwardWhenContentTypeIsUnknown": true,
        "ContentTypeProfiles": {
          "Quantity": 1,
          "Items": [
            {
              "Format": "URLEncoded",
              "ProfileId": "P280MFCLSY0CVU",
              "ContentType": "application/x-www-form-urlencoded"
            }
          ]
        }
      }
    }
  }
}

```

- Untuk detail API, lihat [CreateFieldLevelEncryptionConfig](#) di Referensi AWS CLI Perintah.

## create-field-level-encryption-profile

Contoh kode berikut menunjukkan cara menggunakan `create-field-level-encryption-profile`.

### AWS CLI

Untuk membuat profil CloudFront enkripsi tingkat lapangan

Contoh berikut membuat profil enkripsi tingkat lapangan dengan menyediakan parameter dalam file JSON bernama `fle-profile-config.json`. Sebelum Anda dapat membuat profil enkripsi tingkat lapangan, Anda harus memiliki CloudFront kunci publik. Untuk membuat kunci CloudFront publik, lihat `create-public-key` perintah.

Untuk informasi selengkapnya tentang enkripsi CloudFront tingkat lapangan, lihat [Menggunakan Enkripsi Tingkat Bidang untuk Membantu Melindungi Data Sensitif di Panduan Pengembang Amazon CloudFront](#)

```
aws cloudfront create-field-level-encryption-profile \  
  --field-level-encryption-profile-config file://fle-profile-config.json
```

File tersebut `fle-profile-config.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini:

```
{  
  "Name": "ExampleFLEProfile",  
  "CallerReference": "cli-example",  
  "Comment": "FLE profile for AWS CLI example",  
  "EncryptionEntities": {  
    "Quantity": 1,  
    "Items": [  
      {  
        "PublicKeyId": "K2K8NC4HVFE3M0",  
        "ProviderId": "ExampleFLEProvider",  
        "FieldPatterns": {  
          "Quantity": 1,  
          "Items": [  
            "ExampleSensitiveField"  
          ]  
        }  
      }  
    ]  
  }  
}
```

```

    ]
  }
}

```

Output:

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/field-level-encryption-
profile/PPK0UOSIF5WSV",
  "ETag": "E2QWRUHEXAMPLE",
  "FieldLevelEncryptionProfile": {
    "Id": "PPK0UOSIF5WSV",
    "LastModifiedTime": "2019-12-10T01:03:16.537Z",
    "FieldLevelEncryptionProfileConfig": {
      "Name": "ExampleFLEProfile",
      "CallerReference": "cli-example",
      "Comment": "FLE profile for AWS CLI example",
      "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
          {
            "PublicKeyId": "K2K8NC4HVFE3M0",
            "ProviderId": "ExampleFLEProvider",
            "FieldPatterns": {
              "Quantity": 1,
              "Items": [
                "ExampleSensitiveField"
              ]
            }
          }
        ]
      }
    }
  }
}

```

- Untuk detail API, lihat [CreateFieldLevelEncryptionProfile](#) di Referensi AWS CLI Perintah.

## create-invalidation

Contoh kode berikut menunjukkan cara menggunakan `create-invalidation`.

## AWS CLI

Untuk membuat pembatalan untuk distribusi CloudFront

`create-invalidation` Contoh berikut membuat pembatalan untuk file tertentu dalam distribusi yang ditentukan CloudFront :

```
aws cloudfront create-invalidation \  
  --distribution-id EDFDVBD6EXAMPLE \  
  --paths "/example-path/example-file.jpg" "/example-path/example-file2.png"
```

Output:

```
{  
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/  
EDFDVBD6EXAMPLE/invalidation/I1JLWSDAP8FU89",  
  "Invalidation": {  
    "Id": "I1JLWSDAP8FU89",  
    "Status": "InProgress",  
    "CreateTime": "2019-12-05T18:24:51.407Z",  
    "InvalidationBatch": {  
      "Paths": {  
        "Quantity": 2,  
        "Items": [  
          "/example-path/example-file2.png",  
          "/example-path/example-file.jpg"  
        ]  
      },  
      "CallerReference": "cli-1575570291-670203"  
    }  
  }  
}
```

Pada contoh sebelumnya, AWS CLI secara otomatis menghasilkan acak. `CallerReference` Untuk menentukan sendiri `CallerReference`, atau untuk menghindari meneruskan parameter pembatalan sebagai argumen baris perintah, Anda dapat menggunakan file JSON. Contoh berikut membuat pembatalan untuk dua file, dengan menyediakan parameter pembatalan dalam file JSON bernama: `inv-batch.json`

```
aws cloudfront create-invalidation \  
  --distribution-id EDFDVBD6EXAMPLE \  
  --cli-input-file inv-batch.json
```

```
--invalidation-batch file://inv-batch.json
```

Isi dari `inv-batch.json`:

```
{
  "Paths": {
    "Quantity": 2,
    "Items": [
      "/example-path/example-file.jpg",
      "/example-path/example-file2.png"
    ]
  },
  "CallerReference": "cli-example"
}
```

Output:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/EDFDVBD6EXAMPLE/invalidation/I2J0I21PCUY0IK",
  "Invalidation": {
    "Id": "I2J0I21PCUY0IK",
    "Status": "InProgress",
    "CreateTime": "2019-12-05T18:40:49.413Z",
    "InvalidationBatch": {
      "Paths": {
        "Quantity": 2,
        "Items": [
          "/example-path/example-file.jpg",
          "/example-path/example-file2.png"
        ]
      },
      "CallerReference": "cli-example"
    }
  }
}
```

- Untuk detail API, lihat [CreateInvalidation](#) di Referensi AWS CLI Perintah.

## create-public-key

Contoh kode berikut menunjukkan cara menggunakan `create-public-key`.

## AWS CLI

Untuk membuat kunci CloudFront publik

Contoh berikut membuat kunci CloudFront publik dengan menyediakan parameter dalam file JSON bernama `pub-key-config.json`. Sebelum Anda dapat menggunakan perintah ini, Anda harus memiliki kunci publik yang dikodekan PEM. Untuk informasi selengkapnya, lihat [Membuat Pasangan Kunci RSA](#) di Panduan CloudFront Pengembang Amazon.

```
aws cloudfront create-public-key \
  --public-key-config file://pub-key-config.json
```

File tersebut `pub-key-config.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini. Perhatikan bahwa kunci publik dikodekan dalam format PEM.

```
{
  "CallerReference": "cli-example",
  "Name": "ExampleKey",
  "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAA0CAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LFYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
  "Comment": "example public key"
}
```

Output:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/public-key/
KDFB19YGCR002",
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKey": {
    "Id": "KDFB19YGCR002",
    "CreatedTime": "2019-12-05T18:51:43.781Z",
    "PublicKeyConfig": {
      "CallerReference": "cli-example",
      "Name": "ExampleKey",
```

```

    "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAxPMbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBaz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesp1c0kjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLK0X5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
    "Comment": "example public key"
  }
}
}

```

- Untuk detail API, lihat [CreatePublicKey](#) di Referensi AWS CLI Perintah.

## delete-cloud-front-origin-access-identity

Contoh kode berikut menunjukkan cara menggunakan `delete-cloud-front-origin-access-identity`.

### AWS CLI

Untuk menghapus identitas akses CloudFront asal

Contoh berikut menghapus identitas akses asal (OAI) dengan ID. E74FTE3AEXAMPLE Untuk menghapus OAI, Anda harus memiliki ID OAI dan. ETag ID OAI dikembalikan dalam output perintah `-access-identity` dan `create-cloud-front-origin-access-identities`. `list-cloud-front-origin` Untuk mendapatkan ETag, gunakan perintah `get-cloud-front-origin-access-identity` atau `get-cloud-front-origin-access-identity-config` Gunakan `--if-match` opsi untuk menyediakan OAI. ETag

```

aws cloudfront delete-cloud-front-origin-access-identity \
  --id E74FTE3AEXAMPLE \
  --if-match E2QWRUHEXAMPLE

```

Ketika berhasil, perintah ini tidak memiliki output.

- Untuk detail API, lihat [DeleteCloudFrontOriginAccessIdentity](#) di Referensi AWS CLI Perintah.

## delete-distribution

Contoh kode berikut menunjukkan cara menggunakan `delete-distribution`.

## AWS CLI

Untuk menghapus CloudFront distribusi

Contoh berikut menghapus CloudFront distribusi dengan IDEDFDVBD6EXAMPLE. Sebelum Anda dapat menghapus distribusi, Anda harus menonaktifkannya. Untuk menonaktifkan distribusi, gunakan perintah pembaruan-distribusi. Untuk informasi selengkapnya, lihat contoh distribusi pembaruan.

Ketika distribusi dinonaktifkan, Anda dapat menghapusnya. Untuk menghapus distribusi, Anda harus menggunakan `--if-match` opsi untuk menyediakan distribusiETag. Untuk mendapatkanETag, gunakan `get-distribusi` atau `get-distribution-config` perintah.

```
aws cloudfront delete-distribution \  
  --id EDFDVBD6EXAMPLE \  
  --if-match E2QWRUHEXAMPLE
```

Ketika berhasil, perintah ini tidak memiliki output.

- Untuk detail API, lihat [DeleteDistribution](#) di Referensi AWS CLI Perintah.

## **delete-field-level-encryption-config**

Contoh kode berikut menunjukkan cara menggunakan `delete-field-level-encryption-config`.

## AWS CLI

Untuk menghapus konfigurasi CloudFront enkripsi tingkat lapangan

Contoh berikut menghapus konfigurasi enkripsi CloudFront tingkat lapangan dengan ID. C3KM2WVD605UAY Untuk menghapus konfigurasi enkripsi tingkat lapangan, Anda harus memiliki ID dan. ETag ID dikembalikan dalam output perintah `create-field-level-encryption -config` dan `list-field-level-encryption -configs`. Untuk mendapatkanETag, gunakan perintah `get-field-level-encryption` or `get-field-level-encryption -config`. Gunakan `--if-match` opsi untuk menyediakan konfigurasiETag.

```
aws cloudfront delete-field-level-encryption-config \  
  --id C3KM2WVD605UAY \  
  --if-match E26M4BIAV81ZF6
```



Ketika berhasil, perintah ini tidak memiliki output.

- Untuk detail API, lihat [DeleteFieldLevelEncryptionConfig](#) di Referensi AWS CLI Perintah.

## **delete-field-level-encryption-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-field-level-encryption-profile`.

### AWS CLI

Untuk menghapus profil CloudFront enkripsi tingkat lapangan

Contoh berikut menghapus profil enkripsi CloudFront tingkat lapangan dengan ID. PPK0U0SIF5WSV Untuk menghapus profil enkripsi tingkat lapangan, Anda harus memiliki ID dan. ETag ID dikembalikan dalam output perintah `create-field-level-encryption-profile` dan `list-field-level-encryption-profiles`. Untuk mendapatkan ETag, gunakan perintah `get-field-level-encryption-profile` atau `get-field-level-encryption-profile-config`. Gunakan `--if-match` opsi untuk menyediakan profil ETag.

```
aws cloudfront delete-field-level-encryption-profile \  
  --id PPK0U0SIF5WSV \  
  --if-match EJETYFJ9CL66D
```

Ketika berhasil, perintah ini tidak memiliki output.

- Untuk detail API, lihat [DeleteFieldLevelEncryptionProfile](#) di Referensi AWS CLI Perintah.

## **delete-public-key**

Contoh kode berikut menunjukkan cara menggunakan `delete-public-key`.

### AWS CLI

Untuk menghapus kunci CloudFront publik

Contoh berikut menghapus kunci CloudFront publik dengan ID `DKDFB19YGCR002`. Untuk menghapus kunci publik, Anda harus memiliki ID dan ETag. ID dikembalikan dalam output `list-public-keys` perintah `create-public-key` dan. Untuk mendapatkan ETag, gunakan `get-public-key-config` perintah `get-public-key` or. Gunakan `--if-match` opsi untuk menyediakan kunci publik ETag.

```
aws cloudfront delete-public-key \  
  --id KDFB19YGCR002 \  
  --if-match E2QWRUHEXAMPLE
```

Ketika berhasil, perintah ini tidak memiliki output.

- Untuk detail API, lihat [DeletePublicKey](#) di Referensi AWS CLI Perintah.

## get-cloud-front-origin-access-identity-config

Contoh kode berikut menunjukkan cara menggunakan `get-cloud-front-origin-access-identity-config`.

### AWS CLI

Untuk mendapatkan konfigurasi identitas akses CloudFront asal

Contoh berikut mendapatkan metadata tentang identitas akses CloudFront asal (OAI) dengan IDE74FTE3AEXAMPLE, termasuk nya. ETag ID OAI dikembalikan dalam output perintah `-access-identity` dan `create-cloud-front-origin -access-identities`. `list-cloud-front-origin`

```
aws cloudfront get-cloud-front-origin-access-identity-config --id E74FTE3AEXAMPLE
```

Output:

```
{  
  "ETag": "E2QWRUHEXAMPLE",  
  "CloudFrontOriginAccessIdentityConfig": {  
    "CallerReference": "cli-example",  
    "Comment": "Example OAI"  
  }  
}
```

- Untuk detail API, lihat [GetCloudFrontOriginAccessIdentityConfig](#) di Referensi AWS CLI Perintah.

## get-cloud-front-origin-access-identity

Contoh kode berikut menunjukkan cara menggunakan `get-cloud-front-origin-access-identity`.

## AWS CLI

Untuk mendapatkan identitas akses CloudFront asal

Contoh berikut mendapatkan identitas akses CloudFront asal (OAI) dengan IDE74FTE3AEXAMPLE, termasuk ID kanonik ETag dan S3 terkait. ID OAI dikembalikan dalam output perintah `-access-identity` dan `create-cloud-front-origin -access-identities`. `list-cloud-front-origin`

```
aws cloudfront get-cloud-front-origin-access-identity --id E74FTE3AEXAMPLE
```

Output:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "CloudFrontOriginAccessIdentity": {
    "Id": "E74FTE3AEXAMPLE",
    "S3CanonicalUserId":
    "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example OAI"
    }
  }
}
```

- Untuk detail API, lihat [GetCloudFrontOriginAccessIdentity](#) di Referensi AWS CLI Perintah.

## get-distribution-config

Contoh kode berikut menunjukkan cara menggunakan `get-distribution-config`.

## AWS CLI

Untuk mendapatkan konfigurasi CloudFront distribusi

Contoh berikut mendapatkan metadata tentang CloudFront distribusi dengan IDEDFDVBBD6EXAMPLE, termasuk nya. ETag ID distribusi dikembalikan dalam perintah `create-distribution` dan `list-distributions`.

```
aws cloudfront get-distribution-config --id EDFDVBBD6EXAMPLE
```

## Output:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
          "DomainName": "awsexamplebucket.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          },
          "S3OriginConfig": {
            "OriginAccessIdentity": ""
          }
        }
      ]
    },
    "OriginGroups": {
      "Quantity": 0
    },
    "DefaultCacheBehavior": {
      "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
      "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
          "Forward": "none"
        }
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
```

```
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
```

```

        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
}

```

- Untuk detail API, lihat [GetDistributionConfig](#) di Referensi AWS CLI Perintah.

## get-distribution

Contoh kode berikut menunjukkan cara menggunakan `get-distribution`.

### AWS CLI

Untuk mendapatkan CloudFront distribusi

Contoh berikut mendapatkan CloudFront distribusi dengan ID `EDFDVBD6EXAMPLE`, termasuk `Etag`. ID distribusi dikembalikan dalam perintah `create-distribution` dan `list-distributions`.

```
aws cloudfront get-distribution --id EDFDVBD6EXAMPLE
```

Output:

```

{
  "ETag": "E2QWRUHEXAMPLE",
  "Distribution": {
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
    "Status": "Deployed",
    "LastModifiedTime": "2019-12-04T23:35:41.433Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",

```

```
"ActiveTrustedSigners": {
  "Enabled": false,
  "Quantity": 0
},
"DistributionConfig": {
  "CallerReference": "cli-example",
  "Aliases": {
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
        "S3OriginConfig": {
          "OriginAccessIdentity": ""
        }
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
```

```
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
```



```

        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
}
}

```

- Untuk detail API, lihat [GetDistribution](#) di Referensi AWS CLI Perintah.

## get-field-level-encryption-config

Contoh kode berikut menunjukkan cara menggunakan `get-field-level-encryption-config`.

### AWS CLI

Untuk mendapatkan metadata tentang konfigurasi enkripsi tingkat CloudFront lapangan

Contoh berikut mendapatkan metadata tentang konfigurasi enkripsi CloudFront tingkat lapangan dengan ID, termasuk: C3KM2WVD605UAY ETag

```
aws cloudfront get-field-level-encryption-config --id C3KM2WVD605UAY
```

Output:

```

{
  "ETag": "E2P4Z4VU7TY5SG",
  "FieldLevelEncryptionConfig": {
    "CallerReference": "cli-example",
    "Comment": "Example FLE configuration",
    "QueryArgProfileConfig": {
      "ForwardWhenQueryArgProfileIsUnknown": true,
      "QueryArgProfiles": {

```

```

        "Quantity": 0,
        "Items": []
      }
    },
    "ContentTypeProfileConfig": {
      "ForwardWhenContentTypeIsUnknown": true,
      "ContentTypeProfiles": {
        "Quantity": 1,
        "Items": [
          {
            "Format": "URLEncoded",
            "ProfileId": "P280MFCLSY0CVU",
            "ContentType": "application/x-www-form-urlencoded"
          }
        ]
      }
    }
  }
}

```

- Untuk detail API, lihat [GetFieldLevelEncryptionConfig](#) di Referensi AWS CLI Perintah.

## get-field-level-encryption-profile-config

Contoh kode berikut menunjukkan cara menggunakan `get-field-level-encryption-profile-config`.

### AWS CLI

Untuk mendapatkan konfigurasi CloudFront profil enkripsi tingkat lapangan

Contoh berikut mendapatkan metadata tentang profil enkripsi CloudFront tingkat lapangan dengan ID, termasuk: PPK0U0SIF5WSV ETag

```
aws cloudfront get-field-level-encryption-profile-config --id PPK0U0SIF5WSV
```

### Output:

```
{
  "ETag": "E1QQG65FS2L2GC",
  "FieldLevelEncryptionProfileConfig": {

```

```

    "Name": "ExampleFLEProfile",
    "CallerReference": "cli-example",
    "Comment": "FLE profile for AWS CLI example",
    "EncryptionEntities": {
      "Quantity": 1,
      "Items": [
        {
          "PublicKeyId": "K2K8NC4HVFE3M0",
          "ProviderId": "ExampleFLEProvider",
          "FieldPatterns": {
            "Quantity": 1,
            "Items": [
              "ExampleSensitiveField"
            ]
          }
        }
      ]
    }
  ]
}

```

- Untuk detail API, lihat [GetFieldLevelEncryptionProfileConfig](#) di Referensi AWS CLI Perintah.

## get-field-level-encryption-profile

Contoh kode berikut menunjukkan cara menggunakan `get-field-level-encryption-profile`.

### AWS CLI

Untuk mendapatkan profil CloudFront enkripsi tingkat lapangan

Contoh berikut mendapatkan profil enkripsi CloudFront tingkat lapangan dengan ID `PPK0U0SIF5WSV`, termasuk: ETag

```
aws cloudfront get-field-level-encryption-profile --id PPK0U0SIF5WSV
```

### Output:

```

{
  "ETag": "E1QQG65FS2L2GC",
  "FieldLevelEncryptionProfile": {
    "Id": "PPK0U0SIF5WSV",

```

```

    "LastModifiedTime": "2019-12-10T01:03:16.537Z",
    "FieldLevelEncryptionProfileConfig": {
      "Name": "ExampleFLEProfile",
      "CallerReference": "cli-example",
      "Comment": "FLE profile for AWS CLI example",
      "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
          {
            "PublicKeyId": "K2K8NC4HVFE3M0",
            "ProviderId": "ExampleFLEProvider",
            "FieldPatterns": {
              "Quantity": 1,
              "Items": [
                "ExampleSensitiveField"
              ]
            }
          }
        ]
      }
    }
  ]
}

```

- Untuk detail API, lihat [GetFieldLevelEncryptionProfile](#) di Referensi AWS CLI Perintah.

## get-field-level-encryption

Contoh kode berikut menunjukkan cara menggunakan `get-field-level-encryption`.

### AWS CLI

Untuk mendapatkan konfigurasi CloudFront enkripsi tingkat lapangan

Contoh berikut mendapatkan konfigurasi enkripsi CloudFront tingkat lapangan dengan IDC3KM2WVD605UAY, termasuk: ETag

```
aws cloudfront get-field-level-encryption --id C3KM2WVD605UAY
```

Output:

```
{
```

```

    "ETag": "E2P4Z4VU7TY5SG",
    "FieldLevelEncryption": {
      "Id": "C3KM2WVD605UAY",
      "LastModifiedTime": "2019-12-10T21:30:18.974Z",
      "FieldLevelEncryptionConfig": {
        "CallerReference": "cli-example",
        "Comment": "Example FLE configuration",
        "QueryArgProfileConfig": {
          "ForwardWhenQueryArgProfileIsUnknown": true,
          "QueryArgProfiles": {
            "Quantity": 0,
            "Items": []
          }
        },
        "ContentTypeProfileConfig": {
          "ForwardWhenContentTypeIsUnknown": true,
          "ContentTypeProfiles": {
            "Quantity": 1,
            "Items": [
              {
                "Format": "URLEncoded",
                "ProfileId": "P280MFCLSY0CVU",
                "ContentType": "application/x-www-form-urlencoded"
              }
            ]
          }
        }
      }
    }
  }
}

```

- Untuk detail API, lihat [GetFieldLevelEncryption](#) di Referensi AWS CLI Perintah.

## get-invalidations

Contoh kode berikut menunjukkan cara menggunakan `get-invalidations`.

### AWS CLI

Untuk mendapatkan CloudFront pembatalan

Contoh berikut mendapatkan pembatalan dengan ID I2J0I21PCUY0IK untuk CloudFront distribusi dengan ID: EDFDVBD6EXAMPLE

```
aws cloudfront get-invalidation --id I2J0I21PCUY0IK --distribution-id
EDFDVBD6EXAMPLE
```

Output:

```
{
  "Invalidation": {
    "Status": "Completed",
    "InvalidationBatch": {
      "Paths": {
        "Items": [
          "/example-path/example-file.jpg",
          "/example-path/example-file-2.jpg"
        ],
        "Quantity": 2
      },
      "CallerReference": "cli-example"
    },
    "Id": "I2J0I21PCUY0IK",
    "CreateTime": "2019-12-05T18:40:49.413Z"
  }
}
```

- Untuk detail API, lihat [GetInvalidation](#) di Referensi AWS CLI Perintah.

## get-public-key-config

Contoh kode berikut menunjukkan cara menggunakan `get-public-key-config`.

AWS CLI

Untuk mendapatkan konfigurasi kunci CloudFront publik

Contoh berikut mendapatkan metadata tentang kunci CloudFront publik dengan IDKDFB19YGCR002, termasuk nya. ETag ID kunci publik dikembalikan dalam `list-public-keys` perintah `create-public-key` and.

```
aws cloudfront get-public-key-config --id KDFB19YGCR002
```

Output:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKeyConfig": {
    "CallerReference": "cli-example",
    "Name": "ExampleKey",
    "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAA0CAQ8AMIIBCgKCAQEAxPmbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2Nq0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
    "Comment": "example public key"
  }
}
```

- Untuk detail API, lihat [GetPublicKeyConfig](#) di Referensi AWS CLI Perintah.

## get-public-key

Contoh kode berikut menunjukkan cara menggunakan `get-public-key`.

### AWS CLI

Untuk mendapatkan kunci CloudFront publik

Contoh berikut mendapatkan kunci CloudFront publik dengan ID `KDFB19YGCR002`, termasuk nya `ETag`. ID kunci publik dikembalikan dalam `list-public-keys` perintah `create-public-key` and.

```
aws cloudfront get-public-key --id KDFB19YGCR002
```

Output:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKey": {
    "Id": "KDFB19YGCR002",
    "CreatedTime": "2019-12-05T18:51:43.781Z",
    "PublicKeyConfig": {
      "CallerReference": "cli-example",
```

```

        "Name": "ExampleKey",
        "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIBCgKCAQEAxPmBCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
        "Comment": "example public key"
    }
}
}

```

- Untuk detail API, lihat [GetPublicKey](#) di Referensi AWS CLI Perintah.

## list-cloud-front-origin-access-identities

Contoh kode berikut menunjukkan cara menggunakan `list-cloud-front-origin-access-identities`.

### AWS CLI

Untuk mencantumkan identitas akses CloudFront asal

Contoh berikut mendapatkan daftar identitas akses CloudFront asal (OAI) di akun Anda AWS :

```
aws cloudfront list-cloud-front-origin-access-identities
```

Output:

```

{
  "CloudFrontOriginAccessIdentityList": {
    "Items": [
      {
        "Id": "E74FTE3AEXAMPLE",
        "S3CanonicalUserId":
"cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "Comment": "Example OAI"
      },
      {
        "Id": "EH1HDMBEXAMPLE",

```



```

        "S3CanonicalUserId":
"1489f6f2e6faacaae7ff64c4c3e6956c24f78788abfc1718c3527c263bf7a17EXAMPLE",
        "Comment": "Test OAI"
    },
    {
        "Id": "E2X2C9TEXAMPLE",
        "S3CanonicalUserId":
"cbfeebb915a64749f9be546a45b3fcfd3a31c779673c13c4dd460911ae402c2EXAMPLE",
        "Comment": "Example OAI #2"
    }
]
}
}

```

- Untuk detail API, lihat [ListCloudFrontOriginAccessIdentities](#) di Referensi AWS CLI Perintah.

## list-distributions

Contoh kode berikut menunjukkan cara menggunakan `list-distributions`.

### AWS CLI

Untuk daftar CloudFront distribusi

Contoh berikut mendapatkan daftar CloudFront distribusi di AWS akun Anda:

```
aws cloudfront list-distributions
```

Output:

```

{
  "DistributionList": {
    "Items": [
      {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/
EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-11-22T00:55:15.705Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
          "Enabled": false,

```

```

        "Quantity": 0
    },
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
            "Items": [
                {
                    "Id": "awsexamplebucket.s3.amazonaws.com-cli-
example",
                    "DomainName": "awsexamplebucket.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    },
                    "S3OriginConfig": {
                        "OriginAccessIdentity": ""
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-
example",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {

```

```
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponse": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
```

```

        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
},
{
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-12-04T23:35:41.433Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d930174dauwrn8.cloudfront.net",
    "ActiveTrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
            "Items": [
                {
                    "Id": "awsexamplebucket1.s3.amazonaws.com-cli-example",
                    "DomainName": "awsexamplebucket1.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    }
                }
            ]
        }
    }
}

```

```
        "S3OriginConfig": {
            "OriginAccessIdentity": ""
        }
    ],
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket1.s3.amazonaws.com-cli-
example",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 0
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    }
},
```

```
        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
},
{
    "Id": "E1X5IZQEXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/
E1X5IZQEXAMPLE",
```

```
"Status": "Deployed",
"LastModifiedTime": "2019-11-06T21:31:48.864Z",
"DomainName": "d2e04y12345678.cloudfront.net",
"Aliases": {
  "Quantity": 0
},
"Origins": {
  "Quantity": 1,
  "Items": [
    {
      "Id": "awsexamplebucket2",
      "DomainName": "awsexamplebucket2.s3.us-
west-2.amazonaws.com",
      "OriginPath": "",
      "CustomHeaders": {
        "Quantity": 0
      },
      "S3OriginConfig": {
        "OriginAccessIdentity": ""
      }
    }
  ]
},
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "awsexamplebucket2",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    }
  },
  "Headers": {
    "Quantity": 0
  },
  "QueryStringCacheKeys": {
    "Quantity": 0
  }
},
"TrustedSigners": {
  "Enabled": false,
  "Quantity": 0
},
}
```

```
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponse": {
    "Quantity": 0
  },
  "Comment": "",
  "PriceClass": "PriceClass_All",
  "Enabled": true,
  "ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
  },
  "Restrictions": {
    "GeoRestriction": {
      "RestrictionType": "none",
      "Quantity": 0
    }
  }
}
```



```

    },
    "WebACLId": "",
    "HttpVersion": "HTTP1_1",
    "IsIPV6Enabled": true
  }
]
}
}

```

- Untuk detail API, lihat [ListDistributions](#) di Referensi AWS CLI Perintah.

## list-field-level-encryption-configs

Contoh kode berikut menunjukkan cara menggunakan `list-field-level-encryption-configs`.

### AWS CLI

Untuk membuat daftar konfigurasi CloudFront enkripsi tingkat lapangan

Contoh berikut mendapatkan daftar konfigurasi enkripsi CloudFront tingkat bidang di akun Anda:  
AWS

```
aws cloudfront list-field-level-encryption-configs
```

Output:

```

{
  "FieldLevelEncryptionList": {
    "MaxItems": 100,
    "Quantity": 1,
    "Items": [
      {
        "Id": "C3KM2WVD605UAY",
        "LastModifiedTime": "2019-12-10T21:30:18.974Z",
        "Comment": "Example FLE configuration",
        "QueryArgProfileConfig": {
          "ForwardWhenQueryArgProfileIsUnknown": true,
          "QueryArgProfiles": {
            "Quantity": 0,
            "Items": []
          }
        }
      }
    ]
  }
}

```

```

    },
    "ContentTypeProfileConfig": {
      "ForwardWhenContentTypeIsUnknown": true,
      "ContentTypeProfiles": {
        "Quantity": 1,
        "Items": [
          {
            "Format": "URLEncoded",
            "ProfileId": "P280MFCLSYOCVU",
            "ContentType": "application/x-www-form-urlencoded"
          }
        ]
      }
    }
  ]
}

```

- Untuk detail API, lihat [ListFieldLevelEncryptionConfigs](#) di Referensi AWS CLI Perintah.

## list-field-level-encryption-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-field-level-encryption-profiles`.

### AWS CLI

Untuk membuat daftar profil CloudFront enkripsi tingkat lapangan

Contoh berikut mendapatkan daftar profil enkripsi CloudFront tingkat bidang di akun Anda: AWS

```
aws cloudfront list-field-level-encryption-profiles
```

Output:

```

{
  "FieldLevelEncryptionProfileList": {
    "MaxItems": 100,
    "Quantity": 2,
    "Items": [
      {

```

```
    "Id": "P280MFCLSY0CVU",
    "LastModifiedTime": "2019-12-05T01:05:39.896Z",
    "Name": "ExampleFLEProfile",
    "EncryptionEntities": {
      "Quantity": 1,
      "Items": [
        {
          "PublicKeyId": "K2K8NC4HVFE3M0",
          "ProviderId": "ExampleFLEProvider",
          "FieldPatterns": {
            "Quantity": 1,
            "Items": [
              "ExampleSensitiveField"
            ]
          }
        }
      ]
    },
    "Comment": "FLE profile for AWS CLI example"
  },
  {
    "Id": "PPK0U0SIF5WSV",
    "LastModifiedTime": "2019-12-10T01:03:16.537Z",
    "Name": "ExampleFLEProfile2",
    "EncryptionEntities": {
      "Quantity": 1,
      "Items": [
        {
          "PublicKeyId": "K2ABC10EXAMPLE",
          "ProviderId": "ExampleFLEProvider2",
          "FieldPatterns": {
            "Quantity": 1,
            "Items": [
              "ExampleSensitiveField2"
            ]
          }
        }
      ]
    },
    "Comment": "FLE profile #2 for AWS CLI example"
  }
]
```

```
}
```

- Untuk detail API, lihat [ListFieldLevelEncryptionProfiles](#) di Referensi AWS CLI Perintah.

## list-invalidations

Contoh kode berikut menunjukkan cara menggunakan `list-invalidations`.

### AWS CLI

Untuk daftar CloudFront pembatalan

Contoh berikut mendapatkan daftar pembatalan untuk CloudFront distribusi dengan ID: EDFDVBD6EXAMPLE

```
aws cloudfront list-invalidations --distribution-id EDFDVBD6EXAMPLE
```

Output:

```
{
  "InvalidationList": {
    "Marker": "",
    "Items": [
      {
        "Status": "Completed",
        "Id": "YNY2LI2BVJ4NJU",
        "CreateTime": "2019-08-31T21:15:52.042Z"
      }
    ],
    "IsTruncated": false,
    "MaxItems": 100,
    "Quantity": 1
  }
}
```

- Untuk detail API, lihat [ListInvalidations](#) di Referensi AWS CLI Perintah.

## list-public-keys

Contoh kode berikut menunjukkan cara menggunakan `list-public-keys`.

## AWS CLI

Untuk daftar kunci CloudFront publik

Contoh berikut mendapatkan daftar kunci CloudFront publik di AWS akun Anda:

```
aws cloudfront list-public-keys
```

Output:

```
{
  "PublicKeyList": {
    "MaxItems": 100,
    "Quantity": 2,
    "Items": [
      {
        "Id": "K2K8NC4HVFE3M0",
        "Name": "ExampleKey",
        "CreatedTime": "2019-12-05T01:04:28.818Z",
        "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAA0CAQ8AMIIBCgKCAQEApMbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjm3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
        "Comment": "example public key"
      },
      {
        "Id": "K1S0LWQ2L5HTBU",
        "Name": "ExampleKey2",
        "CreatedTime": "2019-12-09T23:28:11.110Z",
        "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAA0CAQ8AMIIBCgKCAQEAp0CAg88A8+f4dujn9Izt
\n26LxtgAkn2opGgo/NKpMiaisyw5qlg3f1gol7FV6pYN178iJg3E08JBbwt1H
+cR9\nLGSf60NDeVhm760c39Np/vWg0dsGQcRbi9WmKZeS0DqjQGzVZWqPmito3FzWV6k6b
\nfVY5N36U/RdbVAJm95Km+qaMY1bIdF40t72bi3IkKYV5h1B2XoDj1Q9F6ajQKyTB
\nMHa3SN8q+3ZjQ4sJJ7D1V6r4wR8jDcFVD5NckWJmmgIVnk0QM37NYeoDnka0uTpu\nnha/
+3b8t0b2z3LBVHPkp85zJRA0XacSwf5rZtPYKBNFsixTa2n55k2r218m0kMC4\nnUwIDAQAB\n-----END
PUBLIC KEY-----",
        "Comment": "example public key #2"
      }
    ]
  }
}
```

```
    ]
  }
}
```

- Untuk detail API, lihat [ListPublicKeys](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk CloudFront distribusi

Contoh berikut mendapat daftar tag untuk CloudFront distribusi:

```
aws cloudfront list-tags-for-resource \
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE
```

Output:

```
{
  "Tags": {
    "Items": [
      {
        "Key": "DateCreated",
        "Value": "2019-12-04"
      },
      {
        "Key": "Name",
        "Value": "Example name"
      },
      {
        "Key": "Project",
        "Value": "Example project"
      }
    ]
  }
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## sign

Contoh kode berikut menunjukkan cara menggunakan `sign`.

### AWS CLI

Untuk menandatangani CloudFront URL

Contoh berikut menandatangani CloudFront URL. Untuk menandatangani URL, Anda memerlukan ID key pair (disebut ID Kunci Akses di Konsol AWS Manajemen) dan kunci pribadi dari CloudFront key pair penandatanganan tepercaya. Untuk informasi selengkapnya tentang URL yang ditandatangani, lihat [Menayangkan Konten Pribadi dengan URL yang Ditandatangani dan Cookie Bertanda Tangan](#) di Panduan CloudFront Pengembang Amazon.

```
aws cloudfront sign \  
  --url https://d111111abcdef8.cloudfront.net/private-content/private-file.html \  
  --key-pair-id APKAEIBAERJR2EXAMPLE \  
  --private-key file://cf-signer-priv-key.pem \  
  --date-less-than 2020-01-01
```

Output:

```
https://d111111abcdef8.cloudfront.net/private-content/private-  
file.html?Expires=1577836800&Signature=nEXK7Kby47XKeZQKvc6pwkif6oZc-  
JWSpDkH0UH7EBGGqvgurkecCbgL5VfUAXyLQuJxFwRQWscz-  
owcq9KpmewCXrXQbPaJZNi9XSNwf4YKurPDQYaRQawKoeenH0GFteRf9ELK-  
Bs3n1jTLjtbgzIUt7QJNKXcWr8AuUYikzGdJ4-qzx6WnxXfH~fxg4-  
GG1612kgCpXUB6Jx6K~Y3kpV0dzUP0IqFLHANJojbhxqrVejomZZ2XrquDvNUCCIbePGnR3d24UPaLXG4FK0qNEaWDIB-  
GNvjRJxqWf93uMobeM0iVYahb-e0KIItiQewGcm0eLZQ__&Key-Pair-Id=APKAEIBAERJR2EXAMPLE
```

- Untuk detail API, lihat Referensi AWS CLI Perintah [Masuk](#).

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai CloudFront distribusi

`tag-resource` Contoh berikut menambahkan dua tag ke CloudFront distribusi yang ditentukan.

```
aws cloudfront tag-resource \  
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \  
  --tags 'Items=[{Key=Name,Value="Example name"},{Key=Project,Value="Example  
project"}]'
```

Alih-alih menggunakan argumen baris perintah, Anda dapat memberikan tag dalam file JSON, seperti yang ditunjukkan pada contoh berikut:

```
aws cloudfront tag-resource \  
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \  
  --tags file://tags.json
```

Isi dari `tags.json`:

```
{  
  "Items": [  
    {  
      "Key": "Name",  
      "Value": "Example name"  
    },  
    {  
      "Key": "Project",  
      "Value": "Example project"  
    }  
  ]  
}
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari CloudFront distribusi

Contoh berikut menghapus dua tag dari CloudFront distribusi dengan menggunakan argumen baris perintah:



```
aws cloudfront untag-resource \
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \
  --tag-keys Items=Name,Project
```

Alih-alih menggunakan argumen baris perintah, Anda dapat memberikan kunci tag dalam file JSON, seperti yang ditunjukkan pada contoh berikut:

```
aws cloudfront untag-resource \
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \
  --tag-keys file://tag-keys.json
```

File tersebut `tag-keys.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini:

```
{
  "Items": [
    "Name",
    "Project"
  ]
}
```

Ketika berhasil, perintah ini tidak memiliki output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-cloud-front-origin-access-identity

Contoh kode berikut menunjukkan cara menggunakan `update-cloud-front-origin-access-identity`.

### AWS CLI

Untuk memperbarui identitas akses CloudFront asal

Contoh berikut memperbarui identitas akses asal (OAI) dengan `IDE74FTE3AEXAMPLE`. Satu-satunya bidang yang dapat Anda perbarui adalah OAI. Comment

Untuk memperbarui OAI, Anda harus memiliki ID OAI dan. ETag ID OAI dikembalikan dalam output perintah `-access-identity` dan `create-cloud-front-origin -access-identities`. `list-cloud-front-origin` Untuk mendapatkan ETag, gunakan perintah `get-cloud-front-origin -access-identity` atau `get-`

cloud-front-origin -. access-identity-config Gunakan --if-match opsi untuk menyediakan OAI. ETag

```
aws cloudfront update-cloud-front-origin-access-identity \  
  --id E74FTE3AEXAMPLE \  
  --if-match E2QWRUHEXAMPLE \  
  --cloud-front-origin-access-identity-config \  
    CallerReference=cli-example,Comment="Example OAI Updated"
```

Anda dapat mencapai hal yang sama dengan menyediakan konfigurasi OAI dalam file JSON, seperti yang ditunjukkan pada contoh berikut:

```
aws cloudfront update-cloud-front-origin-access-identity \  
  --id E74FTE3AEXAMPLE \  
  --if-match E2QWRUHEXAMPLE \  
  --cloud-front-origin-access-identity-config file://OAI-config.json
```

File tersebut OAI-config.json adalah dokumen JSON di direktori saat ini yang berisi berikut ini:

```
{  
  "CallerReference": "cli-example",  
  "Comment": "Example OAI Updated"  
}
```

Apakah Anda memberikan konfigurasi OAI dengan argumen baris perintah atau file JSON, outputnya sama:

```
{  
  "ETag": "E9LHASXEXAMPLE",  
  "CloudFrontOriginAccessIdentity": {  
    "Id": "E74FTE3AEXAMPLE",  
    "S3CanonicalUserId":  
      "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",  
    "CloudFrontOriginAccessIdentityConfig": {  
      "CallerReference": "cli-example",  
      "Comment": "Example OAI Updated"  
    }  
  }  
}
```

- Untuk detail API, lihat [UpdateCloudFrontOriginAccessIdentity](#) di Referensi AWS CLI Perintah.

## update-distribution

Contoh kode berikut menunjukkan cara menggunakan `update-distribution`.

### AWS CLI

Untuk memperbarui objek root default CloudFront distribusi

Contoh berikut memperbarui objek root default `index.html` untuk CloudFront distribusi dengan `IDEDFDVBD6EXAMPLE`:

```
aws cloudfront update-distribution --id EDFDVBD6EXAMPLE \  
  --default-root-object index.html
```

Output:

```
{  
  "ETag": "E2QWRUHEXAMPLE",  
  "Distribution": {  
    "Id": "EDFDVBD6EXAMPLE",  
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",  
    "Status": "InProgress",  
    "LastModifiedTime": "2019-12-06T18:55:39.870Z",  
    "InProgressInvalidationBatches": 0,  
    "DomainName": "d1111111abcdef8.cloudfront.net",  
    "ActiveTrustedSigners": {  
      "Enabled": false,  
      "Quantity": 0  
    },  
    "DistributionConfig": {  
      "CallerReference": "6b10378d-49be-4c4b-a642-419ccaf8f3b5",  
      "Aliases": {  
        "Quantity": 0  
      },  
      "DefaultRootObject": "index.html",  
      "Origins": {  
        "Quantity": 1,  
        "Items": [  
          {  
            "Id": "example-website",  
            "DomainName": "www.example.com",
```

```
        "OriginPath": "",
        "CustomHeaders": {
            "Quantity": 0
        },
        "CustomOriginConfig": {
            "HTTPPort": 80,
            "HTTPSPort": 443,
            "OriginProtocolPolicy": "match-viewer",
            "OriginSslProtocols": {
                "Quantity": 2,
                "Items": [
                    "SSLv3",
                    "TLSv1"
                ]
            },
            "OriginReadTimeout": 30,
            "OriginKeepaliveTimeout": 5
        }
    ]
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "example-website",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 1,
            "Items": [
                "*"
            ]
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    }
}
```

```
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
    },
    "CachedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ]
    }
  },
  "SmoothStreaming": false,
  "DefaultTTL": 86400,
  "MaxTTL": 31536000,
  "Compress": false,
  "LambdaFunctionAssociations": {
    "Quantity": 0
  },
  "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
  "Quantity": 0
},
"CustomErrorResponses": {
  "Quantity": 0
},
"Comment": "",
"Logging": {
  "Enabled": false,
  "IncludeCookies": false,
  "Bucket": "",
  "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
  "CloudFrontDefaultCertificate": true,
  "MinimumProtocolVersion": "TLSv1",
```

```

        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http1.1",
    "IsIPV6Enabled": true
}
}
}

```

Untuk memperbarui CloudFront distribusi

Contoh berikut menonaktifkan CloudFront distribusi dengan ID EMLARXS9EXAMPLE dengan menyediakan konfigurasi distribusi dalam file JSON bernama `dist-config-disable.json`. Untuk memperbarui distribusi, Anda harus menggunakan `--if-match` opsi untuk menyediakan distribusiETag. Untuk mendapatkanETag, gunakan `get-distribusi` atau `get-distribution-config` perintah.

Setelah Anda menggunakan contoh berikut untuk menonaktifkan distribusi, Anda dapat menggunakan perintah `hapus-distribusi` untuk menghapusnya.

```

aws cloudfront update-distribution \
  --id EMLARXS9EXAMPLE \
  --if-match E2QWRUHEXAMPLE \
  --distribution-config file:///dist-config-disable.json

```

File tersebut `dist-config-disable.json` adalah dokumen JSON di folder saat ini yang berisi berikut ini. Perhatikan bahwa `Enabled` bidang diatur ke `false`:

```

{
  "CallerReference": "cli-1574382155-496510",
  "Aliases": {
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,

```

```
"Items": [
  {
    "Id": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
    "DomainName": "awsexamplebucket.s3.amazonaws.com",
    "OriginPath": "",
    "CustomHeaders": {
      "Quantity": 0
    },
    "S3OriginConfig": {
      "OriginAccessIdentity": ""
    }
  }
],
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    },
    "Headers": {
      "Quantity": 0
    },
    "QueryStringCacheKeys": {
      "Quantity": 0
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ],
    "CachedMethods": {
```

```
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ]
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": false,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
    }
},
"WebACLId": "",
"HttpVersion": "http2",
```



```
"IsIPV6Enabled": true
}
```

## Output:

```
{
  "ETag": "E9LHASXEXAMPLE",
  "Distribution": {
    "Id": "EMLARXS9EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-12-06T18:32:35.553Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
  },
  "DistributionConfig": {
    "CallerReference": "cli-1574382155-496510",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
          "DomainName": "awsexamplebucket.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          },
          "S3OriginConfig": {
            "OriginAccessIdentity": ""
          }
        }
      ]
    },
    "OriginGroups": {
      "Quantity": 0
    },
  },
}
```

```
    "DefaultCacheBehavior": {
      "TargetOriginId":
"awsexamplebucket.s3.amazonaws.com-1574382155-273939",
      "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
          "Forward": "none"
        },
        "Headers": {
          "Quantity": 0
        },
        "QueryStringCacheKeys": {
          "Quantity": 0
        }
      },
      "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
      },
      "ViewerProtocolPolicy": "allow-all",
      "MinTTL": 0,
      "AllowedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ],
        "CachedMethods": {
          "Quantity": 2,
          "Items": [
            "HEAD",
            "GET"
          ]
        }
      },
      "SmoothStreaming": false,
      "DefaultTTL": 86400,
      "MaxTTL": 31536000,
      "Compress": false,
      "LambdaFunctionAssociations": {
        "Quantity": 0
      },
      "FieldLevelEncryptionId": ""
    },
  },
```

```

    "CacheBehaviors": {
      "Quantity": 0
    },
    "CustomErrorResponses": {
      "Quantity": 0
    },
    "Comment": "",
    "Logging": {
      "Enabled": false,
      "IncludeCookies": false,
      "Bucket": "",
      "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": false,
    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}

```

- Untuk detail API, lihat [UpdateDistribution](#) di Referensi AWS CLI Perintah.

## update-field-level-encryption-config

Contoh kode berikut menunjukkan cara menggunakan `update-field-level-encryption-config`.

### AWS CLI

Untuk memperbarui konfigurasi CloudFront enkripsi tingkat lapangan

Contoh berikut memperbarui Comment bidang konfigurasi enkripsi tingkat lapangan dengan ID C3KM2WVD605UAY dengan menyediakan parameter dalam file JSON.

Untuk memperbarui konfigurasi enkripsi tingkat lapangan, Anda harus memiliki ID konfigurasi dan ETag ID dikembalikan dalam output perintah `create-field-level-encryption -config` dan `list-field-level-encryption -configs`. Untuk mendapatkan ETag, gunakan perintah `get-field-level-encryption` or `get-field-level-encryption -config`. Gunakan `--if-match` opsi untuk menyediakan konfigurasi ETag.

```
aws cloudfront update-field-level-encryption-config \  
  --id C3KM2WVD605UAY \  
  --if-match E2P4Z4VU7TY5SG \  
  --field-level-encryption-config file://fle-config.json
```

File tersebut `fle-config.json` adalah dokumen JSON di direktori saat ini yang berisi berikut ini:

```
{  
  "CallerReference": "cli-example",  
  "Comment": "Updated example FLE configuration",  
  "QueryArgProfileConfig": {  
    "ForwardWhenQueryArgProfileIsUnknown": true,  
    "QueryArgProfiles": {  
      "Quantity": 0  
    }  
  },  
  "ContentTypeProfileConfig": {  
    "ForwardWhenContentTypeIsUnknown": true,  
    "ContentTypeProfiles": {  
      "Quantity": 1,  
      "Items": [  
        {  
          "Format": "URLEncoded",  
          "ProfileId": "P280MFCLSY0CVU",  
          "ContentType": "application/x-www-form-urlencoded"  
        }  
      ]  
    }  
  }  
}
```

Output:

```

{
  "ETag": "E26M4BIAV81ZF6",
  "FieldLevelEncryption": {
    "Id": "C3KM2WVD605UAY",
    "LastModifiedTime": "2019-12-10T22:26:26.170Z",
    "FieldLevelEncryptionConfig": {
      "CallerReference": "cli-example",
      "Comment": "Updated example FLE configuration",
      "QueryArgProfileConfig": {
        "ForwardWhenQueryArgProfileIsUnknown": true,
        "QueryArgProfiles": {
          "Quantity": 0,
          "Items": []
        }
      },
      "ContentTypeProfileConfig": {
        "ForwardWhenContentTypeIsUnknown": true,
        "ContentTypeProfiles": {
          "Quantity": 1,
          "Items": [
            {
              "Format": "URLEncoded",
              "ProfileId": "P280MFCLSY0CVU",
              "ContentType": "application/x-www-form-urlencoded"
            }
          ]
        }
      }
    }
  }
}

```

- Untuk detail API, lihat [UpdateFieldLevelEncryptionConfig](#) di Referensi AWS CLI Perintah.

## update-field-level-encryption-profile

Contoh kode berikut menunjukkan cara menggunakan `update-field-level-encryption-profile`.

### AWS CLI

Untuk memperbarui profil CloudFront enkripsi tingkat lapangan

Contoh berikut memperbarui profil enkripsi tingkat lapangan dengan ID. PPK0U0SIF5WSV Contoh ini memperbarui profil Name dan Comment, dan menambahkan FieldPatterns item kedua, dengan menyediakan parameter dalam file JSON.

Untuk memperbarui profil enkripsi tingkat lapangan, Anda harus memiliki ID profil dan. ETag ID dikembalikan dalam output perintah `create-field-level-encryption -profile` dan `list-field-level-encryption -profiles`. Untuk mendapatkan ETag, gunakan perintah `get-field-level-encryption -profile` atau `get-field-level-encryption -profile-config`. Gunakan `--if-match` opsi untuk menyediakan profil ETag.

```
aws cloudfront update-field-level-encryption-profile \  
  --id PPK0U0SIF5WSV \  
  --if-match E1QQG65FS2L2GC \  
  --field-level-encryption-profile-config file://fle-profile-config.json
```

File tersebut `fle-profile-config.json` adalah dokumen JSON di direktori saat ini yang berisi berikut ini:

```
{  
  "Name": "ExampleFLEProfileUpdated",  
  "CallerReference": "cli-example",  
  "Comment": "Updated FLE profile for AWS CLI example",  
  "EncryptionEntities": {  
    "Quantity": 1,  
    "Items": [  
      {  
        "PublicKeyId": "K2K8NC4HVFE3M0",  
        "ProviderId": "ExampleFLEProvider",  
        "FieldPatterns": {  
          "Quantity": 2,  
          "Items": [  
            "ExampleSensitiveField",  
            "SecondExampleSensitiveField"  
          ]  
        }  
      }  
    ]  
  }  
}
```

Output:

```

{
  "ETag": "EJETYFJ9CL66D",
  "FieldLevelEncryptionProfile": {
    "Id": "PPK0UOSIF5WSV",
    "LastModifiedTime": "2019-12-10T19:05:58.296Z",
    "FieldLevelEncryptionProfileConfig": {
      "Name": "ExampleFLEProfileUpdated",
      "CallerReference": "cli-example",
      "Comment": "Updated FLE profile for AWS CLI example",
      "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
          {
            "PublicKeyId": "K2K8NC4HVFE3M0",
            "ProviderId": "ExampleFLEProvider",
            "FieldPatterns": {
              "Quantity": 2,
              "Items": [
                "ExampleSensitiveField",
                "SecondExampleSensitiveField"
              ]
            }
          ]
        ]
      }
    ]
  }
}

```

- Untuk detail API, lihat [UpdateFieldLevelEncryptionProfile](#) di Referensi AWS CLI Perintah.

## CloudSearch Contoh Amazon menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum AWS Command Line Interface dengan menggunakan Amazon CloudSearch.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **upload-documents**

Contoh kode berikut menunjukkan cara menggunakan `upload-documents`.

AWS CLI

`upload-documents` Perintah berikut mengunggah sekumpulan dokumen JSON ke domain Amazon CloudSearch :

```
aws cloudsearchdomain upload-documents --endpoint-url https://doc-my-domain.us-west-1.cloudsearch.amazonaws.com --content-type application/json --documents document-batch.json
```

Output:

```
{
  "status": "success",
  "adds": 5000,
  "deletes": 0
}
```

- Untuk detail API, lihat [UploadDocuments](#) di Referensi AWS CLI Perintah.

## CloudTrail contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CloudTrail.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.



Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags**

Contoh kode berikut menunjukkan cara menggunakan `add-tags`.

AWS CLI

Untuk menambahkan tag ke jejak

`add-tags` Perintah berikut menambahkan tag untuk `Trail1`:

```
aws cloudtrail add-tags --resource-id arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 --tags-list Key=name,Value=Alice
Key=location,Value=us
```

- Untuk detail API, lihat [AddTags](#) di Referensi AWS CLI Perintah.

### **create-subscription**

Contoh kode berikut menunjukkan cara menggunakan `create-subscription`.

AWS CLI

Untuk membuat dan mengonfigurasi AWS sumber daya untuk jejak

`create-subscription` Perintah berikut membuat bucket S3 baru dan topik SNS untuk: `Trail1`

```
aws cloudtrail create-subscription --name Trail1 --s3-new-bucket my-bucket --sns-new-topic my-topic
```

Output:

```
Setting up new S3 bucket my-bucket...
Setting up new SNS topic my-topic...
Creating/updating CloudTrail configuration...
CloudTrail configuration:
{
  "trailList": [
    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail1",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket",
      "SnsTopicName": "my-topic",
      "HomeRegion": "us-east-1"
    }
  ],
  "ResponseMetadata": {
    "HTTPStatusCode": 200,
    "RequestId": "f39e51f6-c615-11e5-85bd-d35ca21ee3e2"
  }
}
Starting CloudTrail service...
Logs will be delivered to my-bucket
```

- Untuk detail API, lihat [CreateSubscription](#) di Referensi AWS CLI Perintah.

## create-trail

Contoh kode berikut menunjukkan cara menggunakan `create-trail`.

### AWS CLI

Untuk membuat jejak

`create-trail` Perintah berikut membuat jejak multi-wilayah bernama `Trail1` dan menentukan bucket S3:

```
aws cloudtrail create-trail --name Trail1 --s3-bucket-name my-bucket --is-multi-region-trail
```

Output:

```
{
  "IncludeGlobalServiceEvents": true,
  "Name": "Trail1",
  "TrailARN": "arn:aws:cloudtrail:us-west-2:123456789012:trail/Trail1",
  "LogFileValidationEnabled": false,
  "IsMultiRegionTrail": true,
  "S3BucketName": "my-bucket"
}
```

- Untuk detail API, lihat [CreateTrail](#) di Referensi AWS CLI Perintah.

## delete-trail

Contoh kode berikut menunjukkan cara menggunakan `delete-trail`.

### AWS CLI

Untuk menghapus jejak

`delete-trail` Perintah berikut menghapus jejak bernama `Trail1`:

```
aws cloudtrail delete-trail --name Trail1
```

- Untuk detail API, lihat [DeleteTrail](#) di Referensi AWS CLI Perintah.

## describe-trails

Contoh kode berikut menunjukkan cara menggunakan `describe-trails`.

### AWS CLI

Untuk menggambarkan jejak

`describe-trails` Perintah berikut mengembalikan pengaturan untuk `Trail1` dan `Trail2`:

```
aws cloudtrail describe-trails --trail-name-list Trail1 Trail2
```

Output:

```
{
  "trailList": [
```

```

    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail1",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket",
      "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/
CloudTrail_CloudWatchLogs_Role",
      "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-east-1:123456789012:log-
group:CloudTrail:*",
      "SnsTopicName": "my-topic",
      "HomeRegion": "us-east-1"
    },
    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail2",
      "S3KeyPrefix": "my-prefix",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket",
      "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/4c5ae5ac-3c13-421e-8335-c7868ef6a769",
      "HomeRegion": "us-east-1"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeTrails](#) di Referensi AWS CLI Perintah.

## get-event-selectors

Contoh kode berikut menunjukkan cara menggunakan `get-event-selectors`.

### AWS CLI

Untuk melihat pengaturan pemilih acara untuk jejak

`get-event-selectors` Perintah berikut mengembalikan pengaturan untuk `Trail1`:

```
aws cloudtrail get-event-selectors --trail-name Trail1
```

**Output:**

```
{
  "EventSelectors": [
    {
      "IncludeManagementEvents": true,
      "DataResources": [],
      "ReadWriteType": "All"
    }
  ],
  "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1"
}
```

- Untuk detail API, lihat [GetEventSelectors](#) di Referensi AWS CLI Perintah.

**get-trail-status**

Contoh kode berikut menunjukkan cara menggunakan `get-trail-status`.

**AWS CLI**

Untuk mendapatkan status jejak

`get-trail-status` Perintah berikut mengembalikan detail pengiriman dan pencatatan untuk `Trail1`:

```
aws cloudtrail get-trail-status --name Trail1
```

**Output:**

```
{
  "LatestNotificationTime": 1454022144.869,
  "LatestNotificationAttemptSucceeded": "2016-01-28T23:02:24Z",
  "LatestDeliveryAttemptTime": "2016-01-28T23:02:24Z",
  "LatestDeliveryTime": 1454022144.869,
  "TimeLoggingStarted": "2015-11-06T18:36:38Z",
  "LatestDeliveryAttemptSucceeded": "2016-01-28T23:02:24Z",
  "IsLogging": true,
  "LatestCloudWatchLogsDeliveryTime": 1454022144.918,
  "StartLoggingTime": 1446834998.695,
  "StopLoggingTime": 1446834996.933,
}
```

```

    "LatestNotificationAttemptTime": "2016-01-28T23:02:24Z",
    "TimeLoggingStopped": "2015-11-06T18:36:36Z"
}

```

- Untuk detail API, lihat [GetTrailStatus](#) di Referensi AWS CLI Perintah.

## list-public-keys

Contoh kode berikut menunjukkan cara menggunakan `list-public-keys`.

### AWS CLI

Untuk membuat daftar semua kunci publik untuk jejak

`list-public-keys` Perintah berikut mengembalikan semua kunci publik yang kunci pribadinya digunakan untuk menandatangani file intisari dalam rentang waktu yang ditentukan:

```
aws cloudtrail list-public-keys --start-time 2016-01-01T20:30:00.000Z
```

Output:

```

{
  "PublicKeyList": [
    {
      "ValidityStartTime": 1453076702.0,
      "ValidityEndTime": 1455668702.0,
      "Value": "MIIBCgKCAQEAlSS3cl92HDycr/MTj0mo0has8habjrxaXw+Kz1WF0axSI2tcF
+3iJ9BKQAVSKxGwxwu3m0wG3J
+kU11xboEcEPHYoIYMbgfSw7KGnuDKwkLzsQWhUJ0cIb0HASox1vv/5fNXkrHhGbDCHeVXm804c83nvHUEFYThr1PfyP
+4WGDk+BGH5m9iuiAKkipEHWmU18/P7XpfpWQk4h8g3pXZ0rNXr081bh4d39svj7Uqdhv0XoBISp9t/
EXYuePGEtBdrKD9Dz+VHwyUPtBQvYr9BnkF88qBnaPNhS44rzwIDAQAB",
      "Fingerprint": "7f3f401420072e50a65a141430817ab3"
    }
  ]
}

```

- Untuk detail API, lihat [ListPublicKeys](#) di Referensi AWS CLI Perintah.

## list-tags

Contoh kode berikut menunjukkan cara menggunakan `list-tags`.

## AWS CLI

Untuk membuat daftar tag untuk jejak

`list-tags` Perintah berikut mencantumkan tag untuk `Trail1` dan `Trail2`:

```
aws cloudtrail list-tags --resource-id-list arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2
```

Output:

```
{
  "ResourceTagList": [
    {
      "ResourceId": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
      "TagsList": [
        {
          "Value": "Alice",
          "Key": "name"
        },
        {
          "Value": "us",
          "Key": "location"
        }
      ]
    },
    {
      "ResourceId": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2",
      "TagsList": [
        {
          "Value": "Bob",
          "Key": "name"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [ListTags](#) di Referensi AWS CLI Perintah.

## lookup-events

Contoh kode berikut menunjukkan cara menggunakan `lookup-events`.

### AWS CLI

Untuk mencari acara untuk jejak

`lookup-events` Perintah berikut mencari peristiwa aktivitas API berdasarkan atribut `EventName`:

```
aws cloudtrail lookup-events --lookup-attributes
  AttributeKey=EventName,AttributeValue=ConsoleLogin
```

Output:

```
{
  "Events": [
    {
      "EventId": "654ccbc0-ba0d-486a-9076-dbf7274677a7",
      "Username": "my-session-name",
      "EventTime": "2021-11-18T09:41:02-08:00",
      "CloudTrailEvent": "{\"eventVersion\":\"1.02\",\"userIdentity\":{\"type\":\
      \"AssumedRole\",\"principalId\":\"AR0AJIKPFTA72SWU4L7T4:my-session-name\",\"arn\":\
      \"arn:aws:sts::123456789012:assumed-role/my-role/my-session-name\",\"accountId\":\
      \"123456789012\",\"sessionContext\":{\"attributes\":{\"mfaAuthenticated\":\"false\
      \",\"creationDate\":\"2016-01-26T21:42:12Z\"},\"sessionIssuer\":{\"type\":\"Role\",
      \"principalId\":\"AR0AJIKPFTA72SWU4L7T4\",\"arn\":\"arn:aws:iam::123456789012:role/
      my-role\",\"accountId\":\"123456789012\",\"userName\":\"my-role\"}}},\"eventTime
      \":\"2016-01-26T21:42:12Z\",\"eventSource\":\"signin.amazonaws.com\",\"eventName\":\
      \"ConsoleLogin\",\"awsRegion\":\"us-east-1\",\"sourceIPAddress\":\"72.21.198.70\",
      \"userAgent\":\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5) AppleWebKit/537.36
      (KHTML, like Gecko) Chrome/47.0.2526.111 Safari/537.36\",\"requestParameters
      \":null,\"responseElements\":{\"ConsoleLogin\":{\"Success\"}},\"additionalEventData\":\
      {\"MobileVersion\":\"No\",\"MFAUsed\":\"No\"},\"eventID\":\"654ccbc0-ba0d-486a-9076-
      dbf7274677a7\",\"eventType\":\"AwsConsoleSignIn\",\"recipientAccountId\":\
      \"123456789012\"}",
      "EventName": "ConsoleLogin",
      "Resources": []
    }
  ]
}
```

- Untuk detail API, lihat [LookupEvents](#) di Referensi AWS CLI Perintah.



## put-event-selectors

Contoh kode berikut menunjukkan cara menggunakan `put-event-selectors`.

### AWS CLI

Untuk mengonfigurasi penyeleksi acara untuk jejak

Untuk membuat pemilih acara, jalankan perintah `put-event-selectors`. Ketika suatu peristiwa terjadi di akun Anda, CloudTrail evaluasi konfigurasi untuk jejak Anda. Jika acara cocok dengan pemilih acara apa pun untuk jejak, jejak akan memproses dan mencatat peristiwa tersebut. Anda dapat mengonfigurasi hingga 5 penyeleksi acara untuk jejak dan hingga 250 sumber daya data untuk jejak.

Contoh berikut membuat pemilih acara untuk jejak bernama "TrailName" untuk menyertakan peristiwa manajemen hanya-baca dan hanya tulis, peristiwa data untuk dua kombinasi bucket/awalan Amazon S3, dan peristiwa data untuk satu fungsi Lambda bernama "": AWS hello-world-python-function

```
aws cloudtrail put-event-selectors --trail-name TrailName --event-selectors '[{"ReadWriteType": "All","IncludeManagementEvents": true,"DataResources": [{"Type":"AWS::S3::Object", "Values": ["arn:aws:s3:::mybucket/prefix", "arn:aws:s3:::mybucket2/prefix2"]}, {"Type": "AWS::Lambda::Function","Values": ["arn:aws:lambda:us-west-2:999999999999:function:hello-world-python-function"]}]]'
```

Output:

```
{
  "EventSelectors": [
    {
      "IncludeManagementEvents": true,
      "DataResources": [
        {
          "Values": [
            "arn:aws:s3:::mybucket/prefix",
            "arn:aws:s3:::mybucket2/prefix2"
          ],
          "Type": "AWS::S3::Object"
        },
        {
          "Values": [
```

```

        "arn:aws:lambda:us-west-2:123456789012:function:hello-world-
python-function"
        ],
        "Type": "AWS::Lambda::Function"
    },
    ],
    "ReadWriteType": "All"
}
],
"TrailARN": "arn:aws:cloudtrail:us-east-2:123456789012:trail/TrailName"
}

```

Contoh berikut membuat pemilih peristiwa untuk jejak bernama "TrailName2" yang mencakup semua peristiwa, termasuk peristiwa manajemen hanya-baca dan hanya tulis, dan semua peristiwa data untuk semua bucket Amazon S3 dan fungsi Lambda di akun: AWS AWS

```

aws cloudtrail put-event-selectors --trail-name TrailName2 --event-selectors
' [{"ReadWriteType": "All", "IncludeManagementEvents": true, "DataResources":
[{"Type": "AWS::S3::Object", "Values": ["arn:aws:s3:::"}], {"Type":
"AWS::Lambda::Function", "Values": ["arn:aws:lambda"]}]} ]'

```

Output:

```

{
  "EventSelectors": [
    {
      "IncludeManagementEvents": true,
      "DataResources": [
        {
          "Values": [
            "arn:aws:s3:::"
          ],
          "Type": "AWS::S3::Object"
        },
        {
          "Values": [
            "arn:aws:lambda"
          ],
          "Type": "AWS::Lambda::Function"
        }
      ],
      "ReadWriteType": "All"
    }
  ]
}

```

```
    }  
  ],  
  "TrailARN": "arn:aws:cloudtrail:us-east-2:123456789012:trail/TrailName2"  
}
```

- Untuk detail API, lihat [PutEventSelectors](#) di Referensi AWS CLI Perintah.

## **remove-tags**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags`.

### AWS CLI

Untuk menghapus tag untuk jejak

`remove-tags` Perintah berikut menghapus tag yang ditentukan untuk `Trail1`:

```
aws cloudtrail remove-tags --resource-id arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 --tags-list Key=name Key=location
```

- Untuk detail API, lihat [RemoveTags](#) di Referensi AWS CLI Perintah.

## **start-logging**

Contoh kode berikut menunjukkan cara menggunakan `start-logging`.

### AWS CLI

Untuk memulai logging untuk jejak

`start-logging` Perintah berikut mengaktifkan logging untuk `Trail1`:

```
aws cloudtrail start-logging --name Trail1
```

- Untuk detail API, lihat [StartLogging](#) di Referensi AWS CLI Perintah.

## **stop-logging**

Contoh kode berikut menunjukkan cara menggunakan `stop-logging`.

## AWS CLI

Untuk berhenti mencatat jejak

`stop-logging` Perintah berikut mematikan logging untuk `Trail1`:

```
aws cloudtrail stop-logging --name Trail1
```

- Untuk detail API, lihat [StopLogging](#) di Referensi AWS CLI Perintah.

## update-subscription

Contoh kode berikut menunjukkan cara menggunakan `update-subscription`.

## AWS CLI

Untuk memperbarui pengaturan konfigurasi untuk jejak

`update-subscription` Perintah berikut memperbarui jejak untuk menentukan bucket S3 baru dan topik SNS:

```
aws cloudtrail update-subscription --name Trail1 --s3-new-bucket my-bucket-new --sns-new-topic my-topic-new
```

Output:

```
Setting up new S3 bucket my-bucket-new...
Setting up new SNS topic my-topic-new...
Creating/Updating CloudTrail configuration...
CloudTrail configuration:
{
  "trailList": [
    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail1",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket-new",
      "SnsTopicName": "my-topic-new",
      "HomeRegion": "us-east-1"
    }
  ]
}
```

```
],
"ResponseMetadata": {
"HTTPStatusCode": 200,
"RequestId": "31126f8a-c616-11e5-9cc6-2fd637936879"
}
}
```

- Untuk detail API, lihat [UpdateSubscription](#) di Referensi AWS CLI Perintah.

## update-trail

Contoh kode berikut menunjukkan cara menggunakan `update-trail`.

### AWS CLI

Untuk memperbarui jejak

`update-trail` Perintah berikut memperbarui jejak untuk menggunakan bucket yang ada untuk pengiriman log:

```
aws cloudtrail update-trail --name Trail1 --s3-bucket-name my-bucket
```

Output:

```
{
  "IncludeGlobalServiceEvents": true,
  "Name": "Trail1",
  "TrailARN": "arn:aws:cloudtrail:us-west-2:123456789012:trail/Trail1",
  "LogFileValidationEnabled": false,
  "IsMultiRegionTrail": true,
  "S3BucketName": "my-bucket"
}
```

- Untuk detail API, lihat [UpdateTrail](#) di Referensi AWS CLI Perintah.

## validate-logs

Contoh kode berikut menunjukkan cara menggunakan `validate-logs`.

### AWS CLI

Untuk memvalidasi file log

**validate-logs** Perintah berikut memvalidasi log untuk Trail1:

```
aws cloudtrail validate-logs --trail-arn arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 --start-time 20160129T19:00:00Z
```

Output:

```
Validating log files for trail arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 between 2016-01-29T19:00:00Z and 2016-01-29T22:15:43Z
Results requested for 2016-01-29T19:00:00Z to 2016-01-29T22:15:43Z
Results found for 2016-01-29T19:24:57Z to 2016-01-29T21:24:57Z:
3/3 digest files valid
15/15 log files valid
```

- Untuk detail API, lihat [ValidateLogs](#) di Referensi AWS CLI Perintah.

## CloudWatch contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CloudWatch.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-alarms**

Contoh kode berikut menunjukkan cara menggunakan `delete-alarms`.

## AWS CLI

Cara menghapus sebuah alarm

Contoh berikut menggunakan `delete-alarms` perintah untuk menghapus CloudWatch alarm Amazon bernama “myalarm”:

```
aws cloudwatch delete-alarms --alarm-names myalarm
```

Output:

```
This command returns to the prompt if successful.
```

- Untuk detail API, lihat [DeleteAlarms](#) di Referensi AWS CLI Perintah.

## describe-alarm-history

Contoh kode berikut menunjukkan cara menggunakan `describe-alarm-history`.

## AWS CLI

Cara mengambil riwayat untuk sebuah alarm

Contoh berikut menggunakan `describe-alarm-history` perintah untuk mengambil riwayat untuk CloudWatch alarm Amazon bernama “myalarm”:

```
aws cloudwatch describe-alarm-history --alarm-name "myalarm" --history-item-type StateUpdate
```

Output:

```
{
  "AlarmHistoryItems": [
    {
      "Timestamp": "2014-04-09T18:59:06.442Z",
      "HistoryItemType": "StateUpdate",
      "AlarmName": "myalarm",
      "HistoryData": "{\"version\":\"1.0\",\"oldState\":{\"stateValue\":\"ALARM\",\"stateReason\":\"testing purposes\"},\"newState\":{\"stateValue\":\"OK\",\"stateReason\":\"Threshold Crossed: 2 datapoints were not greater than the threshold (70.0). The most recent datapoints: [38.958, 40.292].\"},\"stateReasonData\":{\"version\":\"1.0\",\"queryDate\":\"2014-04-09T18:59:06.419+0000\",\"startDate
```

```

\":"2014-04-09T18:44:00.000+0000\","statistic\":"Average\","period\":"300,
\"recentDatapoints\":[38.958,40.292],\"threshold\":"70.0}}}],
  "HistorySummary": "Alarm updated from ALARM to OK"
},
{
  "Timestamp": "2014-04-09T18:59:05.805Z",
  "HistoryItemType": "StateUpdate",
  "AlarmName": "myalarm",
  "HistoryData": "{\"version\":"1.0\", \"oldState\":{\"stateValue
\":"OK\", \"stateReason\":"Threshold Crossed: 2 datapoints were
not greater than the threshold (70.0). The most recent datapoints:
[38.839999999999996, 39.714].\", \"stateReasonData\":{\"version\":"
\"1.0\", \"queryDate\":"2014-03-11T22:45:41.569+0000\", \"startDate\":"
\"2014-03-11T22:30:00.000+0000\", \"statistic\":"Average\", \"period\":"300,
\"recentDatapoints\":[38.839999999999996, 39.714], \"threshold\":"70.0}}, \"newState\":"
{\"stateValue\":"ALARM\", \"stateReason\":"testing purposes\"}}",
  "HistorySummary": "Alarm updated from OK to ALARM"
}
]
}

```

- Untuk detail API, lihat [DescribeAlarmHistory](#) di Referensi AWS CLI Perintah.

## describe-alarms-for-metric

Contoh kode berikut menunjukkan cara menggunakan `describe-alarms-for-metric`.

### AWS CLI

Untuk menampilkan informasi tentang alarm-alarm yang terkait dengan sebuah metrik

Contoh berikut menggunakan perintah `describe-alarms-for-metric` untuk menampilkan informasi tentang setiap alarm yang terkait dengan metrik CPUUtilization Amazon EC2 dan instans dengan ID `i-0c986c72`.

```
aws cloudwatch describe-alarms-for-metric --metric-name CPUUtilization --namespace
AWS/EC2 --dimensions Name=InstanceId,Value=i-0c986c72
```

### Output:

```
{
  "MetricAlarms": [
```



```

    {
      "EvaluationPeriods": 10,
      "AlarmArn": "arn:aws:cloudwatch:us-
east-1:111122223333:alarm:myHighCpuAlarm2",
      "StateUpdatedTimestamp": "2013-10-30T03:03:51.479Z",
      "AlarmConfigurationUpdatedTimestamp": "2013-10-30T03:03:50.865Z",
      "ComparisonOperator": "GreaterThanOrEqualToThreshold",
      "AlarmActions": [
        "arn:aws:sns:us-east-1:111122223333:NotifyMe"
      ],
      "Namespace": "AWS/EC2",
      "AlarmDescription": "CPU usage exceeds 70 percent",
      "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\\\"2013-10-30T03:03:51.479+0000\\\",\\\"startDate\\\":\\\"2013-10-30T02:08:00.000+0000\\\",
\\\"statistic\\\":\\\"Average\\\",\\\"period\\\":300,\\\"recentDatapoints\\\":
[40.698,39.612,42.432,39.796,38.816,42.28,42.854,40.088,40.760000000000005,41.316],
\\\"threshold\\\":70.0}\",
      "Period": 300,
      "StateValue": "OK",
      "Threshold": 70.0,
      "AlarmName": "myHighCpuAlarm2",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-0c986c72"
        }
      ],
      "Statistic": "Average",
      "StateReason": "Threshold Crossed: 10 datapoints were not greater than
or equal to the threshold (70.0). The most recent datapoints: [40.760000000000005,
41.316].",
      "InsufficientDataActions": [],
      "OKActions": [],
      "ActionsEnabled": true,
      "MetricName": "CPUUtilization"
    },
    {
      "EvaluationPeriods": 2,
      "AlarmArn": "arn:aws:cloudwatch:us-
east-1:111122223333:alarm:myHighCpuAlarm",
      "StateUpdatedTimestamp": "2014-04-09T18:59:06.442Z",
      "AlarmConfigurationUpdatedTimestamp": "2014-04-09T22:26:05.958Z",
      "ComparisonOperator": "GreaterThanThreshold",
      "AlarmActions": [

```

```

        "arn:aws:sns:us-east-1:111122223333:HighCPUAlarm"
    ],
    "Namespace": "AWS/EC2",
    "AlarmDescription": "CPU usage exceeds 70 percent",
    "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\\\"2014-04-09T18:59:06.419+0000\\\", \"startDate\": \"2014-04-09T18:44:00.000+0000\",
\\\"statistic\": \"Average\", \"period\": 300, \"recentDatapoints\": [38.958, 40.292],
\\\"threshold\": 70.0}\",
    "Period": 300,
    "StateValue": "OK",
    "Threshold": 70.0,
    "AlarmName": "myHighCpuAlarm",
    "Dimensions": [
        {
            "Name": "InstanceId",
            "Value": "i-0c986c72"
        }
    ],
    "Statistic": "Average",
    "StateReason": "Threshold Crossed: 2 datapoints were not greater than
the threshold (70.0). The most recent datapoints: [38.958, 40.292].",
    "InsufficientDataActions": [],
    "OKActions": [],
    "ActionsEnabled": false,
    "MetricName": "CPUUtilization"
    }
]
}

```

- Untuk detail API, lihat [DescribeAlarmsForMetric](#) di Referensi AWS CLI Perintah.

## describe-alarms

Contoh kode berikut menunjukkan cara menggunakan `describe-alarms`.

### AWS CLI

Cara membuat daftar informasi tentang sebuah alarm

Contoh berikut menggunakan perintah `describe-alarms` untuk memberikan informasi tentang alarm yang bernama "myalarm":

```
aws cloudwatch describe-alarms --alarm-names "myalarm"
```

**Output:**

```
{
  "MetricAlarms": [
    {
      "EvaluationPeriods": 2,
      "AlarmArn": "arn:aws:cloudwatch:us-east-1:123456789012:alarm:myalarm",
      "StateUpdatedTimestamp": "2014-04-09T18:59:06.442Z",
      "AlarmConfigurationUpdatedTimestamp": "2012-12-27T00:49:54.032Z",
      "ComparisonOperator": "GreaterThanThreshold",
      "AlarmActions": [
        "arn:aws:sns:us-east-1:123456789012:myHighCpuAlarm"
      ],
      "Namespace": "AWS/EC2",
      "AlarmDescription": "CPU usage exceeds 70 percent",
      "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":\"2014-04-09T18:59:06.419+0000\",\"startDate\":\"2014-04-09T18:44:00.000+0000\",\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":[38.958,40.292],\"threshold\":70.0}\",
      "Period": 300,
      "StateValue": "OK",
      "Threshold": 70.0,
      "AlarmName": "myalarm",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-0c986c72"
        }
      ],
      "Statistic": "Average",
      "StateReason": "Threshold Crossed: 2 datapoints were not greater than the threshold (70.0). The most recent datapoints: [38.958, 40.292].",
      "InsufficientDataActions": [],
      "OKActions": [],
      "ActionsEnabled": true,
      "MetricName": "CPUUtilization"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeAlarms](#) di Referensi AWS CLI Perintah.

## **disable-alarm-actions**

Contoh kode berikut menunjukkan cara menggunakan `disable-alarm-actions`.

### AWS CLI

Cara menonaktifkan tindakan untuk sebuah alarm

Contoh berikut menggunakan perintah `disable-alarm-actions` untuk menonaktifkan semua tindakan untuk alarm bernama `myalarm`..:

```
aws cloudwatch disable-alarm-actions --alarm-names myalarm
```

Perintah ini akan kembali ke prompt jika berhasil.

- Untuk detail API, lihat [DisableAlarmActions](#) di Referensi AWS CLI Perintah.

## **enable-alarm-actions**

Contoh kode berikut menunjukkan cara menggunakan `enable-alarm-actions`.

### AWS CLI

Cara mengaktifkan semua tindakan untuk sebuah alarm

Contoh berikut menggunakan perintah `enable-alarm-actions` untuk mengaktifkan semua tindakan untuk alarm bernama `myalarm`..:

```
aws cloudwatch enable-alarm-actions --alarm-names myalarm
```

Perintah ini akan kembali ke prompt jika berhasil.

- Untuk detail API, lihat [EnableAlarmActions](#) di Referensi AWS CLI Perintah.

## **get-metric-statistics**

Contoh kode berikut menunjukkan cara menggunakan `get-metric-statistics`.

### AWS CLI

Cara mendapatkan pemanfaatan CPU per instans EC2

Contoh berikut menggunakan perintah `get-metric-statistics` untuk mendapatkan pemanfaatan CPU untuk instans EC2 dengan ID `i-abcdef`.

```
aws cloudwatch get-metric-statistics --metric-name CPUUtilization --start-time
2014-04-08T23:18:00Z --end-time 2014-04-09T23:18:00Z --period 3600 --namespace AWS/
EC2 --statistics Maximum --dimensions Name=InstanceId,Value=i-abcdef
```

Output:

```
{
  "Datapoints": [
    {
      "Timestamp": "2014-04-09T11:18:00Z",
      "Maximum": 44.79,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T20:18:00Z",
      "Maximum": 47.92,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T19:18:00Z",
      "Maximum": 50.85,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T09:18:00Z",
      "Maximum": 47.92,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T03:18:00Z",
      "Maximum": 76.84,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T21:18:00Z",
      "Maximum": 48.96,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T14:18:00Z",
```

```
    "Maximum": 47.92,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T08:18:00Z",  
    "Maximum": 47.92,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T16:18:00Z",  
    "Maximum": 45.55,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T06:18:00Z",  
    "Maximum": 47.92,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T13:18:00Z",  
    "Maximum": 45.08,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T05:18:00Z",  
    "Maximum": 47.92,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T18:18:00Z",  
    "Maximum": 46.88,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T17:18:00Z",  
    "Maximum": 52.08,  
    "Unit": "Percent"  
  },  
  {  
    "Timestamp": "2014-04-09T07:18:00Z",  
    "Maximum": 47.92,  
    "Unit": "Percent"  
  },  
  {
```

```
    "Timestamp": "2014-04-09T02:18:00Z",
    "Maximum": 51.23,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T12:18:00Z",
    "Maximum": 47.67,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-08T23:18:00Z",
    "Maximum": 46.88,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T10:18:00Z",
    "Maximum": 51.91,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T04:18:00Z",
    "Maximum": 47.13,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T15:18:00Z",
    "Maximum": 48.96,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T00:18:00Z",
    "Maximum": 48.16,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T01:18:00Z",
    "Maximum": 49.18,
    "Unit": "Percent"
  }
],
"Label": "CPUUtilization"
}
```

## Menentukan beberapa dimensi

Contoh berikut menggambarkan cara menentukan beberapa dimensi. Masing-masing dimensi ditentukan sebagai sebuah pasangan Nama/Nilai, yang menggunakan koma antara nama dan nilai tersebut. Beberapa dimensi dipisahkan dengan satu spasi. Jika sebuah metrik mencakup beberapa dimensi, Anda harus menetapkan sebuah nilai untuk masing-masing dimensi yang ditetapkan.

Untuk contoh lainnya menggunakan `get-metric-statistics` perintah, lihat Mendapatkan Statistik untuk Metrik di Panduan CloudWatch Pengembang Amazon.

```
aws cloudwatch get-metric-statistics --metric-name Buffers --namespace MyNameSpace
--dimensions Name=InstanceID,Value=i-abcdef Name=InstanceType,Value=m1.small --
start-time 2016-10-15T04:00:00Z --end-time 2016-10-19T07:00:00Z --statistics Average
--period 60
```

- Untuk detail API, lihat [GetMetricStatistics](#) di Referensi AWS CLI Perintah.

## list-metrics

Contoh kode berikut menunjukkan cara menggunakan `list-metrics`.

### AWS CLI

Cara membuat daftar metrik untuk Amazon SNS

Contoh `list-metrics` berikut menampilkan metrik-metrik untuk Amazon SNS.

```
aws cloudwatch list-metrics \
--namespace "AWS/SNS"
```

Output:

```
{
  "Metrics": [
    {
      "Namespace": "AWS/SNS",
      "Dimensions": [
        {
          "Name": "TopicName",
          "Value": "NotifyMe"
        }
      ]
    }
  ]
}
```



```
    ],
    "MetricName": "PublishSize"
  },
  {
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "CF0"
      }
    ],
    "MetricName": "PublishSize"
  },
  {
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "NotifyMe"
      }
    ],
    "MetricName": "NumberOfNotificationsFailed"
  },
  {
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "NotifyMe"
      }
    ],
    "MetricName": "NumberOfNotificationsDelivered"
  },
  {
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "NotifyMe"
      }
    ],
    "MetricName": "NumberOfMessagesPublished"
  },
  {
```

```
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "CF0"
      }
    ],
    "MetricName": "NumberOfMessagesPublished"
  },
  {
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "CF0"
      }
    ],
    "MetricName": "NumberOfNotificationsDelivered"
  },
  {
    "Namespace": "AWS/SNS",
    "Dimensions": [
      {
        "Name": "TopicName",
        "Value": "CF0"
      }
    ],
    "MetricName": "NumberOfNotificationsFailed"
  }
]
```

- Untuk detail API, lihat [ListMetrics](#) di Referensi AWS CLI Perintah.

## **put-metric-alarm**

Contoh kode berikut menunjukkan cara menggunakan `put-metric-alarm`.

### AWS CLI

Untuk mengirim sebuah pesan email Layanan Notifikasi Sederhana Amazon saat pemanfaatan CPU melebihi 70 persen

Contoh berikut menggunakan perintah `put-metric-alarm` untuk mengirim sebuah pesan email Amazon Simple Notification Service ketika penggunaan CPU melebihi 70 persen:

```
aws cloudwatch put-metric-alarm --alarm-name cpu-mon --alarm-description "Alarm
when CPU exceeds 70 percent" --metric-name CPUUtilization --namespace AWS/
EC2 --statistic Average --period 300 --threshold 70 --comparison-operator
GreaterThanOrEqualToThreshold --dimensions "Name=InstanceId,Value=i-12345678" --evaluation-
periods 2 --alarm-actions arn:aws:sns:us-east-1:111122223333:MyTopic --unit Percent
```

Perintah ini akan kembali ke prompt jika berhasil. Jika suatu alarm yang memiliki nama yang sama sudah ada, alarm itu akan ditimpa oleh alarm yang baru.

Cara menentukan beberapa dimensi

Contoh berikut menggambarkan cara menentukan beberapa dimensi. Masing-masing dimensi ditentukan sebagai sebuah pasangan Nama/Nilai, yang menggunakan koma antara nama dan nilai tersebut. Beberapa dimensi dipisahkan dengan satu spasi:

```
aws cloudwatch put-metric-alarm --alarm-name "Default_Test_Alarm3" --alarm-
description "The default example alarm" --namespace "CW EXAMPLE METRICS" --
metric-name Default_Test --statistic Average --period 60 --evaluation-periods 3
--threshold 50 --comparison-operator GreaterThanOrEqualToThreshold --dimensions
Name=key1,Value=value1 Name=key2,Value=value2
```

- Untuk detail API, lihat [PutMetricAlarm](#) di Referensi AWS CLI Perintah.

## **put-metric-data**

Contoh kode berikut menunjukkan cara menggunakan `put-metric-data`.

### AWS CLI

Untuk mempublikasikan metrik khusus ke Amazon CloudWatch

Contoh berikut menggunakan `put-metric-data` perintah untuk menerbitkan metrik kustom ke Amazon CloudWatch:

```
aws cloudwatch put-metric-data --namespace "Usage Metrics" --metric-data file://
metric.json
```

Nilai-nilai untuk metrik itu sendiri disimpan dalam file JSON, `metric.json`.

Berikut adalah isi dari file tersebut:

```
[
  {
    "MetricName": "New Posts",
    "Timestamp": "Wednesday, June 12, 2013 8:28:20 PM",
    "Value": 0.50,
    "Unit": "Count"
  }
]
```

Untuk informasi selengkapnya, lihat [Menerbitkan Metrik Kustom di Panduan CloudWatch Pengembang Amazon](#).

Cara menentukan beberapa dimensi

Contoh berikut menggambarkan cara menentukan beberapa dimensi. Masing-masing dimensi ditentukan sebagai sebuah pasangan `Name=Value`. Beberapa dimensi dipisahkan menggunakan koma.:

```
aws cloudwatch put-metric-data --metric-name Buffers --namespace MyNameSpace --unit Bytes --value 231434333 --dimensions InstanceID=1-23456789,InstanceType=m1.small
```

- Untuk detail API, lihat [PutMetricData](#) di Referensi AWS CLI Perintah.

## set-alarm-state

Contoh kode berikut menunjukkan cara menggunakan `set-alarm-state`.

### AWS CLI

Untuk sementara mengubah status alarm

Contoh berikut menggunakan `set-alarm-state` perintah untuk sementara mengubah status CloudWatch alarm Amazon bernama "myalarm" dan mengaturnya ke status ALARM untuk tujuan pengujian:

```
aws cloudwatch set-alarm-state --alarm-name "myalarm" --state-value ALARM --state-reason "testing purposes"
```

Perintah ini akan kembali ke prompt jika berhasil.

- Untuk detail API, lihat [SetAlarmState](#) di Referensi AWS CLI Perintah.

## CloudWatch Log contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With CloudWatch Logs.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-log-group**

Contoh kode berikut menunjukkan cara menggunakan `create-log-group`.

AWS CLI

Perintah berikut membuat grup log bernama `my-logs`:

```
aws logs create-log-group --log-group-name my-logs
```

- Untuk detail API, lihat [CreateLogGroup](#) di Referensi AWS CLI Perintah.

### **create-log-stream**

Contoh kode berikut menunjukkan cara menggunakan `create-log-stream`.

## AWS CLI

Perintah berikut membuat aliran log bernama 20150601 dalam grup logmy-logs:

```
aws logs create-log-stream --log-group-name my-logs --log-stream-name 20150601
```

- Untuk detail API, lihat [CreateLogStream](#) di Referensi AWS CLI Perintah.

## delete-log-group

Contoh kode berikut menunjukkan cara menggunakan `delete-log-group`.

## AWS CLI

Perintah berikut menghapus grup log bernama my-logs:

```
aws logs delete-log-group --log-group-name my-logs
```

- Untuk detail API, lihat [DeleteLogGroup](#) di Referensi AWS CLI Perintah.

## delete-log-stream

Contoh kode berikut menunjukkan cara menggunakan `delete-log-stream`.

## AWS CLI

Perintah berikut menghapus aliran log bernama 20150531 dari grup log bernama my-logs:

```
aws logs delete-log-stream --log-group-name my-logs --log-stream-name 20150531
```

- Untuk detail API, lihat [DeleteLogStream](#) di Referensi AWS CLI Perintah.

## delete-retention-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-retention-policy`.

## AWS CLI

Perintah berikut menghapus kebijakan retensi yang sebelumnya telah diterapkan ke grup log bernama my-logs:

```
aws logs delete-retention-policy --log-group-name my-logs
```

- Untuk detail API, lihat [DeleteRetentionPolicy](#) di Referensi AWS CLI Perintah.

## describe-log-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-log-groups`.

### AWS CLI

Perintah berikut menjelaskan grup log bernama `my-logs`:

```
aws logs describe-log-groups --log-group-name-prefix my-logs
```

Output:

```
{
  "logGroups": [
    {
      "storedBytes": 0,
      "metricFilterCount": 0,
      "creationTime": 1433189500783,
      "logGroupName": "my-logs",
      "retentionInDays": 5,
      "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:*"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeLogGroups](#) di Referensi AWS CLI Perintah.

## describe-log-streams

Contoh kode berikut menunjukkan cara menggunakan `describe-log-streams`.

### AWS CLI

Perintah berikut menunjukkan semua aliran log yang dimulai dengan awalan 2015 di grup log: `my-logs`

```
aws logs describe-log-streams --log-group-name my-logs --log-stream-name-prefix 2015
```

Output:

```
{
  "logStreams": [
    {
      "creationTime": 1433189871774,
      "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:log-stream:20150531",
      "logStreamName": "20150531",
      "storedBytes": 0
    },
    {
      "creationTime": 1433189873898,
      "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:log-stream:20150601",
      "logStreamName": "20150601",
      "storedBytes": 0
    }
  ]
}
```

- Untuk detail API, lihat [DescribeLogStreams](#) di Referensi AWS CLI Perintah.

## get-log-events

Contoh kode berikut menunjukkan cara menggunakan `get-log-events`.

### AWS CLI

Perintah berikut mengambil peristiwa log dari aliran log bernama `20150601` dalam grup `my-logs` log:

```
aws logs get-log-events --log-group-name my-logs --log-stream-name 20150601
```

Output:

```
{
  "nextForwardToken":
  "f/31961209122447488583055879464742346735121166569214640130",
```



```
"events": [
  {
    "ingestionTime": 1433190494190,
    "timestamp": 1433190184356,
    "message": "Example Event 1"
  },
  {
    "ingestionTime": 1433190516679,
    "timestamp": 1433190184356,
    "message": "Example Event 1"
  },
  {
    "ingestionTime": 1433190494190,
    "timestamp": 1433190184358,
    "message": "Example Event 2"
  }
],
"nextBackwardToken":
"b/31961209122358285602261756944988674324553373268216709120"
}
```

- Untuk detail API, lihat [GetLogEvents](#) di Referensi AWS CLI Perintah.

## put-log-events

Contoh kode berikut menunjukkan cara menggunakan `put-log-events`.

### AWS CLI

Perintah berikut menempatkan peristiwa log ke aliran log bernama `20150601` dalam grup `logmy-logs`:

```
aws logs put-log-events --log-group-name my-logs --log-stream-name 20150601 --log-events file://events
```

Output:

```
{
  "nextSequenceToken": "49542672486831074009579604567656788214806863282469607346"
}
```

Contoh di atas membaca array JSON peristiwa dari file bernama `events` dalam direktori saat ini:

```
[
  {
    "timestamp": 1433190184356,
    "message": "Example Event 1"
  },
  {
    "timestamp": 1433190184358,
    "message": "Example Event 2"
  },
  {
    "timestamp": 1433190184360,
    "message": "Example Event 3"
  }
]
```

Setiap panggilan berikutnya memerlukan token urutan berikutnya yang disediakan oleh panggilan sebelumnya untuk ditentukan dengan opsi token urutan:

```
aws logs put-log-events --log-group-name my-logs --log-stream-
name 20150601 --log-events file://events2 --sequence-token
"49542672486831074009579604567656788214806863282469607346"
```

Output:

```
{
  "nextSequenceToken": "49542672486831074009579604567900991230369019956308219826"
}
```

- Untuk detail API, lihat [PutLogEvents](#) di Referensi AWS CLI Perintah.

## put-retention-policy

Contoh kode berikut menunjukkan cara menggunakan `put-retention-policy`.

### AWS CLI

Perintah berikut menambahkan kebijakan retensi 5 hari ke grup log bernama `my-logs`:

```
aws logs put-retention-policy --log-group-name my-logs --retention-in-days 5
```

- Untuk detail API, lihat [PutRetentionPolicy](#) di Referensi AWS CLI Perintah.

## CloudWatch Contoh Pemantauan Jaringan menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CloudWatch Network Monitoring.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-probe**

Contoh kode berikut menunjukkan cara menggunakan `create-probe`.

AWS CLI

Contoh 1: Untuk membuat probe yang menggunakan TCP dan menambahkannya ke monitor jaringan

`create-probe` Contoh berikut membuat probe yang menggunakan TCP protocol dan menambahkan probe ke monitor bernama `Example_NetworkMonitor`. Setelah dibuat, monitor dengan probe akan `PENDING` sampai monitor berada `ACTIVE`. `state` Ini mungkin memakan waktu beberapa menit, di mana status akan berubah `ACTIVE`, dan Anda dapat mulai melihat CloudWatch metrik.

```
aws networkmonitor create-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe sourceArn=arn:aws:ec2:region:111122223333:subnet/subnet-  
id,destination=10.0.0.100,destinationPort=80,protocol=TCP,packetSize=56,tags={Name=Probe1}
```

Output:

```
{
  "probeId": "probe-12345",
  "probeArn": "arn:aws:networkmonitor:region:111122223333:probe/probe-12345",
  "destination": "10.0.0.100",
  "destinationPort": 80,
  "packetSize": 56,
  "addressFamily": "IPV4",
  "vpcId": "vpc-12345",
  "state": "PENDING",
  "createdAt": "2024-03-29T12:41:57.314000-04:00",
  "modifiedAt": "2024-03-29T12:41:57.314000-04:00",
  "tags": {
    "Name": "Probe1"
  }
}
```

Contoh 2: Untuk membuat probe yang menggunakan probe menggunakan ICMP dan menambahkannya ke monitor jaringan

`create-probe` Contoh berikut membuat probe yang menggunakan ICMP protocol dan menambahkan probe ke monitor bernama `Example_NetworkMonitor`. Setelah dibuat, monitor dengan probe akan `PENDING` sampai monitor berada `ACTIVE`. state ini mungkin memakan waktu beberapa menit, di mana status akan berubah `ACTIVE`, dan Anda dapat mulai melihat `CloudWatch` metrik.

```
aws networkmonitor create-probe \
  --monitor-name Example_NetworkMonitor \
  --probe sourceArn=arn:aws:ec2:region:012345678910:subnet/subnet-
id,destination=10.0.0.100,protocol=ICMP,packetSize=56,tags={Name=Probe1}
```

Output:

```
{
  "probeId": "probe-12345",
  "probeArn": "arn:aws:networkmonitor:region:111122223333:probe/probe-12345",
  "destination": "10.0.0.100",
  "packetSize": 56,
  "addressFamily": "IPV4",
  "vpcId": "vpc-12345",
  "state": "PENDING",
  "createdAt": "2024-03-29T12:44:02.452000-04:00",
```

```
"modifiedAt": "2024-03-29T12:44:02.452000-04:00",
"tags": {
  "Name": "Probe1"
}
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

- Untuk detail API, lihat [CreateProbe](#) di Referensi AWS CLI Perintah.

## **delete-monitor**

Contoh kode berikut menunjukkan cara menggunakan `delete-monitor`.

### AWS CLI

Untuk menghapus monitor

`delete-monitor` Contoh berikut menghapus monitor bernama `Example_NetworkMonitor`.

```
aws networkmonitor delete-monitor \
  --monitor-name Example_NetworkMonitor
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

- Untuk detail API, lihat [DeleteMonitor](#) di Referensi AWS CLI Perintah.

## **delete-probe**

Contoh kode berikut menunjukkan cara menggunakan `delete-probe`.

### AWS CLI

Untuk menghapus probe

`delete-probe` Contoh berikut menghapus probe dengan ID `probe-12345` dari monitor jaringan bernama `Example_NetworkMonitor`.

```
aws networkmonitor delete-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe-id probe-12345
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

- Untuk detail API, lihat [DeleteProbe](#) di Referensi AWS CLI Perintah.

## get-probe

Contoh kode berikut menunjukkan cara menggunakan `get-probe`.

### AWS CLI

Untuk melihat detail probe

`get-probe` Contoh berikut mengembalikan rincian tentang probe dengan probeID `probe-12345` yang terkait dengan monitor bernama `Example_NetworkMonitor`.

```
aws networkmonitor get-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe-id probe-12345
```

Output:

```
{  
  "probeId": "probe-12345",  
  "probeArn": "arn:aws:networkmonitor:region:012345678910:probe/probe-12345",  
  "sourceArn": "arn:aws:ec2:region:012345678910:subnet/subnet-12345",  
  "destination": "10.0.0.100",  
  "destinationPort": 80,  
  "protocol": "TCP",  
  "packetSize": 56,  
  "addressFamily": "IPV4",  
  "vpcId": "vpc-12345",  
  "state": "ACTIVE",  
  "createdAt": "2024-03-29T12:41:57.314000-04:00",  
  "modifiedAt": "2024-03-29T12:42:28.610000-04:00",  
  "tags": {
```

```
    "Name": "Probe1"
  }
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

- Untuk detail API, lihat [GetProbed](#) di Referensi AWS CLI Perintah.

## list-monitors

Contoh kode berikut menunjukkan cara menggunakan `list-monitors`.

### AWS CLI

Contoh 1: Untuk daftar semua monitor (monitor tunggal)

`list-monitors` Contoh berikut mengembalikan daftar hanya satu monitor. Monitor state ini ACTIVE dan memiliki `aggregationPeriod` 60 detik.

```
aws networkmonitor list-monitors
```

Output:

```
{
  "monitors": [{
    "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor",
    "monitorName": "Example_NetworkMonitor",
    "state": "ACTIVE",
    "aggregationPeriod": 60,
    "tags": {
      "Monitor": "Monitor1"
    }
  }
]
```

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

Contoh 2: Untuk daftar semua monitor (beberapa monitor)

`list-monitors` Contoh berikut mengembalikan daftar tiga monitor. Salah state satu monitor adalah ACTIVE dan menghasilkan CloudWatch metrik. Status dari dua monitor lainnya adalah INACTIVE dan tidak menghasilkan CloudWatch metrik. Ketiga monitor menggunakan `aggregationPeriod` 60 detik.

```
aws networkmonitor list-monitors
```

Output:

```
{
  "monitors": [
    {
      "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
Example_NetworkMonitor",
      "monitorName": "Example_NetworkMonitor",
      "state": "INACTIVE",
      "aggregationPeriod": 60,
      "tags": {}
    },
    {
      "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
Example_NetworkMonitor2",
      "monitorName": "Example_NetworkMonitor2",
      "state": "ACTIVE",
      "aggregationPeriod": 60,
      "tags": {
        "Monitor": "Monitor1"
      }
    },
    {
      "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
TestNetworkMonitor_CLI",
      "monitorName": "TestNetworkMonitor_CLI",
      "state": "INACTIVE",
      "aggregationPeriod": 60,
      "tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.



- Untuk detail API, lihat [ListMonitors](#) di Referensi AWS CLI Perintah.

## update-monitor

Contoh kode berikut menunjukkan cara menggunakan `update-monitor`.

### AWS CLI

Untuk memperbarui monitor

`update-monitor` Contoh berikut mengubah monitor `aggregationPeriod` dari 60 detik ke 30 detik.

```
aws networkmonitor update-monitor \  
  --monitor-name Example_NetworkMonitor \  
  --aggregation-period 30
```

Output:

```
{  
  "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor",  
  "monitorName": "Example_NetworkMonitor",  
  "state": "PENDING",  
  "aggregationPeriod": 30,  
  "tags": {  
    "Monitor": "Monitor1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

- Untuk detail API, lihat [UpdateMonitor](#) di Referensi AWS CLI Perintah.

## update-probe

Contoh kode berikut menunjukkan cara menggunakan `update-probe`.

### AWS CLI

Untuk memperbarui probe

update-probeContoh berikut memperbarui alamat destination IP asli probe dan juga memperbarui packetSize ke60.

```
aws networkmonitor update-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe-id probe-12345 \  
  --destination 10.0.0.150 \  
  --packet-size 60
```

Output:

```
{  
  "probeId": "probe-12345",  
  "probeArn": "arn:aws:networkmonitor:region:012345678910:probe/probe-12345",  
  "sourceArn": "arn:aws:ec2:region:012345678910:subnet/subnet-12345",  
  "destination": "10.0.0.150",  
  "destinationPort": 80,  
  "protocol": "TCP",  
  "packetSize": 60,  
  "addressFamily": "IPV4",  
  "vpcId": "vpc-12345",  
  "state": "PENDING",  
  "createdAt": "2024-03-29T12:41:57.314000-04:00",  
  "modifiedAt": "2024-03-29T13:52:23.115000-04:00",  
  "tags": {  
    "Name": "Probe1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja Monitor CloudWatch Jaringan Amazon](#) di Panduan CloudWatch Pengguna Amazon.

- Untuk detail API, lihat [UpdateProbe](#) di Referensi AWS CLI Perintah.

## CodeArtifact contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CodeArtifact.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-external-connection**

Contoh kode berikut menunjukkan cara menggunakan `associate-external-connection`.

AWS CLI

Untuk menambahkan koneksi eksternal ke repositori

`associate-external-connection` Contoh berikut menambahkan koneksi eksternal ke `npmjs.com` ke repositori bernama `test-repo`.

```
aws codeartifact associate-external-connection \
  --repository test-repo \
  --domain test-domain \
  --external-connection public:npmjs
```

Output:

```
{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo",
    "upstreams": [],
    "externalConnections": [
      {
        "externalConnectionName": "public:npmjs",
        "packageFormat": "npm",
```

```
    "status": "AVAILABLE"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Menambahkan koneksi eksternal](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [AssociateExternalConnection](#) di Referensi AWS CLI Perintah.

## copy-package-versions

Contoh kode berikut menunjukkan cara menggunakan `copy-package-versions`.

### AWS CLI

Untuk menyalin versi paket dari satu repositori ke repositori lainnya

Berikut ini `copy-package-versions` memindahkan versi 4.0.0 dan 5.0.0 dari paket bernama `test-package` dari `my-repo` ke `test-repo`.

```
aws codeartifact copy-package-versions \
  --domain test-domain \
  --source-repository my-repo \
  --destination-repository test-repo \
  --format npm \
  --package test-package \
  --versions '["4.0.0", "5.0.0"]'
```

Output:

```
{
  "format": "npm",
  "package": "test-package",
  "versions": [
    {
      "version": "5.0.0",
      "revision": "REVISION-1-SAMPLE-6C81EFF7DA55CC",
      "status": "Published"
    },
    {
      "version": "4.0.0",
```

```
    "revision": "REVISION-2-SAMPLE-55C752BEE772FC",
    "status": "Published"
  }
]
```

Untuk informasi selengkapnya, lihat [Menyalin paket antar repositori](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [CopyPackageVersions](#) di Referensi AWS CLI Perintah.

## create-domain

Contoh kode berikut menunjukkan cara menggunakan `create-domain`.

### AWS CLI

Untuk membuat domain

`create-domain` Contoh berikut membuat domain bernama `test-domain`.

```
aws codeartifact create-domain \
  --domain test-domain
```

Output:

```
{
  "domain": {
    "name": "test-domain",
    "owner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:domain/test-domain",
    "status": "Active",
    "createdTime": "2020-10-20T13:16:48.559000-04:00",
    "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryCount": 0,
    "assetSizeBytes": 0
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [CreateDomain](#) di Referensi AWS CLI Perintah.

## create-repository

Contoh kode berikut menunjukkan cara menggunakan `create-repository`.

### AWS CLI

Untuk membuat repositori

`create-repository` Contoh berikut membuat repositori bernama `test-repo` di dalam domain bernama `test-domain`.

```
aws codeartifact create-repository \
  --domain test-domain \
  --domain-owner 111122223333 \
  --repository test-repo \
  --description "This is a test repository."
```

Output:

```
{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo",
    "description": "This is a test repository.",
    "upstreams": [],
    "externalConnections": []
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [CreateRepository](#) di Referensi AWS CLI Perintah.

## delete-domain-permissions-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-domain-permissions-policy`.

## AWS CLI

Untuk menghapus dokumen kebijakan izin dari domain

`delete-domain-permissions-policy` Contoh berikut menghapus kebijakan izin dari domain bernama `test-domain`.

```
aws codeartifact delete-domain-permissions-policy \  
  --domain test-domain
```

Output:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "BasicDomainPolicy",  
      "Action": [  
        "codeartifact:GetDomainPermissionsPolicy",  
        "codeartifact:ListRepositoriesInDomain",  
        "codeartifact:GetAuthorizationToken",  
        "codeartifact:CreateRepository"  
      ],  
      "Effect": "Allow",  
      "Resource": "*",  
      "Principal": {  
        "AWS": "arn:aws:iam::111122223333:root"  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menghapus kebijakan domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DeleteDomainPermissionsPolicy](#) di Referensi AWS CLI Perintah.

### `delete-domain`

Contoh kode berikut menunjukkan cara menggunakan `delete-domain`.

## AWS CLI

Untuk menghapus domain

`delete-domain` Contoh berikut menghapus domain bernama `test-domain`.

```
aws codeartifact delete-domain \  
  --domain test-domain
```

Output:

```
{  
  "domain": {  
    "name": "test-domain",  
    "owner": "417498243647",  
    "arn": "arn:aws:codeartifact:us-west-2:417498243647:domain/test-domain",  
    "status": "Deleted",  
    "createdTime": "2020-10-20T13:16:48.559000-04:00",  
    "encryptionKey": "arn:aws:kms:us-west-2:417498243647:key/c9fe2447-0795-4fda-  
afbe-8464574ae162",  
    "repositoryCount": 0,  
    "assetSizeBytes": 0  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DeleteDomain](#) di Referensi AWS CLI Perintah.

## `delete-package-versions`

Contoh kode berikut menunjukkan cara menggunakan `delete-package-versions`.

## AWS CLI

Untuk menghapus versi paket

`delete-package-versions` Contoh berikut menghapus versi 4.0.0 dari paket bernama `test-package`.

```
aws codeartifact delete-package-versions \  
  --package test-package
```



```
--domain test-domain \  
--repo test-repo \  
--format npm \  
--package test-package \  
--versions 4.0.0
```

Output:

```
{  
  "successfulVersions": {  
    "4.0.0": {  
      "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",  
      "status": "Deleted"  
    }  
  },  
  "failedVersions": {}  
}
```

Untuk informasi selengkapnya, lihat [Menghapus versi paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DeletePackageVersions](#) di Referensi AWS CLI Perintah.

## **delete-repository-permissions-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-repository-permissions-policy`.

AWS CLI

Untuk menghapus kebijakan izin dari repositori

`delete-repository-permissions-policy` Contoh berikut menghapus kebijakan izin dari repositori bernama `test-repo`.

```
aws codeartifact delete-repository-permissions-policy \  
  --domain test-domain \  
  --repository test-repo
```

Output:

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:root"
    },
    "Action": [
      "codeartifact:DescribePackageVersion",
      "codeartifact:DescribeRepository",
      "codeartifact:GetPackageVersionReadme",
      "codeartifact:GetRepositoryEndpoint",
      "codeartifact:ListPackages",
      "codeartifact:ListPackageVersions",
      "codeartifact:ListPackageVersionAssets",
      "codeartifact:ListPackageVersionDependencies",
      "codeartifact:ReadFromRepository"
    ],
    "Resource": "*"
  }
]
```

Untuk informasi selengkapnya, lihat [Menghapus kebijakan](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DeleteRepositoryPermissionsPolicy](#) di Referensi AWS CLI Perintah.

## delete-repository

Contoh kode berikut menunjukkan cara menggunakan `delete-repository`.

### AWS CLI

Untuk menghapus repositori

`delete-repository` Contoh berikut menghapus repositori bernama `test-repo` dalam domain bernama `test-domain`

```
aws codeartifact delete-repository \
  --domain test-domain \
  --repository test-repo
```

**Output:**

```
{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo",
    "description": "This is a test repository",
    "upstreams": [],
    "externalConnections": []
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus repositori](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [DeleteRepository](#) di Referensi AWS CLI Perintah.

**describe-domain**

Contoh kode berikut menunjukkan cara menggunakan `describe-domain`.

**AWS CLI**

Untuk mendapatkan informasi tentang domain

`describe-domain` Contoh berikut mengembalikan `DomainDescription` objek untuk domain bernama `test-domain`.

```
aws codeartifact describe-domain \
  --domain test-domain
```

**Output:**

```
{
  "domain": {
    "name": "test-domain",
    "owner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:domain/test-domain",
```

```

    "status": "Active",
    "createdTime": "2020-10-20T13:16:48.559000-04:00",
    "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    "repositoryCount": 2,
    "assetSizeBytes": 0,
    "s3BucketArn": "arn:aws:s3:::assets-111122223333-us-west-2"
  }
}

```

Untuk informasi selengkapnya, lihat [Ikhtisar domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DescribeDomain](#) di Referensi AWS CLI Perintah.

## describe-repository

Contoh kode berikut menunjukkan cara menggunakan `describe-repository`.

### AWS CLI

Untuk mendapatkan informasi tentang repositori

`describe-repository` Contoh berikut mengembalikan `RepositoryDescription` objek untuk repositori bernama `test-repo`.

```

aws codeartifact describe-repository \
  --domain test-domain \
  --repository test-repo

```

Output:

```

{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
    "description": "This is a test repository.",
    "upstreams": [],
    "externalConnections": []
  }
}

```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Membuat domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DescribeRepository](#) di Referensi AWS CLI Perintah.

## disassociate-external-connection

Contoh kode berikut menunjukkan cara menggunakan `disassociate-external-connection`.

### AWS CLI

Untuk menghapus koneksi eksternal dari repositori

`disassociate-external-connection` Contoh berikut menghapus koneksi eksternal ke `npmjs.com` dari repositori bernama `test-repo`.

```
aws codeartifact disassociate-external-connection \  
  --repository test-repo \  
  --domain test-domain \  
  --external-connection public:npmjs
```

### Output:

```
{  
  "repository": {  
    "name": "test-repo",  
    "administratorAccount": "111122223333",  
    "domainName": "test-domain",  
    "domainOwner": "111122223333",  
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/  
test-repo",  
    "upstreams": [],  
    "externalConnections": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus sambungan eksternal](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [DisassociateExternalConnection](#) di Referensi AWS CLI Perintah.

## dispose-package-versions

Contoh kode berikut menunjukkan cara menggunakan `dispose-package-versions`.

### AWS CLI

Untuk menghapus aset versi paket dan menyetel statusnya ke Disposed

`dispose-package-versions` Contoh berikut menghapus aset paket uji versi 4.0.0 dan menetapkan statusnya ke Disposed.

```
aws codeartifact dispose-package-versions \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --versions 4.0.0
```

Output:

```
{
  "successfulVersions": {
    "4.0.0": {
      "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
      "status": "Disposed"
    }
  },
  "failedVersions": {}
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan paket CodeArtifact di Panduan AWS CodeArtifact Pengguna](#).

- Untuk detail API, lihat [DisposePackageVersions](#) di Referensi AWS CLI Perintah.

## get-authorization-token

Contoh kode berikut menunjukkan cara menggunakan `get-authorization-token`.

### AWS CLI

Untuk mendapatkan token otorisasi

`get-authorization-token` Contoh berikut mengambil token CodeArtifact otorisasi.

```
aws codeartifact get-authorization-token \  
  --domain test-domain \  
  --query authorizationToken \  
  --output text
```

Output:

```
This command will return the authorization token. You can store the output in an  
environment variable when calling the command.
```

Untuk informasi selengkapnya, lihat [Mengkonfigurasi pip tanpa perintah login](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [GetAuthorizationToken](#) di Referensi AWS CLI Perintah.

## `get-domain-permissions-policy`

Contoh kode berikut menunjukkan cara menggunakan `get-domain-permissions-policy`.

AWS CLI

Untuk mendapatkan dokumen kebijakan izin untuk domain

`get-domain-permissions-policy` Contoh berikut mendapatkan kebijakan izin yang dilampirkan ke domain bernama `test-domain`.

```
aws codeartifact get-domain-permissions-policy \  
  --domain test-domain
```

Output:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "BasicDomainPolicy",  
      "Action": [  
        "codeartifact:GetDomainPermissionsPolicy",  
        "codeartifact:ListRepositoriesInDomain",  
        "codeartifact:GetAuthorizationToken",
```

```

        "codeartifact:CreateRepository"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
    }
}
]
}

```

Untuk informasi [selengkapnya, lihat Membaca kebijakan domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [GetDomainPermissionsPolicy](#) di Referensi AWS CLI Perintah.

## get-package-version-asset

Contoh kode berikut menunjukkan cara menggunakan `get-package-version-asset`.

### AWS CLI

Untuk mendapatkan aset dari versi paket

`get-package-version-asset` Contoh berikut mengambil `package.tgz` aset untuk versi 4.0.0 dari paket npm bernama `test-package`.

```

aws codeartifact get-package-version-asset \
  --domain test-domain \
  --repository test-repo \
  --format npm \
  --package test-package \
  --package-version 4.0.0 \
  --asset 'package.tgz' \
  outfileName

```

### Output:

The output for this command will also store the raw asset in the file provided in place of `outfileName`.

```

{
  "assetName": "package.tgz",

```



```

    "packageVersion": "4.0.0",
    "packageVersionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs="
  }

```

Untuk informasi selengkapnya, lihat [Daftar aset versi paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [GetPackageVersionAsset](#) di Referensi AWS CLI Perintah.

## get-package-version-readme

Contoh kode berikut menunjukkan cara menggunakan `get-package-version-readme`.

### AWS CLI

Untuk mendapatkan file readme versi paket

`get-package-version-readme` Contoh berikut mengambil file readme untuk versi 4.0.0 dari paket npm bernama `test-package`.

```

aws codeartifact get-package-version-readme \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --package-version 4.0.0

```

Output:

```

{
  "format": "npm",
  "package": "test-package",
  "version": "4.0.0",
  "readme": "<div align=\"center\">\n  <a href=\"https://github.com/test-package/testpack\"> ... more content ... \n",
  "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs="
}

```

Untuk informasi selengkapnya, [lihat Melihat file readme versi paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [GetPackageVersionReadme](#) di Referensi AWS CLI Perintah.

## get-repository-endpoint

Contoh kode berikut menunjukkan cara menggunakan `get-repository-endpoint`.

### AWS CLI

Untuk mendapatkan titik akhir URL repositori

`get-repository-endpoint` Contoh berikut mengembalikan endpoint npm untuk repositori test-repo.

```
aws codeartifact get-repository-endpoint \  
  --domain test-domain \  
  --repository test-repo \  
  --format npm
```

Output:

```
{  
  "repositoryEndpoint": "https://test-domain-111122223333.d.codeartifact.us-  
west-2.amazonaws.com/npm/test-repo/"  
}
```

Untuk informasi selengkapnya, lihat [Connect ke repositori](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [GetRepositoryEndpoint](#) di Referensi AWS CLI Perintah.

## get-repository-permissions-policy

Contoh kode berikut menunjukkan cara menggunakan `get-repository-permissions-policy`.

### AWS CLI

Untuk mendapatkan dokumen kebijakan izin untuk repositori

`get-repository-permissions-policy` Contoh berikut mendapatkan kebijakan izin yang dilampirkan ke repositori bernama test-repo.

```
aws codeartifact get-repository-permissions-policy \  
  --domain test-domain \  
  --repository test-repo
```

```
--repository test-repo
```

Output:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      },
      "Action": [
        "codeartifact:DescribePackageVersion",
        "codeartifact:DescribeRepository",
        "codeartifact:GetPackageVersionReadme",
        "codeartifact:GetRepositoryEndpoint",
        "codeartifact:ListPackages",
        "codeartifact:ListPackageVersions",
        "codeartifact:ListPackageVersionAssets",
        "codeartifact:ListPackageVersionDependencies",
        "codeartifact:ReadFromRepository"
      ],
      "Resource": "*"
    }
  ]
}
```

Untuk informasi [selengkapnya, lihat Membaca kebijakan](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [GetRepositoryPermissionsPolicy](#) di Referensi AWS CLI Perintah.

## list-domains

Contoh kode berikut menunjukkan cara menggunakan `list-domains`.

### AWS CLI

Untuk daftar domain

`list-domains` Contoh berikut mengembalikan ringkasan semua domain yang dimiliki oleh AWS akun yang membuat panggilan.

```
aws codeartifact list-domains
```

Output:

```
{
  "domains": [
    {
      "name": "my-domain",
      "owner": "111122223333",
      "status": "Active",
      "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    {
      "name": "test-domain",
      "owner": "111122223333",
      "status": "Active",
      "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan domain CodeArtifact di Panduan AWS CodeArtifact Pengguna](#).

- Untuk detail API, lihat [ListDomains](#) di Referensi AWS CLI Perintah.

## list-package-version-assets

Contoh kode berikut menunjukkan cara menggunakan `list-package-version-assets`.

### AWS CLI

Untuk melihat aset versi paket

`list-package-version-assets` Contoh berikut mengambil aset untuk versi 4.0.0 dari paket npm bernama `test-package`.

```
aws codeartifact list-package-version-assets \
  --domain test-domain \
  --repo test-repo \
```

```
--format npm \  
--package test-package \  
--package-version 4.0.0
```

## Output:

```
{  
  "format": "npm",  
  "package": "test-package",  
  "version": "4.0.0",  
  "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+Fi1Ps=",  
  "assets": [  
    {  
      "name": "package.tgz",  
      "size": 316680,  
      "hashes": {  
        "MD5": "60078ec6d9e76b89fb55c860832742b2",  
        "SHA-1": "b44a9b6297bcb698f1c51a3545a2b3b368d59c52",  
        "SHA-256":  
"d2aa8c6afc3c8591765785a37d1c5acae482a8eb3ab9729ed28922692454f2e2",  
        "SHA-512":  
"3e585d15c8a594e20d7de57b362ea81754c011acb2641a19f1b72c8531ea39825896bab344ae616a0a5a824cb9"  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar aset versi paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [ListPackageVersionAssets](#) di Referensi AWS CLI Perintah.

## list-package-version-dependencies

Contoh kode berikut menunjukkan cara menggunakan `list-package-version-dependencies`.

### AWS CLI

Untuk melihat dependensi versi paket

`list-package-version-dependencies` Contoh berikut mengambil dependensi untuk versi 4.0.0 dari paket npm bernama test-package.

```
aws codeartifact list-package-version-dependencies \  
  --domain test-domain \  
  --repo test-repo \  
  --format npm \  
  --package test-package \  
  --package-version 4.0.0
```

Output:

```
{  
  "format": "npm",  
  "package": "test-package",  
  "version": "4.0.0",  
  "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",  
  "dependencies": [  
    {  
      "namespace": "testns",  
      "package": "testdep1",  
      "dependencyType": "regular",  
      "versionRequirement": "1.8.5"  
    },  
    {  
      "namespace": "testns",  
      "package": "testdep2",  
      "dependencyType": "regular",  
      "versionRequirement": "1.8.5"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, [lihat Melihat dan memperbarui detail dan dependensi versi paket di AWS CodeArtifact Panduan Pengguna](#).

- Untuk detail API, lihat [ListPackageVersionDependencies](#) di Referensi AWS CLI Perintah.

## list-package-versions

Contoh kode berikut menunjukkan cara menggunakan `list-package-versions`.

### AWS CLI

Untuk daftar versi paket untuk sebuah paket

`list-package-versions` Contoh berikut mengembalikan daftar versi paket untuk paket bernama `kind-of`.

```
aws codeartifact list-package-versions \  
  --package kind-of \  
  --domain test-domain \  
  --repository test-repo \  
  --format npm
```

Output:

```
{  
  "defaultDisplayVersion": "1.0.1",  
  "format": "npm",  
  "package": "kind-of",  
  "versions": [  
    {  
      "version": "1.0.1",  
      "revision": "REVISION-SAMPLE-1-C7F4S5E9B772FC",  
      "status": "Published"  
    },  
    {  
      "version": "1.0.0",  
      "revision": "REVISION-SAMPLE-2-C752BEEF6D2CFC",  
      "status": "Published"  
    },  
    {  
      "version": "0.1.2",  
      "revision": "REVISION-SAMPLE-3-654S65A5C5E1FC",  
      "status": "Published"  
    },  
    {  
      "version": "0.1.1",  
      "revision": "REVISION-SAMPLE-1-C7F4S5E9B772FC",  
      "status": "Published"  
    },  
    {  
      "version": "0.1.0",  
      "revision": "REVISION-SAMPLE-4-AF669139B772FC",  
      "status": "Published"  
    }  
  ]  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Daftar versi paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [ListPackageVersions](#) di Referensi AWS CLI Perintah.

## list-packages

Contoh kode berikut menunjukkan cara menggunakan `list-packages`.

### AWS CLI

Untuk daftar paket dalam repositori

Berikut `list-packages` contoh daftar paket dalam repositori bernama `test-repo` dalam domain bernama `test-domain`

```
aws codeartifact list-packages \  
  --domain test-domain \  
  --repository test-repo
```

Output:

```
{  
  "packages": [  
    {  
      "format": "npm",  
      "package": "lodash"  
    },  
    {  
      "format": "python",  
      "package": "test-package"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar nama paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [ListPackages](#) di Referensi AWS CLI Perintah.



## list-repositories-in-domain

Contoh kode berikut menunjukkan cara menggunakan `list-repositories-in-domain`.

### AWS CLI

Untuk daftar repositori dalam domain

`list-repositories-in-domain` Contoh berikut mengembalikan ringkasan semua repositori dalam domain `test-domain`.

```
aws codeartifact list-repositories-in-domain \  
  --domain test-domain
```

Output:

```
{  
  "repositories": [  
    {  
      "name": "test-repo",  
      "administratorAccount": "111122223333",  
      "domainName": "test-domain",  
      "domainOwner": "111122223333",  
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-  
domain/test-repo",  
      "description": "This is a test repository."  
    },  
    {  
      "name": "test-repo2",  
      "administratorAccount": "111122223333",  
      "domainName": "test-domain",  
      "domainOwner": "111122223333",  
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-  
domain/test-repo2",  
      "description": "This is a test repository."  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar repositori](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [ListRepositoriesInDomain](#) di Referensi AWS CLI Perintah.

## list-repositories

Contoh kode berikut menunjukkan cara menggunakan `list-repositories`.

### AWS CLI

Untuk daftar repositori

`list-repositories` Contoh berikut mengembalikan ringkasan semua repositori dalam domain yang dimiliki oleh AWS akun yang membuat panggilan.

```
aws codeartifact list-repositories
```

Output:

```
{
  "repositories": [
    {
      "name": "npm-store",
      "administratorAccount": "111122223333",
      "domainName": "my-domain",
      "domainOwner": "111122223333",
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/my-domain/npm-store",
      "description": "Provides npm artifacts from npm, Inc."
    },
    {
      "name": "target-repo",
      "administratorAccount": "111122223333",
      "domainName": "my-domain",
      "domainOwner": "111122223333",
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/my-domain/target-repo",
      "description": "test target repo"
    },
    {
      "name": "test-repo2",
      "administratorAccount": "111122223333",
      "domainName": "test-domain",
      "domainOwner": "111122223333",
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo2",
      "description": "This is a test repository."
    }
  ]
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar repositori](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [ListRepositories](#) di Referensi AWS CLI Perintah.

## login

Contoh kode berikut menunjukkan cara menggunakan `login`.

### AWS CLI

Untuk mengkonfigurasi otentikasi ke repositori Anda dengan perintah `login`

`login` Contoh berikut mengkonfigurasi manajer paket npm dengan repositori bernama `test-repo` dalam domain bernama `test-domain`.

```
aws codeartifact login \  
  --domain test-domain \  
  --repository test-repo \  
  --tool npm
```

Output:

```
Successfully configured npm to use AWS CodeArtifact repository https://test-  
domain-111122223333.d.codeartifact.us-west-2.amazonaws.com/npm/test-repo/  
Login expires in 12 hours at 2020-11-12 01:53:16-05:00
```

Untuk informasi selengkapnya, lihat [Memulai AWS CLI](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [Login](#) di Referensi AWS CLI Perintah.

## put-domain-permissions-policy

Contoh kode berikut menunjukkan cara menggunakan `put-domain-permissions-policy`.

### AWS CLI

Untuk melampirkan kebijakan izin ke domain

`put-domain-permissions-policy` Contoh berikut melampirkan kebijakan izin yang didefinisikan dalam file `policy.json` ke domain bernama `test-domain`.

```
aws codeartifact put-domain-permissions-policy \  
  --domain test-domain \  
  --policy-document file://PATH/T0/policy.json
```

Output:

```
{  
  "policy": {  
    "resourceArn": "arn:aws:codeartifact:region-id:111122223333:domain/test-  
domain",  
    "document": "{ ...policy document content...}",  
    "revision": "MQlyyTQRASRU3HB58gBtSDHXG7Q3hvxxxxxxxxx="
```

Untuk informasi selengkapnya, lihat [Menetapkan kebijakan domain](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [PutDomainPermissionsPolicy](#) di Referensi AWS CLI Perintah.

## **put-repository-permissions-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-repository-permissions-policy`.

AWS CLI

Untuk melampirkan kebijakan izin ke repositori

`put-repository-permissions-policy` Contoh berikut melampirkan kebijakan izin yang didefinisikan dalam file `policy.json` ke repositori bernama `test-repo`.

```
aws codeartifact put-repository-permissions-policy \  
  --domain test-domain \  
  --repository test-repo \  
  --policy-document file://PATH/T0/policy.json
```

Output:

```
{
```

```

    "policy": {
      "resourceArn": "arn:aws:codeartifact:region-id:111122223333:repository/test-
domain/test-repo",
      "document": "{ ...policy document content...}",
      "revision": "MQ1yyTQRASRU3HB58gBtSDHXG7Q3hvxxxxxxxxx="
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Menetapkan kebijakan](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [PutRepositoryPermissionsPolicy](#) di Referensi AWS CLI Perintah.

## update-package-versions-status

Contoh kode berikut menunjukkan cara menggunakan `update-package-versions-status`.

### AWS CLI

Untuk memperbarui status versi paket

`update-package-versions-status` Contoh berikut memperbarui status versi 4.0.0 dari paket paket uji ke Diarsipkan.

```

aws codeartifact update-package-versions-status \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --versions 4.0.0 \
  --target-status Archived

```

Output:

```

{
  "successfulVersions": {
    "4.0.0": {
      "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
      "status": "Archived"
    }
  },
  "failedVersions": {}
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Memperbarui status versi paket](#) di Panduan AWS CodeArtifact Pengguna.

- Untuk detail API, lihat [UpdatePackageVersionsStatus](#) di Referensi AWS CLI Perintah.

## update-repository

Contoh kode berikut menunjukkan cara menggunakan `update-repository`.

### AWS CLI

Untuk memperbarui repositori

`update-repository` Contoh berikut memperbarui deskripsi repo bernama `test-repo` dalam domain bernama `test-domain` menjadi “ini adalah deskripsi yang diperbarui”.

```
aws codeartifact update-repository \
  --domain test-domain \
  --repository test-repo \
  --description "this is an updated description"
```

Output:

```
{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo",
    "description": "this is an updated description",
    "upstreams": [],
    "externalConnections": []
  }
}
```

Untuk informasi selengkapnya, [lihat Melihat atau mengubah konfigurasi repositori](#) di AWS CodeArtifact Panduan Pengguna.

- Untuk detail API, lihat [UpdateRepository](#) di Referensi AWS CLI Perintah.

## CodeBuild contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CodeBuild.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-delete-builds**

Contoh kode berikut menunjukkan cara menggunakan `batch-delete-builds`.

AWS CLI

Untuk menghapus build in AWS CodeBuild.

`batch-delete-builds` Contoh berikut menghapus build in CodeBuild dengan ID yang ditentukan.

```
aws codebuild batch-delete-builds --ids my-build-project-one:a1b2c3d4-5678-9012-abcd-11111EXAMPLE my-build-project-two:a1b2c3d4-5678-9012-abcd-22222EXAMPLE
```

Output:

```
{
  "buildsNotDeleted": [
    {
      "id": "arn:aws:codebuild:us-west-2:123456789012:build/my-build-project-one:a1b2c3d4-5678-9012-abcd-11111EXAMPLE",
      "statusCode": "BUILD_IN_PROGRESS"
    }
  ]
}
```

```

    }
  ],
  "buildsDeleted": [
    "arn:aws:codebuild:us-west-2:123456789012:build/my-build-project-
two:a1b2c3d4-5678-9012-abcd-22222EXAMPLE"
  ]
}

```

Untuk informasi selengkapnya, lihat [Menghapus Build \(AWS CLI\)](#) di Panduan Pengguna AWS CodeBuild .

- Untuk detail API, lihat [BatchDeleteBuilds](#) di Referensi AWS CLI Perintah.

## batch-get-build-batches

Contoh kode berikut menunjukkan cara menggunakan `batch-get-build-batches`.

### AWS CLI

Untuk melihat detail build in AWS CodeBuild.

`batch-get-build-batches` Contoh berikut mendapatkan informasi tentang kumpulan build CodeBuild dengan ID yang ditentukan.

```

aws codebuild batch-get-build-batches \
  --ids codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE

```

Output:

```

{
  "buildBatches": [
    {
      "id": "codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
      "arn": "arn:aws:codebuild:us-west-2:123456789012:build-batch/codebuild-
demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
      "startTime": "2020-11-03T21:52:20.775000+00:00",
      "endTime": "2020-11-03T21:56:59.784000+00:00",
      "currentPhase": "SUCCEEDED",
      "buildBatchStatus": "SUCCEEDED",
      "resolvedSourceVersion": "0a6546f68309560d08a310daac92314c4d378f6b",
      "projectName": "codebuild-demo-project",
      "phases": [
        {

```



```
    "phaseType": "SUBMITTED",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:52:20.775000+00:00",
    "endTime": "2020-11-03T21:52:20.976000+00:00",
    "durationInSeconds": 0
  },
  {
    "phaseType": "DOWNLOAD_BATCHSPEC",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:52:20.976000+00:00",
    "endTime": "2020-11-03T21:52:57.401000+00:00",
    "durationInSeconds": 36
  },
  {
    "phaseType": "IN_PROGRESS",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:52:57.401000+00:00",
    "endTime": "2020-11-03T21:56:59.751000+00:00",
    "durationInSeconds": 242
  },
  {
    "phaseType": "COMBINE_ARTIFACTS",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:56:59.751000+00:00",
    "endTime": "2020-11-03T21:56:59.784000+00:00",
    "durationInSeconds": 0
  },
  {
    "phaseType": "SUCCEEDED",
    "startTime": "2020-11-03T21:56:59.784000+00:00"
  }
],
"source": {
  "type": "GITHUB",
  "location": "https://github.com/my-repo/codebuild-demo-project.git",
  "gitCloneDepth": 1,
  "gitSubmodulesConfig": {
    "fetchSubmodules": false
  },
  "reportBuildStatus": false,
  "insecureSsl": false
},
"secondarySources": [],
"secondarySourceVersions": [],
```

```
"artifacts": {
  "location": ""
},
"secondaryArtifacts": [],
"cache": {
  "type": "NO_CACHE"
},
"environment": {
  "type": "LINUX_CONTAINER",
  "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
  "computeType": "BUILD_GENERAL1_SMALL",
  "environmentVariables": [],
  "privilegedMode": false,
  "imagePullCredentialsType": "CODEBUILD"
},
"logConfig": {
  "cloudWatchLogs": {
    "status": "ENABLED"
  },
  "s3Logs": {
    "status": "DISABLED",
    "encryptionDisabled": false
  }
},
"buildTimeoutInMinutes": 60,
"queuedTimeoutInMinutes": 480,
"complete": true,
"initiator": "Strohm",
"encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
"buildBatchNumber": 6,
"buildBatchConfig": {
  "serviceRole": "arn:aws:iam::123456789012:role/service-role/
codebuild-demo-project",
  "restrictions": {
    "maximumBuildsAllowed": 100
  },
  "timeoutInMins": 480
},
"buildGroups": [
  {
    "identifier": "DOWNLOAD_SOURCE",
    "ignoreFailure": false,
    "currentBuildSummary": {
```

```
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:379737d8-bc35-48ec-97fd-776d27545315",
    "requestedOn": "2020-11-03T21:52:21.394000+00:00",
    "buildStatus": "SUCCEEDED",
    "primaryArtifact": {
      "type": "no_artifacts",
      "identifier": "DOWNLOAD_SOURCE"
    },
    "secondaryArtifacts": []
  },
  {
    "identifier": "linux_small",
    "dependsOn": [],
    "ignoreFailure": false,
    "currentBuildSummary": {
      "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:dd785171-ed84-4bb6-8ede-ceeb86e54bdb",
      "requestedOn": "2020-11-03T21:52:57.604000+00:00",
      "buildStatus": "SUCCEEDED",
      "primaryArtifact": {
        "type": "no_artifacts",
        "identifier": "linux_small"
      },
      "secondaryArtifacts": []
    }
  },
  {
    "identifier": "linux_medium",
    "dependsOn": [
      "linux_small"
    ],
    "ignoreFailure": false,
    "currentBuildSummary": {
      "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:97cf7bd4-5313-4786-8243-4aef350a1267",
      "requestedOn": "2020-11-03T21:54:18.474000+00:00",
      "buildStatus": "SUCCEEDED",
      "primaryArtifact": {
        "type": "no_artifacts",
        "identifier": "linux_medium"
      },
      "secondaryArtifacts": []
    }
  }
}
```

```

    },
    {
      "identifier": "linux_large",
      "dependsOn": [
        "linux_medium"
      ],
      "ignoreFailure": false,
      "currentBuildSummary": {
        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:60a194cd-0d03-4337-9db1-d41476a17d27",
        "requestedOn": "2020-11-03T21:55:39.203000+00:00",
        "buildStatus": "SUCCEEDED",
        "primaryArtifact": {
          "type": "no_artifacts",
          "identifier": "linux_large"
        },
        "secondaryArtifacts": []
      }
    }
  ]
},
"buildBatchesNotFound": []
}

```

Untuk informasi selengkapnya, lihat Batch builds in\_\_ AWS CodeBuild di Panduan Pengguna.AWS CodeBuild < <https://docs.aws.amazon.com/codebuild/latest/userguide/batch-build.html>>

- Untuk detail API, lihat [BatchGetBuildBatches](#) di Referensi AWS CLI Perintah.

## batch-get-builds

Contoh kode berikut menunjukkan cara menggunakan `batch-get-builds`.

### AWS CLI

Untuk melihat detail build in AWS CodeBuild.

`batch-get-builds` Contoh berikut mendapatkan informasi tentang build in CodeBuild dengan ID yang ditentukan.

```
aws codebuild batch-get-builds --ids codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE
```

## Output:

```
{
  "buildsNotFound": [],
  "builds": [
    {
      "artifacts": {
        "md5sum": "0e95edf915048a0c22efe6d139fff837",
        "location": "arn:aws:s3:::codepipeline-us-west-2-820783811474/CodeBuild-Python-Pip/BuildArtif/6DJsQa",
        "encryptionDisabled": false,
        "sha256sum":
"cfa0df33a090966a737f64ae4fe498969fdc842a0c9aec540bf93c37ac0d05a2"
      },
      "logs": {
        "cloudWatchLogs": {
          "status": "ENABLED"
        },
        "s3Logs": {
          "status": "DISABLED"
        },
        "streamName": "46472baf-8f6b-43c2-9255-b3b963af2732",
        "groupName": "/aws/codebuild/codebuild-demo-project",
        "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#logEvent:group=/aws/codebuild/codebuild-demo-project;stream=46472baf-8f6b-43c2-9255-b3b963af2732"
      },
      "timeoutInMinutes": 60,
      "environment": {
        "privilegedMode": false,
        "computeType": "BUILD_GENERAL1_MEDIUM",
        "image": "aws/codebuild/windows-base:1.0",
        "environmentVariables": [],
        "type": "WINDOWS_CONTAINER"
      },
      "projectName": "codebuild-demo-project",
      "buildComplete": true,
      "source": {
        "gitCloneDepth": 1,
        "insecureSsl": false,

```

```
    "type": "CODEPIPELINE"
  },
  "buildStatus": "SUCCEEDED",
  "secondaryArtifacts": [],
  "phases": [
    {
      "durationInSeconds": 0,
      "startTime": 1548717462.122,
      "phaseType": "SUBMITTED",
      "endTime": 1548717462.484,
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 0,
      "startTime": 1548717462.484,
      "phaseType": "QUEUED",
      "endTime": 1548717462.775,
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 34,
      "endTime": 1548717496.909,
      "contexts": [
        {
          "statusCode": "",
          "message": ""
        }
      ],
      "startTime": 1548717462.775,
      "phaseType": "PROVISIONING",
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 15,
      "endTime": 1548717512.555,
      "contexts": [
        {
          "statusCode": "",
          "message": ""
        }
      ],
      "startTime": 1548717496.909,
      "phaseType": "DOWNLOAD_SOURCE",
      "phaseStatus": "SUCCEEDED"
    }
  ]
}
```

```
    },
    {
      "durationInSeconds": 0,
      "endTime": 1548717512.734,
      "contexts": [
        {
          "statusCode": "",
          "message": ""
        }
      ],
      "startTime": 1548717512.555,
      "phaseType": "INSTALL",
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 0,
      "endTime": 1548717512.924,
      "contexts": [
        {
          "statusCode": "",
          "message": ""
        }
      ],
      "startTime": 1548717512.734,
      "phaseType": "PRE_BUILD",
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 9,
      "endTime": 1548717522.254,
      "contexts": [
        {
          "statusCode": "",
          "message": ""
        }
      ],
      "startTime": 1548717512.924,
      "phaseType": "BUILD",
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 3,
      "endTime": 1548717525.498,
      "contexts": [
```

```
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548717522.254,
    "phaseType": "POST_BUILD",
    "phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 9,
    "endTime": 1548717534.646,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548717525.498,
    "phaseType": "UPLOAD_ARTIFACTS",
    "phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 2,
    "endTime": 1548717536.846,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548717534.646,
    "phaseType": "FINALIZING",
    "phaseStatus": "SUCCEEDED"
},
{
    "startTime": 1548717536.846,
    "phaseType": "COMPLETED"
}
],
"startTime": 1548717462.122,
"encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
"initiator": "codepipeline/CodeBuild-Pipeline",
"secondarySources": [],
```



```

    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-
codebuild-service-role",
    "currentPhase": "COMPLETED",
    "id": "codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
    "cache": {
        "type": "NO_CACHE"
    },
    "sourceVersion": "arn:aws:s3:::codepipeline-us-west-2-820783811474/
CodeBuild-Python-Pip/SourceArti/1TspnN3.zip",
    "endTime": 1548717536.846,
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/codebuild-demo-
project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
    "queuedTimeoutInMinutes": 480,
    "resolvedSourceVersion": "f2194c1757bbdcb0f8f229254a4b3c8b27d43e0b"
},
{
    "artifacts": {
        "md5sum": "",
        "overrideArtifactName": false,
        "location": "arn:aws:s3:::my-artifacts/codebuild-demo-project",
        "encryptionDisabled": false,
        "sha256sum": ""
    },
    "logs": {
        "cloudWatchLogs": {
            "status": "ENABLED"
        },
        "s3Logs": {
            "status": "DISABLED"
        },
        "streamName": "4dea3ca4-20ec-4898-b22a-a9eb9292775d",
        "groupName": "/aws/codebuild/codebuild-demo-project",
        "deepLink": "https://console.aws.amazon.com/cloudwatch/
home?region=us-west-2#logEvent:group=/aws/codebuild/codebuild-demo-
project;stream=4dea3ca4-20ec-4898-b22a-a9eb9292775d"
    },
    "timeoutInMinutes": 60,
    "environment": {
        "privilegedMode": false,
        "computeType": "BUILD_GENERAL1_MEDIUM",
        "image": "aws/codebuild/windows-base:1.0",
        "environmentVariables": [],
        "type": "WINDOWS_CONTAINER"
    },
},

```

```
"projectName": "codebuild-demo-project",
"buildComplete": true,
"source": {
  "gitCloneDepth": 1,
  "location": "https://github.com/my-repo/codebuild-demo-project.git",
  "insecureSsl": false,
  "reportBuildStatus": false,
  "type": "GITHUB"
},
"buildStatus": "SUCCEEDED",
"secondaryArtifacts": [],
"phases": [
  {
    "durationInSeconds": 0,
    "startTime": 1548716241.89,
    "phaseType": "SUBMITTED",
    "endTime": 1548716242.241,
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 0,
    "startTime": 1548716242.241,
    "phaseType": "QUEUED",
    "endTime": 1548716242.536,
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 33,
    "endTime": 1548716276.171,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716242.536,
    "phaseType": "PROVISIONING",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 15,
    "endTime": 1548716291.809,
    "contexts": [
      {
```

```
        "statusCode": "",
        "message": ""
    }
],
"startTime": 1548716276.171,
"phaseType": "DOWNLOAD_SOURCE",
"phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 0,
    "endTime": 1548716291.993,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548716291.809,
    "phaseType": "INSTALL",
    "phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 0,
    "endTime": 1548716292.191,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548716291.993,
    "phaseType": "PRE_BUILD",
    "phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 9,
    "endTime": 1548716301.622,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548716292.191,
```

```
    "phaseType": "BUILD",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 3,
    "endTime": 1548716304.783,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716301.622,
    "phaseType": "POST_BUILD",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 8,
    "endTime": 1548716313.775,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716304.783,
    "phaseType": "UPLOAD_ARTIFACTS",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 2,
    "endTime": 1548716315.935,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716313.775,
    "phaseType": "FINALIZING",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "startTime": 1548716315.935,
```

```

        "phaseType": "COMPLETED"
      }
    ],
    "startTime": 1548716241.89,
    "secondarySourceVersions": [],
    "initiator": "my-codebuild-project",
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/codebuild-demo-
project:815e755f-bade-4a7e-80f0-efe51EXAMPLE",
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-
codebuild-service-role",
    "currentPhase": "COMPLETED",
    "id": "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE",
    "cache": {
      "type": "NO_CACHE"
    },
    "endTime": 1548716315.935,
    "secondarySources": [],
    "queuedTimeoutInMinutes": 480,
    "resolvedSourceVersion": "f2194c1757bbdcb0f8f229254a4b3c8b27d43e0b"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat Detail Build \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [BatchGetBuilds](#) di Referensi AWS CLI Perintah.

## batch-get-projects

Contoh kode berikut menunjukkan cara menggunakan `batch-get-projects`.

### AWS CLI

Untuk mendapatkan daftar nama proyek AWS CodeBuild build.

`batch-get-projects` Contoh berikut mendapatkan daftar proyek CodeBuild build yang ditentukan oleh nama.

```
aws codebuild batch-get-projects --names codebuild-demo-project codebuild-demo-
project2 my-other-demo-project
```

Pada output berikut, `projectsNotFound` array mencantumkan nama proyek build apa pun yang ditentukan, tetapi tidak ditemukan. `projectsArray` mencantumkan detail untuk setiap proyek build tempat informasi ditemukan.

```
{
  "projectsNotFound": [],
  "projects": [
    {
      "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
      "name": "codebuild-demo-project2",
      "queuedTimeoutInMinutes": 480,
      "timeoutInMinutes": 60,
      "source": {
        "buildspec": "version: 0.2\n\n#env:\n #variables:\n # key:\n\n# key: \"value\"\n # key: \"value\"\n #parameter-store:\n # key: \"value\"\n\n# key: \"value\"\n\n#phases:\n #install:\n #commands:\n # - command\n\n# - command\n #pre_build:\n #commands:\n # - command\n # - command\n\n# build:\n #commands:\n # - command\n # - command\n\n# post_build:\n\n#commands:\n # - command\n # - command\n\n#artifacts:\n #files:\n #\n - location\n # - location\n\n#name: $(date +%Y-%m-%d)\n #discard-paths: yes\n\n#base-directory: location\n\n#cache:\n #paths:\n # - paths",
        "type": "NO_SOURCE",
        "insecureSsl": false,
        "gitCloneDepth": 1
      },
      "artifacts": {
        "type": "NO_ARTIFACTS"
      },
      "badge": {
        "badgeEnabled": false
      },
      "lastModified": 1540588091.108,
      "created": 1540588091.108,
      "arn": "arn:aws:codebuild:us-west-2:123456789012:project/test-for-sample",
      "secondarySources": [],
      "secondaryArtifacts": [],
      "cache": {
        "type": "NO_CACHE"
      },
      "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-test-role",
      "environment": {
```

```
        "image": "aws/codebuild/java:openjdk-8",
        "privilegedMode": true,
        "type": "LINUX_CONTAINER",
        "computeType": "BUILD_GENERAL1_SMALL",
        "environmentVariables": []
    },
    "tags": []
},
{
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "name": "my-other-demo-project",
    "queuedTimeoutInMinutes": 480,
    "timeoutInMinutes": 60,
    "source": {
        "location": "https://github.com/iversonic/codedeploy-sample.git",
        "reportBuildStatus": false,
        "buildspec": "buildspec.yml",
        "insecureSsl": false,
        "gitCloneDepth": 1,
        "type": "GITHUB",
        "auth": {
            "type": "OAUTH"
        }
    },
    "artifacts": {
        "type": "NO_ARTIFACTS"
    },
    "badge": {
        "badgeEnabled": false
    },
    "lastModified": 1523401711.73,
    "created": 1523401711.73,
    "arn": "arn:aws:codebuild:us-west-2:123456789012:project/Project2",
    "cache": {
        "type": "NO_CACHE"
    },
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/codebuild-Project2-service-role",
    "environment": {
        "image": "aws/codebuild/nodejs:4.4.7",
        "privilegedMode": false,
        "type": "LINUX_CONTAINER",
        "computeType": "BUILD_GENERAL1_SMALL",
        "environmentVariables": []
    }
}
```

```

    },
    "tags": []
  }
]
}

```

Untuk informasi selengkapnya, [lihat Melihat Detail Proyek Pembuatan \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [BatchGetProjects](#) di Referensi AWS CLI Perintah.

## batch-get-report-groups

Contoh kode berikut menunjukkan cara menggunakan `batch-get-report-groups`.

### AWS CLI

Untuk mendapatkan informasi tentang satu atau lebih grup laporan di AWS CodeBuild.

`batch-get-report-groups` Contoh berikut mengambil informasi tentang kelompok laporan dengan ARN yang ditentukan.

```

aws codebuild batch-get-report-groups \
  --report-group-arns arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
  <report-group-name>

```

### Output:

```

{
  "reportGroups": [
    {
      "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-
group-name>",
      "name": "report-group-name",
      "type": "TEST",
      "exportConfig": {
        "exportConfigType": "NO_EXPORT"
      },
      "created": "2020-10-01T18:04:08.466000+00:00",
      "lastModified": "2020-10-01T18:04:08.466000+00:00",
      "tags": []
    }
  ],
}

```



```
"reportGroupsNotFound": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [BatchGetReportGroups](#) di Referensi AWS CLI Perintah.

## batch-get-reports

Contoh kode berikut menunjukkan cara menggunakan `batch-get-reports`.

### AWS CLI

Untuk mendapatkan informasi tentang satu atau lebih laporan di AWS CodeBuild.

`batch-get-reports` Contoh berikut mengambil informasi tentang laporan dengan ARN tertentu.

```
aws codebuild batch-get-reports \
  --report-arns arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 1 ID> arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 2 ID>
```

### Output:

```
{
  "reports": [
    {
      "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 1 ID>",
      "type": "TEST",
      "name": "<report-group-name>",
      "reportGroupArn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>",
      "executionId": "arn:aws:codebuild:<region-ID>:<user-ID>:build/test-
reports:<ID>",
      "status": "FAILED",
      "created": "2020-10-01T11:25:22.531000-07:00",
      "expired": "2020-10-31T11:25:22-07:00",
      "exportConfig": {
        "exportConfigType": "NO_EXPORT"
      },
      "truncated": false,
    }
  ]
}
```

```

        "testSummary": {
            "total": 28,
            "statusCounts": {
                "ERROR": 5,
                "FAILED": 1,
                "SKIPPED": 4,
                "SUCCEEDED": 18,
                "UNKNOWN": 0
            },
            "durationInNanoSeconds": 94000000
        }
    },
    {
        "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 2 ID>",
        "type": "TEST",
        "name": "<report-group-name>",
        "reportGroupArn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>",
        "executionId": "arn:aws:codebuild:<region-ID>:<user-ID>:build/test-
reports:<ID>",
        "status": "FAILED",
        "created": "2020-10-01T11:13:05.816000-07:00",
        "expired": "2020-10-31T11:13:05-07:00",
        "exportConfig": {
            "exportConfigType": "NO_EXPORT"
        },
        "truncated": false,
        "testSummary": {
            "total": 28,
            "statusCounts": {
                "ERROR": 5,
                "FAILED": 1,
                "SKIPPED": 4,
                "SUCCEEDED": 18,
                "UNKNOWN": 0
            },
            "durationInNanoSeconds": 94000000
        }
    }
],
"reportsNotFound": []
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [BatchGetReports](#) di Referensi AWS CLI Perintah.

## create-project

Contoh kode berikut menunjukkan cara menggunakan `create-project`.

### AWS CLI

Contoh 1: Untuk membuat proyek AWS CodeBuild build

`create-project` Contoh berikut membuat proyek CodeBuild build menggunakan file sumber dari bucket S3

```
aws codebuild create-project \
  --name "my-demo-project" \
  --source "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-input-bucket/my-source.zip\"}" \
  --artifacts "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-output-bucket\"}" \
  --environment "{\"type\": \"LINUX_CONTAINER\", \"image\": \"aws/codebuild/standard:1.0\", \"computeType\": \"BUILD_GENERAL1_SMALL\"}" \
  --service-role "arn:aws:iam::123456789012:role/service-role/my-codebuild-service-role"
```

Output:

```
{
  "project": {
    "arn": "arn:aws:codebuild:us-west-2:123456789012:project/my-demo-project",
    "name": "my-cli-demo-project",
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-service-role",
    "lastModified": 1556839783.274,
    "badge": {
      "badgeEnabled": false
    },
    "queuedTimeoutInMinutes": 480,
    "environment": {
      "image": "aws/codebuild/standard:1.0",
```

```

        "computeType": "BUILD_GENERAL1_SMALL",
        "type": "LINUX_CONTAINER",
        "imagePullCredentialsType": "CODEBUILD",
        "privilegedMode": false,
        "environmentVariables": []
    },
    "artifacts": {
        "location": "codebuild-us-west-2-123456789012-output-bucket",
        "name": "my-cli-demo-project",
        "namespaceType": "NONE",
        "type": "S3",
        "packaging": "NONE",
        "encryptionDisabled": false
    },
    "source": {
        "type": "S3",
        "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip",
        "insecureSsl": false
    },
    "timeoutInMinutes": 60,
    "cache": {
        "type": "NO_CACHE"
    },
    "created": 1556839783.274
}
}

```

Contoh 2: Untuk membuat proyek AWS CodeBuild build menggunakan file input JSON untuk parameter

`create-project` Contoh berikut membuat proyek CodeBuild build dengan meneruskan semua parameter yang diperlukan dalam file input JSON. Buat template file input dengan menjalankan perintah hanya dengan file `--generate-cli-skeleton` parameter.

```
aws codebuild create-project --cli-input-json file://create-project.json
```

File JSON masukan `create-project.json` berisi konten berikut:

```
{
  "name": "codebuild-demo-project",
  "source": {
```

```
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
  },
  "artifacts": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-output-bucket"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:1.0",
    "computeType": "BUILD_GENERAL1_SMALL"
  },
  "serviceRole": "serviceIAMRole"
}
```

### Output:

```
{
  "project": {
    "name": "codebuild-demo-project",
    "serviceRole": "serviceIAMRole",
    "tags": [],
    "artifacts": {
      "packaging": "NONE",
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-output-bucket",
      "name": "message-util.zip"
    },
    "lastModified": 1472661575.244,
    "timeoutInMinutes": 60,
    "created": 1472661575.244,
    "environment": {
      "computeType": "BUILD_GENERAL1_SMALL",
      "image": "aws/codebuild/standard:1.0",
      "type": "LINUX_CONTAINER",
      "environmentVariables": []
    },
    "source": {
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-input-bucket/
MessageUtil.zip"
    },
    "encryptionKey": "arn:aws:kms:region-ID:account-ID:alias/aws/s3",
  }
}
```

```
"arn": "arn:aws:codebuild:region-ID:account-ID:project/codebuild-demo-  
project"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Proyek Build \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [CreateProject](#) di Referensi AWS CLI Perintah.

## create-report-group

Contoh kode berikut menunjukkan cara menggunakan `create-report-group`.

### AWS CLI

Untuk membuat grup laporan di AWS CodeBuild.

`create-report-group` Contoh berikut membuat grup laporan baru.

```
aws codebuild create-report-group \  
  --cli-input-json file://create-report-group-source.json
```

Isi dari `create-report-group-source.json`:

```
{  
  "name": "cli-created-report-group",  
  "type": "TEST",  
  "exportConfig": {  
    "exportConfigType": "S3",  
    "s3Destination": {  
      "bucket": "my-s3-bucket",  
      "path": "",  
      "packaging": "ZIP",  
      "encryptionDisabled": true  
    }  
  }  
}
```

Output:

```
{
```

```

    "reportGroup": {
      "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-
report-group",
      "name": "cli-created-report-group",
      "type": "TEST",
      "exportConfig": {
        "exportConfigType": "S3",
        "s3Destination": {
          "bucket": "my-s3-bucket",
          "path": "",
          "packaging": "ZIP",
          "encryptionDisabled": true
        }
      },
      "created": 1602020026.775,
      "lastModified": 1602020026.775
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [CreateReportGroup](#) di Referensi AWS CLI Perintah.

## create-webhook

Contoh kode berikut menunjukkan cara menggunakan `create-webhook`.

### AWS CLI

Untuk membuat filter webhook untuk sebuah proyek AWS CodeBuild

`create-webhook` Contoh berikut membuat webhook untuk CodeBuild proyek bernama `my-project` yang memiliki dua grup filter. Grup filter pertama menentukan permintaan tarik yang dibuat, diperbarui, atau dibuka kembali pada cabang dengan nama referensi Git yang cocok dengan ekspresi reguler `^refs/heads/master$` dan referensi kepala yang cocok. `^refs/heads/myBranch$` Grup filter kedua menentukan permintaan push pada cabang dengan nama referensi Git yang tidak cocok dengan ekspresi `^refs/heads/myBranch$` reguler.

```

aws codebuild create-webhook \
  --project-name my-project \

```

```
--filter-groups "[[{\\"type\\":\\"EVENT\\",\\"pattern\\":\\"PULL_REQUEST_CREATED,
PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED\\"},{\\"type\\":\\"HEAD_REF\\",\\"pattern
\\":\\"^refs/heads/myBranch$\\",\\"excludeMatchedPattern\\":true},{\\"type\\":\\"BASE_REF
\\",\\"pattern\\":\\"^refs/heads/master$\\",\\"excludeMatchedPattern\\":true}],[{\\"type\\":
\\"EVENT\\",\\"pattern\\":\\"PUSH\\"},{\\"type\\":\\"HEAD_REF\\",\\"pattern\\":\\"^refs/heads/
myBranch$\\",\\"excludeMatchedPattern\\":true}]]]"
```

## Output:

```
{
  "webhook": {
    "payloadUrl": "https://codebuild.us-west-2.amazonaws.com/webhooks?
t=eyJlbnNyeXB0ZWREYXRhIjoiVVl5MGtoeGRwSzZFRXl2Wnh4bld1Z0tKZ291TVpQNEtFamQ3RDlDYWpRaGIreVFrdm
"url": "https://api.github.com/repos/iversonic/codedeploy-sample/
hooks/105190656",
    "lastModifiedSecret": 1556311319.069,
    "filterGroups": [
      [
        {
          "type": "EVENT",
          "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
PULL_REQUEST_REOPENED",
          "excludeMatchedPattern": false
        },
        {
          "type": "HEAD_REF",
          "pattern": "refs/heads/myBranch$",
          "excludeMatchedPattern": true
        },
        {
          "type": "BASE_REF",
          "pattern": "refs/heads/master$",
          "excludeMatchedPattern": true
        }
      ],
      [
        {
          "type": "EVENT",
          "pattern": "PUSH",
          "excludeMatchedPattern": false
        },
        {
          "type": "HEAD_REF",
```



```

        "pattern": "refs/heads/myBranch$",
        "excludeMatchedPattern": true
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Memfilter Acara GitHub Webhook \(SDK\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [CreateWebhook](#) di Referensi AWS CLI Perintah.

## delete-build-batch

Contoh kode berikut menunjukkan cara menggunakan `delete-build-batch`.

### AWS CLI

Untuk menghapus batch build in AWS CodeBuild.

`delete-build-batch` Contoh berikut menghapus build batch yang ditentukan.

```
aws codebuild delete-build-batch \
  --id <project-name>:<batch-ID>
```

### Output:

```
{
  "statusCode": "BATCH_DELETED",
  "buildsDeleted": [
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>",
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>",
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>",
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>"
  ],
  "buildsNotDeleted": []
}
```

Untuk informasi selengkapnya, lihat [Batch build AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [DeleteBuildBatch](#) di Referensi AWS CLI Perintah.

## **delete-project**

Contoh kode berikut menunjukkan cara menggunakan `delete-project`.

### AWS CLI

Untuk menghapus proyek AWS CodeBuild build

`delete-project` Contoh berikut menghapus proyek CodeBuild build yang ditentukan.

```
aws codebuild delete-project --name my-project
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Proyek Pembuatan \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [DeleteProject](#) di Referensi AWS CLI Perintah.

## **delete-report-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-report-group`.

### AWS CLI

Untuk menghapus grup laporan di AWS CodeBuild.

`delete-report-group` Contoh berikut menghapus grup laporan dengan ARN yang ditentukan.

```
aws codebuild delete-report-group \  
  --arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-group-name>
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [DeleteReportGroup](#) di Referensi AWS CLI Perintah.

## delete-report

Contoh kode berikut menunjukkan cara menggunakan `delete-report`.

### AWS CLI

Untuk menghapus laporan di AWS CodeBuild.

`delete-report` Contoh berikut menghapus laporan yang ditentukan.

```
aws codebuild delete-report \  
  --arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-  
name>:<report-ID>
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [DeleteReport](#) di Referensi AWS CLI Perintah.

## delete-source-credentials

Contoh kode berikut menunjukkan cara menggunakan `delete-source-credentials`.

### AWS CLI

Untuk memutuskan sambungan dari penyedia sumber dan menghapus token aksesnya.

`delete-source-credentials` Contoh berikut terputus dari penyedia sumber dan menghapus tokennya. ARN kredensial sumber yang digunakan untuk terhubung ke penyedia sumber menentukan kredensial sumber mana.

```
aws codebuild delete-source-credentials --arn arn-of-your-credentials
```

Output:

```
{  
  "arn": "arn:aws:codebuild:your-region:your-account-id:token/your-server-type"
```

```
}
```

Untuk informasi selengkapnya, lihat [Connect Source Providers with Access Tokens \(CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [DeleteSourceCredentials](#) di Referensi AWS CLI Perintah.

## **delete-webhook**

Contoh kode berikut menunjukkan cara menggunakan `delete-webhook`.

### AWS CLI

Untuk menghapus filter webhook dari proyek AWS CodeBuild

`delete-webhook` Contoh berikut menghapus webhook dari proyek yang ditentukan CodeBuild .

```
aws codebuild delete-webhook --project-name my-project
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Stop Running Builds Automatically \(AWS CLI\)](#) di Panduan Pengguna AWS CodeBuild .

- Untuk detail API, lihat [DeleteWebhook](#) di Referensi AWS CLI Perintah.

## **describe-code-coverages**

Contoh kode berikut menunjukkan cara menggunakan `describe-code-coverages`.

### AWS CLI

Untuk mendapatkan informasi rinci tentang hasil tes cakupan kode di AWS CodeBuild.

`describe-code-coverages` Contoh berikut mendapatkan informasi tentang hasil uji cakupan kode dalam laporan yang ditentukan.

```
aws codebuild describe-code-coverages \  
  --report-arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-  
name>:<report-ID>
```

Output:

```
{
  "codeCoverages": [
    {
      "id": "20a0adcc-db13-4b66-804b-ecaf9f852855",
      "reportARN": "arn:aws:codebuild:<region-ID>:972506530580:report/<report-
group-name>:<report-ID>",
      "filePath": "<source-file-1-path>",
      "lineCoveragePercentage": 83.33,
      "linesCovered": 5,
      "linesMissed": 1,
      "branchCoveragePercentage": 50.0,
      "branchesCovered": 1,
      "branchesMissed": 1,
      "expired": "2020-11-20T21:22:45+00:00"
    },
    {
      "id": "0887162d-bf57-4cf1-a164-e432373d1a83",
      "reportARN": "arn:aws:codebuild:<region-ID>:972506530580:report/<report-
group-name>:<report-ID>",
      "filePath": "<source-file-2-path>",
      "lineCoveragePercentage": 90.9,
      "linesCovered": 10,
      "linesMissed": 1,
      "branchCoveragePercentage": 50.0,
      "branchesCovered": 1,
      "branchesMissed": 1,
      "expired": "2020-11-20T21:22:45+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Laporan cakupan kode](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [DescribeCodeCoverages](#) di Referensi AWS CLI Perintah.

## describe-test-cases

Contoh kode berikut menunjukkan cara menggunakan `describe-test-cases`.

### AWS CLI

Untuk mendapatkan informasi rinci tentang kasus uji di AWS CodeBuild.

`describe-test-cases` Contoh berikut mendapatkan informasi tentang kasus uji dalam laporan yang ditentukan.

```
aws codebuild describe-test-cases \
  --report-arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>
```

Output:

```
{
  "testCases": [
    {
      "reportArn": "arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>",
      "testRawDataPath": "<test-report-path>",
      "prefix": "NUnit.Tests.Assemblies.MockTestFixture",
      "name": "NUnit.Tests.Assemblies.MockTestFixture.NotRunnableTest",
      "status": "ERROR",
      "durationInNanoSeconds": 0,
      "message": "No arguments were provided\n",
      "expired": "2020-11-20T17:52:10+00:00"
    },
    {
      "reportArn": "arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>",
      "testRawDataPath": "<test-report-path>",
      "prefix": "NUnit.Tests.Assemblies.MockTestFixture",
      "name": "NUnit.Tests.Assemblies.MockTestFixture.TestWithException",
      "status": "ERROR",
      "durationInNanoSeconds": 0,
      "message": "System.ApplicationException : Intentional Exception
\nat NUnit.Tests.Assemblies.MockTestFixture.MethodThrowsException()\nat
NUnit.Tests.Assemblies.MockTestFixture.TestWithException()\n\n",
      "expired": "2020-11-20T17:52:10+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan pelaporan pengujian AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [DescribeTestCases](#) di Referensi AWS CLI Perintah.

## **import-source-credentials**

Contoh kode berikut menunjukkan cara menggunakan `import-source-credentials`.

### AWS CLI

Connect AWS CodeBuild pengguna ke penyedia sumber dengan mengimpor kredensi untuk penyedia sumber.

`import-source-credentials` Contoh berikut mengimpor token untuk repositori Bitbucket yang menggunakan BASIC\_AUTH untuk jenis otentikasi.

```
aws codebuild import-source-credentials --server-type BITBUCKET --auth-type
BASIC_AUTH --token my-Bitbucket-password --username my-Bitbucket-username
```

Output:

```
{
  "arn": "arn:aws:codebuild:us-west-2:123456789012:token/bitbucket"
}
```

Untuk informasi selengkapnya, lihat [Connect Source Providers with Access Tokens \(CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [ImportSourceCredentials](#) di Referensi AWS CLI Perintah.

## **invalidate-project-cache**

Contoh kode berikut menunjukkan cara menggunakan `invalidate-project-cache`.

### AWS CLI

Untuk mengatur ulang cache untuk proyek AWS CodeBuild build.

`invalidate-project-cache` Contoh berikut me-reset cache untuk CodeBuild proyek yang ditentukan.

```
aws codebuild invalidate-project-cache --project-name my-project
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membangun Caching CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [InvalidateProjectCache](#) di Referensi AWS CLI Perintah.

## **list-build-batches-for-project**

Contoh kode berikut menunjukkan cara menggunakan `list-build-batches-for-project`.

### AWS CLI

Untuk membuat daftar build batch untuk proyek build tertentu di AWS CodeBuild.

`list-build-batches-for-project` Contoh berikut mencantumkan build CodeBuild batch untuk proyek yang ditentukan.

```
aws codebuild list-build-batches-for-project \  
  --project-name "<project-name>"
```

Output:

```
{  
  "ids": [  
    "<project-name>:<batch-ID>",  
    "<project-name>:<batch-ID>"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Batch build AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [ListBuildBatchesForProject](#) di Referensi AWS CLI Perintah.

## **list-build-batches**

Contoh kode berikut menunjukkan cara menggunakan `list-build-batches`.

### AWS CLI

Untuk membuat daftar batch build di AWS CodeBuild.

`list-build-batches` Contoh berikut mencantumkan build CodeBuild batch untuk akun saat ini.



```
aws codebuild list-build-batches
```

Output:

```
{
  "ids": [
    "<project-name>:<batch-ID>",
    "<project-name>:<batch-ID>"
  ]
}
```

Untuk informasi selengkapnya, lihat [Batch builds in\\_\\_ AWS CodeBuild di Panduan Pengguna.AWS CodeBuild < https://docs.aws.amazon.com/codebuild/latest/userguide/batch-build.html>](https://docs.aws.amazon.com/codebuild/latest/userguide/batch-build.html)

- Untuk detail API, lihat [ListBuildBatches](#) di Referensi AWS CLI Perintah.

## list-builds-for-project

Contoh kode berikut menunjukkan cara menggunakan `list-builds-for-project`.

AWS CLI

Untuk melihat daftar build untuk proyek AWS CodeBuild build.

`list-builds-for-project` Contoh berikut mencantumkan ID build dalam urutan menurun untuk proyek CodeBuild build yang ditentukan.

```
aws codebuild list-builds-for-project --project-name codebuild-demo-project --sort-order DESCENDING
```

Output:

```
{
  "ids": [
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-11111example",
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-22222example",
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-33333example",
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-44444example",
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-55555example"
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Melihat Daftar ID Build untuk Build Project \(AWS CLI\)](#) di Panduan Pengguna AWS CodeBuild

- Untuk detail API, lihat [ListBuildsForProject](#) di Referensi AWS CLI Perintah.

## list-builds

Contoh kode berikut menunjukkan cara menggunakan `list-builds`.

### AWS CLI

Untuk mendapatkan daftar ID AWS CodeBuild build.

`list-builds` Contoh berikut mendapatkan daftar CodeBuild ID diurutkan dalam urutan menaik.

```
aws codebuild list-builds --sort-order ASCENDING
```

Output mencakup `nextToken` nilai yang menunjukkan bahwa ada lebih banyak output yang tersedia.

```
{
  "nextToken": "4AEA6u7J...The full token has been omitted for
brevity...MzY20A==",
  "ids": [
    "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE"
    "codebuild-demo-project:84a7f3d1-d40e-4956-b4cf-7a9d4EXAMPLE"
    ... The full list of build IDs has been omitted for brevity ...
    "codebuild-demo-project:931d0b72-bf6f-4040-a472-5c707EXAMPLE"
  ]
}
```

Jalankan perintah ini lagi dan berikan `nextToken` nilai dalam respons sebelumnya sebagai parameter untuk mendapatkan bagian berikutnya dari output. Ulangi sampai Anda tidak menerima `nextToken` nilai dalam respons.

```
aws codebuild list-builds --sort-order ASCENDING --next-token 4AEA6u7J...The full
token has been omitted for brevity...MzY20A==
```

Bagian selanjutnya dari output:

```
{
  "ids": [
    "codebuild-demo-project:49015049-21cf-4b50-9708-df115EXAMPLE",
    "codebuild-demo-project:543e7206-68a3-46d6-a4da-759abEXAMPLE",
    ... The full list of build IDs has been omitted for brevity ...
    "codebuild-demo-project:c282f198-4582-4b38-bdc0-26f96EXAMPLE"
  ]
}
```

Untuk informasi selengkapnya, [lihat Melihat Daftar ID Build \(AWS CLI\)](#) di Panduan Pengguna AWS CodeBuild

- Untuk detail API, lihat [ListBuilds](#) di Referensi AWS CLI Perintah.

## list-curated-environment-images

Contoh kode berikut menunjukkan cara menggunakan `list-curated-environment-images`.

### AWS CLI

Untuk mendapatkan daftar gambar Docker AWS CodeBuild yang dikelola olehnya, Anda dapat menggunakannya untuk build Anda.

`list-curated-environment-images` Contoh berikut mencantumkan gambar Docker yang dikelola oleh CodeBuild yang dapat digunakan untuk build. :

```
aws codebuild list-curated-environment-images
```

Output:

```
{
  "platforms": [
    {
      "platform": "AMAZON_LINUX",
      "languages": [
        {
          "language": "JAVA",
          "images": [
            {
```

```
        "description": "AWS ElasticBeanstalk - Java 7 Running on
Amazon Linux 64bit v2.1.3",
        "name": "aws/codebuild/eb-java-7-amazonlinux-64:2.1.3",
        "versions": [
            "aws/codebuild/eb-java-7-amazonlinux-64:2.1.3-1.0.0"
        ]
    },
    {
        "description": "AWS ElasticBeanstalk - Java 8 Running on
Amazon Linux 64bit v2.1.3",
        "name": "aws/codebuild/eb-java-8-amazonlinux-64:2.1.3",
        "versions": [
            "aws/codebuild/eb-java-8-amazonlinux-64:2.1.3-1.0.0"
        ]
    },
    ... LIST TRUNCATED FOR BREVITY ...
]
}
```

Untuk informasi selengkapnya, lihat [Gambar Docker Disediakan oleh CodeBuild](#) di AWS CodeBuild Panduan Pengguna

- Untuk detail API, lihat [ListCuratedEnvironmentImages](#) di Referensi AWS CLI Perintah.

## list-projects

Contoh kode berikut menunjukkan cara menggunakan `list-projects`.

### AWS CLI

Untuk mendapatkan daftar nama proyek AWS CodeBuild build.

`list-projects` Contoh berikut mendapatkan daftar proyek CodeBuild build yang diurutkan berdasarkan nama dalam urutan menaik.

```
aws codebuild list-projects --sort-by NAME --sort-order ASCENDING
```

Output mencakup `nextToken` nilai yang menunjukkan bahwa ada lebih banyak output yang tersedia.

```
{
  "nextToken": "Ci33ACF6...The full token has been omitted for brevity...U
+AkMx8=",
  "projects": [
    "codebuild-demo-project",
    "codebuild-demo-project2",
    ... The full list of build project names has been omitted for
brevity ...
    "codebuild-demo-project99"
  ]
}
```

Jalankan perintah ini lagi dan berikan nextToken nilai dari respons sebelumnya sebagai parameter untuk mendapatkan bagian berikutnya dari output. Ulangi sampai Anda tidak menerima nextToken nilai dalam respons.

```
aws codebuild list-projects --sort-by NAME --sort-order ASCENDING --next-token
Ci33ACF6...The full token has been omitted for brevity...U+AkMx8=

{
  "projects": [
    "codebuild-demo-project100",
    "codebuild-demo-project101",
    ... The full list of build project names has been omitted for
brevity ...
    "codebuild-demo-project122"
  ]
}
```

Untuk informasi selengkapnya, [lihat Melihat Daftar Nama Proyek Build \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [ListProjects](#) di Referensi AWS CLI Perintah.

## list-report-groups

Contoh kode berikut menunjukkan cara menggunakan `list-report-groups`.

### AWS CLI

Untuk mendapatkan daftar ARN grup laporan di AWS CodeBuild.

`list-report-groups` Contoh berikut mengambil ARN grup laporan untuk akun di wilayah tersebut.

```
aws codebuild list-report-groups
```

Output:

```
{
  "reportGroups": [
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-1",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-2",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-3"
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [ListReportGroups](#) di Referensi AWS CLI Perintah.

## **list-reports-for-report-group**

Contoh kode berikut menunjukkan cara menggunakan `list-reports-for-report-group`.

AWS CLI

Untuk mendapatkan daftar laporan dalam grup laporan di AWS CodeBuild.

`list-report-for-report-groups` Contoh berikut mengambil laporan dalam grup laporan yang ditentukan untuk akun di wilayah tersebut.

```
aws codebuild list-reports-for-report-group \
  --report-group-arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-
  group-name>
```

Output:

```
{
  "reports": [
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-1",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-2",
  ]
}
```

```

    "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-3"
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [ListReportsForReportGroup](#) di Referensi AWS CLI Perintah.

## list-reports

Contoh kode berikut menunjukkan cara menggunakan `list-reports`.

### AWS CLI

Untuk mendapatkan daftar laporan untuk akun saat ini di AWS CodeBuild.

`list-reports` Contoh berikut mengambil ARN dari laporan untuk akun saat ini.

```
aws codebuild list-reports
```

Output:

```

{
  "reports": [
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report ID>",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report ID>",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report ID>"
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [ListReports](#) di Referensi AWS CLI Perintah.

## list-shared-projects

Contoh kode berikut menunjukkan cara menggunakan `list-shared-projects`.

## AWS CLI

Untuk membuat daftar proyek bersama di AWS CodeBuild.

`list-shared-projects` Contoh berikut mencantumkan proyek CodeBuild bersama yang tersedia untuk akun saat ini.

```
aws codebuild list-shared-projects
```

Output:

```
{
  "projects": [
    "arn:aws:codebuild:<region-ID>:<account-ID>:project/<shared-project-name-1>",
    "arn:aws:codebuild:<region-ID>:<account-ID>:project/<shared-project-name-2>"
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan proyek bersama](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [ListSharedProjects](#) di Referensi AWS CLI Perintah.

## `list-shared-report-groups`

Contoh kode berikut menunjukkan cara menggunakan `list-shared-report-groups`.

## AWS CLI

Untuk mendapatkan daftar ARN grup laporan bersama. AWS CodeBuild

`list-shared-report-groups` Contoh berikut mengambil ARN grup laporan untuk akun di wilayah tersebut.

```
aws codebuild list-shared-report-groups
```

Output:

```
{
  "reportGroups": [
```



```

    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-1",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-2",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-3"
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [ListSharedReportGroups](#) di Referensi AWS CLI Perintah.

## list-source-credentials

Contoh kode berikut menunjukkan cara menggunakan `list-source-credentials`.

### AWS CLI

Untuk melihat daftar `sourceCredentialsObjects`

`list-source-credentials` Contoh berikut mencantumkan token untuk AWS akun yang terhubung ke satu akun Bitbucket dan satu GitHub akun. Setiap `sourceCredentialsInfos` objek dalam respon berisi informasi kredensial sumber yang terhubung.

```
aws codebuild list-source-credentials
```

Output:

```

{
  "sourceCredentialsInfos": [
    {
      "serverType": "BITBUCKET",
      "arn": "arn:aws:codebuild:us-west-2:123456789012:token/bitbucket",
      "authType": "BASIC_AUTH"
    },
    {
      "serverType": "GITHUB",
      "arn": "arn:aws:codebuild:us-west-2:123456789012:token/github",
      "authType": "OAUTH"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Connect Source Providers with Access Tokens \(CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [ListSourceCredentials](#) di Referensi AWS CLI Perintah.

## retry-build-batch

Contoh kode berikut menunjukkan cara menggunakan `retry-build-batch`.

### AWS CLI

Untuk mencoba lagi pembuatan batch yang gagal. AWS CodeBuild

`retry-build-batch` Contoh berikut memulai ulang build batch yang ditentukan.

```
aws codebuild retry-build-batch \  
  --id <project-name>:<batch-ID>
```

### Output:

```
{  
  "buildBatch": {  
    "id": "<project-name>:<batch-ID>",  
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-  
name>:<batch-ID>",  
    "startTime": "2020-10-21T17:26:23.099000+00:00",  
    "currentPhase": "SUBMITTED",  
    "buildBatchStatus": "IN_PROGRESS",  
    "resolvedSourceVersion": "3a9e11cb419e8fff14b03883dc4e64f6155aaa7e",  
    "projectName": "<project-name>",  
    "phases": [  
      {  
        "phaseType": "SUBMITTED",  
        "phaseStatus": "SUCCEEDED",  
        "startTime": "2020-10-21T17:26:23.099000+00:00",  
        "endTime": "2020-10-21T17:26:23.457000+00:00",  
        "durationInSeconds": 0  
      },  
      {  
        "phaseType": "DOWNLOAD_BATCHSPEC",  
        "phaseStatus": "SUCCEEDED",  
        "startTime": "2020-10-21T17:26:23.457000+00:00",  
        "endTime": "2020-10-21T17:26:54.902000+00:00",  
      }  
    ]  
  }  
}
```

```
        "durationInSeconds": 31
      },
      {
        "phaseType": "IN_PROGRESS",
        "phaseStatus": "CLIENT_ERROR",
        "startTime": "2020-10-21T17:26:54.902000+00:00",
        "endTime": "2020-10-21T17:28:16.060000+00:00",
        "durationInSeconds": 81
      },
      {
        "phaseType": "FAILED",
        "phaseStatus": "RETRY",
        "startTime": "2020-10-21T17:28:16.060000+00:00",
        "endTime": "2020-10-21T17:29:39.709000+00:00",
        "durationInSeconds": 83
      },
      {
        "phaseType": "SUBMITTED",
        "startTime": "2020-10-21T17:29:39.709000+00:00"
      }
    ],
    "source": {
      "type": "GITHUB",
      "location": "https://github.com/strohm-a/<project-name>-graph.git",
      "gitCloneDepth": 1,
      "gitSubmodulesConfig": {
        "fetchSubmodules": false
      },
      "reportBuildStatus": false,
      "insecureSsl": false
    },
    "secondarySources": [],
    "secondarySourceVersions": [],
    "artifacts": {
      "location": ""
    },
    "secondaryArtifacts": [],
    "cache": {
      "type": "NO_CACHE"
    },
    "environment": {
      "type": "LINUX_CONTAINER",
      "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
      "computeType": "BUILD_GENERAL1_SMALL",
```

```
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "logConfig": {
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  },
  "buildTimeoutInMinutes": 60,
  "queuedTimeoutInMinutes": 480,
  "complete": false,
  "initiator": "<username>",
  "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
  "buildBatchNumber": 4,
  "buildBatchConfig": {
    "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<project-
name>",
    "restrictions": {
      "maximumBuildsAllowed": 100
    },
    "timeoutInMins": 480
  },
  "buildGroups": [
    {
      "identifier": "DOWNLOAD_SOURCE",
      "ignoreFailure": false,
      "currentBuildSummary": {
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
        "requestedOn": "2020-10-21T17:26:23.889000+00:00",
        "buildStatus": "SUCCEEDED",
        "primaryArtifact": {
          "type": "no_artifacts",
          "identifier": "DOWNLOAD_SOURCE"
        },
        "secondaryArtifacts": []
      }
    }
  ],
  {

```

```

        "identifier": "linux_small",
        "dependsOn": [],
        "ignoreFailure": false,
        "currentBuildSummary": {
            "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
            "requestedOn": "2020-10-21T17:26:55.115000+00:00",
            "buildStatus": "FAILED",
            "primaryArtifact": {
                "type": "no_artifacts",
                "identifier": "linux_small"
            },
            "secondaryArtifacts": []
        }
    },
    {
        "identifier": "linux_medium",
        "dependsOn": [
            "linux_small"
        ],
        "ignoreFailure": false,
        "currentBuildSummary": {
            "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
            "requestedOn": "2020-10-21T17:26:54.594000+00:00",
            "buildStatus": "STOPPED"
        }
    },
    {
        "identifier": "linux_large",
        "dependsOn": [
            "linux_medium"
        ],
        "ignoreFailure": false,
        "currentBuildSummary": {
            "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
            "requestedOn": "2020-10-21T17:26:54.701000+00:00",
            "buildStatus": "STOPPED"
        }
    }
]
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Batch build AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [RetryBuildBatch](#) di Referensi AWS CLI Perintah.

## retry-build

Contoh kode berikut menunjukkan cara menggunakan `retry-build`.

### AWS CLI

Untuk mencoba lagi build in AWS CodeBuild yang gagal.

`retry-build` Contoh berikut memulai ulang build yang ditentukan.

```
aws codebuild retry-build \  
  --id <project-name>:<build-ID>
```

### Output:

```
{  
  "build": {  
    "id": "<project-name>:<build-ID>",  
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-  
name>:<build-ID>",  
    "buildNumber": 9,  
    "startTime": "2020-10-21T17:51:38.161000+00:00",  
    "currentPhase": "QUEUED",  
    "buildStatus": "IN_PROGRESS",  
    "projectName": "<project-name>",  
    "phases": [  
      {  
        "phaseType": "SUBMITTED",  
        "phaseStatus": "SUCCEEDED",  
        "startTime": "2020-10-21T17:51:38.161000+00:00",  
        "endTime": "2020-10-21T17:51:38.210000+00:00",  
        "durationInSeconds": 0  
      },  
      {  
        "phaseType": "QUEUED",  
        "startTime": "2020-10-21T17:51:38.210000+00:00"  
      }  
    ]  
  }  
}
```

```
    }
  ],
  "source": {
    "type": "GITHUB",
    "location": "<GitHub-repo-URL>",
    "gitCloneDepth": 1,
    "gitSubmodulesConfig": {
      "fetchSubmodules": false
    },
    "reportBuildStatus": false,
    "insecureSsl": false
  },
  "secondarySources": [],
  "secondarySourceVersions": [],
  "artifacts": {
    "location": ""
  },
  "secondaryArtifacts": [],
  "cache": {
    "type": "NO_CACHE"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<service-role-name>",
  "logs": {
    "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=<region-ID>#logEvent:group=null;stream=null",
    "cloudWatchLogsArn": "arn:aws:logs:<region-ID>:<account-ID>:log-group:null:log-stream:null",
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  }
},
```

```

    "timeoutInMinutes": 60,
    "queuedTimeoutInMinutes": 480,
    "buildComplete": false,
    "initiator": "<username>",
    "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3"
  }
}

```

Untuk informasi selengkapnya, lihat [Batch build AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [RetryBuild](#) di Referensi AWS CLI Perintah.

## start-build-batch

Contoh kode berikut menunjukkan cara menggunakan `start-build-batch`.

### AWS CLI

Untuk memulai pembuatan batch AWS CodeBuild.

`start-build-batch` Contoh berikut memulai batch build dari proyek yang ditentukan.

```

aws codebuild start-build-batch \
  --project-name <project-name>

```

Output:

```

{
  "buildBatch": {
    "id": "<project-name>:<batch-ID>",
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-name>:<batch-ID>",
    "startTime": "2020-10-21T16:54:24.740000+00:00",
    "currentPhase": "SUBMITTED",
    "buildBatchStatus": "IN_PROGRESS",
    "projectName": "<project-name>",
    "source": {
      "type": "GITHUB",
      "location": "<GitHub-repo-URL>",
      "gitCloneDepth": 1,
      "gitSubmodulesConfig": {
        "fetchSubmodules": false
      }
    }
  }
}

```



```
    },
    "reportBuildStatus": false,
    "insecureSsl": false
  },
  "secondarySources": [],
  "secondarySourceVersions": [],
  "artifacts": {
    "location": ""
  },
  "secondaryArtifacts": [],
  "cache": {
    "type": "NO_CACHE"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "logConfig": {
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  },
  "buildTimeoutInMinutes": 60,
  "queuedTimeoutInMinutes": 480,
  "complete": false,
  "initiator": "<username>",
  "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
  "buildBatchNumber": 3,
  "buildBatchConfig": {
    "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<service-
role-name>",
    "restrictions": {
      "maximumBuildsAllowed": 100
    }
  },
  "timeoutInMins": 480
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Batch build AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [StartBuildBatch](#) di Referensi AWS CLI Perintah.

## start-build

Contoh kode berikut menunjukkan cara menggunakan `start-build`.

### AWS CLI

Untuk mulai menjalankan build proyek AWS CodeBuild build.

`start-build` Contoh berikut memulai build untuk CodeBuild proyek yang ditentukan. Build mengesampingkan pengaturan proyek selama beberapa menit build diizinkan untuk antri sebelum waktu habis dan pengaturan artefak proyek.

```
aws codebuild start-build \  
  --project-name "my-demo-project" \  
  --queued-timeout-in-minutes-override 5 \  
  --artifacts-override {"\type\": \"S3\", \"location\": \"arn:aws:s3::artifacts-override\", \"overrideArtifactName\": true}
```

### Output:

```
{  
  "build": {  
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-service-role",  
    "buildStatus": "IN_PROGRESS",  
    "buildComplete": false,  
    "projectName": "my-demo-project",  
    "timeoutInMinutes": 60,  
    "source": {  
      "insecureSsl": false,  
      "type": "S3",  
      "location": "codebuild-us-west-2-123456789012-input-bucket/my-source.zip"  
    },  
    "queuedTimeoutInMinutes": 5,  
  }  
}
```

```
"encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
"currentPhase": "QUEUED",
"startTime": 1556905683.568,
"environment": {
  "computeType": "BUILD_GENERAL1_MEDIUM",
  "environmentVariables": [],
  "type": "LINUX_CONTAINER",
  "privilegedMode": false,
  "image": "aws/codebuild/standard:1.0",
  "imagePullCredentialsType": "CODEBUILD"
},
"phases": [
  {
    "phaseStatus": "SUCCEEDED",
    "startTime": 1556905683.568,
    "phaseType": "SUBMITTED",
    "durationInSeconds": 0,
    "endTime": 1556905684.524
  },
  {
    "startTime": 1556905684.524,
    "phaseType": "QUEUED"
  }
],
"logs": {
  "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#logEvent:group=null;stream=null"
},
"artifacts": {
  "encryptionDisabled": false,
  "location": "arn:aws:s3:::artifacts-override/my-demo-project",
  "overrideArtifactName": true
},
"cache": {
  "type": "NO_CACHE"
},
"id": "my-demo-project::12345678-a1b2-c3d4-e5f6-11111EXAMPLE",
"initiator": "my-aws-account-name",
"arn": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-project::12345678-a1b2-c3d4-e5f6-11111EXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Build \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [StartBuild](#) di Referensi AWS CLI Perintah.

## stop-build-batch

Contoh kode berikut menunjukkan cara menggunakan `stop-build-batch`.

### AWS CLI

Untuk menghentikan pembuatan batch yang sedang berlangsung. AWS CodeBuild `stop-build-batch` Contoh berikut menghentikan build batch yang ditentukan.

```
aws codebuild stop-build-batch \  
  --id <project-name>:<batch-ID>
```

### Output:

```
{  
  "buildBatch": {  
    "id": "<project-name>:<batch-ID>",  
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-name>:<batch-ID>",  
    "startTime": "2020-10-21T16:54:24.740000+00:00",  
    "endTime": "2020-10-21T16:56:05.152000+00:00",  
    "currentPhase": "STOPPED",  
    "buildBatchStatus": "STOPPED",  
    "resolvedSourceVersion": "aef7744ed069c51098e15c360f4102cd2cd1ad64",  
    "projectName": "<project-name>",  
    "phases": [  
      {  
        "phaseType": "SUBMITTED",  
        "phaseStatus": "SUCCEEDED",  
        "startTime": "2020-10-21T16:54:24.740000+00:00",  
        "endTime": "2020-10-21T16:54:25.039000+00:00",  
        "durationInSeconds": 0  
      },  
      {  
        "phaseType": "DOWNLOAD_BATCHSPEC",  
        "phaseStatus": "SUCCEEDED",  
        "startTime": "2020-10-21T16:54:25.039000+00:00",  
        "durationInSeconds": 0  
      }  
    ]  
  }  
}
```

```

        "endTime": "2020-10-21T16:54:56.583000+00:00",
        "durationInSeconds": 31
    },
    {
        "phaseType": "IN_PROGRESS",
        "phaseStatus": "STOPPED",
        "startTime": "2020-10-21T16:54:56.583000+00:00",
        "endTime": "2020-10-21T16:56:05.152000+00:00",
        "durationInSeconds": 68
    },
    {
        "phaseType": "STOPPED",
        "startTime": "2020-10-21T16:56:05.152000+00:00"
    }
],
"source": {
    "type": "GITHUB",
    "location": "<GitHub-repo-URL>",
    "gitCloneDepth": 1,
    "gitSubmodulesConfig": {
        "fetchSubmodules": false
    },
    "reportBuildStatus": false,
    "insecureSsl": false
},
"secondarySources": [],
"secondarySourceVersions": [],
"artifacts": {
    "location": ""
},
"secondaryArtifacts": [],
"cache": {
    "type": "NO_CACHE"
},
"environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
},
"logConfig": {
    "cloudWatchLogs": {

```

```

        "status": "ENABLED"
    },
    "s3Logs": {
        "status": "DISABLED",
        "encryptionDisabled": false
    }
},
"buildTimeoutInMinutes": 60,
"queuedTimeoutInMinutes": 480,
"complete": true,
"initiator": "Strohm",
"encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
"buildBatchNumber": 3,
"buildBatchConfig": {
    "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<project-
name>",
    "restrictions": {
        "maximumBuildsAllowed": 100
    },
    "timeoutInMins": 480
},
"buildGroups": [
    {
        "identifier": "DOWNLOAD_SOURCE",
        "ignoreFailure": false,
        "currentBuildSummary": {
            "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
            "requestedOn": "2020-10-21T16:54:25.468000+00:00",
            "buildStatus": "SUCCEEDED",
            "primaryArtifact": {
                "type": "no_artifacts",
                "identifier": "DOWNLOAD_SOURCE"
            },
            "secondaryArtifacts": []
        },
        "dependsOn": [],
        "ignoreFailure": false,
        "currentBuildSummary": {
            "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",

```

```

        "requestedOn": "2020-10-21T16:54:56.833000+00:00",
        "buildStatus": "IN_PROGRESS"
    }
},
{
    "identifier": "linux_medium",
    "dependsOn": [
        "linux_small"
    ],
    "ignoreFailure": false,
    "currentBuildSummary": {
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
        "requestedOn": "2020-10-21T16:54:56.211000+00:00",
        "buildStatus": "PENDING"
    }
},
{
    "identifier": "linux_large",
    "dependsOn": [
        "linux_medium"
    ],
    "ignoreFailure": false,
    "currentBuildSummary": {
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
        "requestedOn": "2020-10-21T16:54:56.330000+00:00",
        "buildStatus": "PENDING"
    }
}
]
}
}

```

Untuk informasi selengkapnya, lihat [Batch build AWS CodeBuild di Panduan AWS CodeBuild Pengguna](#).

- Untuk detail API, lihat [StopBuildBatch](#) di Referensi AWS CLI Perintah.

## stop-build

Contoh kode berikut menunjukkan cara menggunakan `stop-build`.

## AWS CLI

Untuk menghentikan pembangunan proyek AWS CodeBuild pembangunan.

`stop-build` Contoh berikut menghentikan CodeBuild build yang ditentukan.

```
aws codebuild stop-build --id my-demo-project:12345678-a1b2-c3d4-e5f6-11111EXAMPLE
```

Output:

```
{
  "build": {
    "startTime": 1556906956.318,
    "initiator": "my-aws-account-name",
    "projectName": "my-demo-project",
    "currentPhase": "COMPLETED",
    "cache": {
      "type": "NO_CACHE"
    },
    "source": {
      "insecureSsl": false,
      "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip",
      "type": "S3"
    },
    "id": "my-demo-project:1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
    "endTime": 1556906974.781,
    "phases": [
      {
        "durationInSeconds": 0,
        "phaseType": "SUBMITTED",
        "endTime": 1556906956.935,
        "phaseStatus": "SUCCEEDED",
        "startTime": 1556906956.318
      },
      {
        "durationInSeconds": 1,
        "phaseType": "QUEUED",
        "endTime": 1556906958.272,
        "phaseStatus": "SUCCEEDED",
        "startTime": 1556906956.935
      },
      {

```



```
    "phaseType": "PROVISIONING",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 14,
    "contexts": [
      {
        "message": "",
        "statusCode": ""
      }
    ],
    "endTime": 1556906972.847,
    "startTime": 1556906958.272
  },
  {
    "phaseType": "DOWNLOAD_SOURCE",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 0,
    "contexts": [
      {
        "message": "",
        "statusCode": ""
      }
    ],
    "endTime": 1556906973.552,
    "startTime": 1556906972.847
  },
  {
    "phaseType": "INSTALL",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 0,
    "contexts": [
      {
        "message": "",
        "statusCode": ""
      }
    ],
    "endTime": 1556906973.75,
    "startTime": 1556906973.552
  },
  {
    "phaseType": "PRE_BUILD",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 0,
    "contexts": [
      {
```

```
        "message": "",
        "statusCode": ""
    }
  ],
  "endTime": 1556906973.937,
  "startTime": 1556906973.75
},
{
  "durationInSeconds": 0,
  "phaseType": "BUILD",
  "endTime": 1556906974.781,
  "phaseStatus": "STOPPED",
  "startTime": 1556906973.937
},
{
  "phaseType": "COMPLETED",
  "startTime": 1556906974.781
}
],
"artifacts": {
  "location": "arn:aws:s3:::artifacts-override/my-demo-project",
  "encryptionDisabled": false,
  "overrideArtifactName": true
},
"buildComplete": true,
"buildStatus": "STOPPED",
"encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
"serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
"queuedTimeoutInMinutes": 5,
"timeoutInMinutes": 60,
"environment": {
  "type": "LINUX_CONTAINER",
  "environmentVariables": [],
  "computeType": "BUILD_GENERAL1_MEDIUM",
  "privilegedMode": false,
  "image": "aws/codebuild/standard:1.0",
  "imagePullCredentialsType": "CODEBUILD"
},
"logs": {
  "streamName": "1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
  "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-
west-2#logEvent:group=/aws/codebuild/my-demo-project;stream=1a2b3c4d-5678-90ab-
cdef-11111EXAMPLE",
```

```

        "groupName": "/aws/codebuild/my-demo-project"
    },
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-
project:1a2b3c4d-5678-90ab-cdef-11111EXAMPLE"
}
}

```

Untuk informasi selengkapnya, lihat [Menghentikan Build \(AWS CLI\)](#) di AWS CodeBuild Panduan Pengguna.

- Untuk detail API, lihat [StopBuild](#) di Referensi AWS CLI Perintah.

## update-project

Contoh kode berikut menunjukkan cara menggunakan `update-project`.

### AWS CLI

Untuk mengubah pengaturan proyek AWS CodeBuild build.

`update-project` Contoh berikut mengubah pengaturan proyek CodeBuild build yang ditentukan bernama `my-demo-project`.

```

aws codebuild update-project --name "my-demo-project" \
  --description "This project is updated" \
  --source "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-
input-bucket/my-source-2.zip\"}" \
  --artifacts "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-
output-bucket-2\"}" \
  --environment "{\"type\": \"LINUX_CONTAINER\", \"image\": \"aws/codebuild/
standard:1.0\", \"computeType\": \"BUILD_GENERAL1_MEDIUM\"}" \
  --service-role "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role"

```

Output menampilkan pengaturan yang diperbarui.

```

{
  "project": {
    "arn": "arn:aws:codebuild:us-west-2:123456789012:project/my-demo-project",
    "environment": {
      "privilegedMode": false,

```

```

        "environmentVariables": [],
        "type": "LINUX_CONTAINER",
        "image": "aws/codebuild/standard:1.0",
        "computeType": "BUILD_GENERAL1_MEDIUM",
        "imagePullCredentialsType": "CODEBUILD"
    },
    "queuedTimeoutInMinutes": 480,
    "description": "This project is updated",
    "artifacts": {
        "packaging": "NONE",
        "name": "my-demo-project",
        "type": "S3",
        "namespaceType": "NONE",
        "encryptionDisabled": false,
        "location": "codebuild-us-west-2-123456789012-output-bucket-2"
    },
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "badge": {
        "badgeEnabled": false
    },
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
    "lastModified": 1556840545.967,
    "tags": [],
    "timeoutInMinutes": 60,
    "created": 1556839783.274,
    "name": "my-demo-project",
    "cache": {
        "type": "NO_CACHE"
    },
    "source": {
        "type": "S3",
        "insecureSsl": false,
        "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source-2.zip"
    }
}

```

Untuk informasi selengkapnya, lihat [Mengubah Setelan \(AWS CLI\) Build Project](#) di Panduan Pengguna AWS CodeBuild

- Untuk detail API, lihat [UpdateProject](#) di Referensi AWS CLI Perintah.

## update-report-group

Contoh kode berikut menunjukkan cara menggunakan `update-report-group`.

### AWS CLI

Untuk memperbarui grup laporan di AWS CodeBuild.

`update-report-group` Contoh berikut mengubah jenis ekspor grup laporan menjadi "NO\_EXPORT".

```
aws codebuild update-report-group \
  --arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-report-
group \
  --export-config="exportConfigType=NO_EXPORT"
```

Output:

```
{
  "reportGroup": {
    "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-
report-group",
    "name": "cli-created-report-group",
    "type": "TEST",
    "exportConfig": {
      "exportConfigType": "NO_EXPORT"
    },
    "created": 1602020686.009,
    "lastModified": 1602021033.454,
    "tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup laporan](#) di Panduan AWS CodeBuild Pengguna.

- Untuk detail API, lihat [UpdateReportGroup](#) di Referensi AWS CLI Perintah.

## update-webhook

Contoh kode berikut menunjukkan cara menggunakan `update-webhook`.

## AWS CLI

Untuk memperbarui webhook untuk sebuah proyek AWS CodeBuild

`update-webhook` Contoh berikut memperbarui webhook untuk CodeBuild proyek yang ditentukan dengan dua grup filter. `--rotate-secret` Parameter menentukan bahwa GitHub memutar kunci rahasia proyek setiap kali perubahan kode memicu build. Grup filter pertama menentukan permintaan tarik yang dibuat, diperbarui, atau dibuka kembali pada cabang dengan nama referensi Git yang cocok dengan ekspresi reguler `^refs/heads/master$` dan referensi kepala yang cocok. `^refs/heads/myBranch$` Grup filter kedua menentukan permintaan push pada cabang dengan nama referensi Git yang tidak cocok dengan ekspresi `^refs/heads/myBranch$` reguler.

```
aws codebuild update-webhook \
  --project-name Project2 \
  --rotate-secret \
  --filter-groups "[[{"type\\":\\"EVENT\\",\\"pattern\\":\\"PULL_REQUEST_CREATED,
  PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED\\"}, {"type\\":\\"HEAD_REF\\",\\"pattern
  \\":\\"^refs/heads/myBranch$\\",\\"excludeMatchedPattern\\":true}, {"type\\":\\"BASE_REF
  \\",\\"pattern\\":\\"^refs/heads/master$\\",\\"excludeMatchedPattern\\":true}], [{"type\\":
  \\"EVENT\\",\\"pattern\\":\\"PUSH\\"}, {"type\\":\\"HEAD_REF\\",\\"pattern\\":\\"^refs/heads/
  myBranch$\\",\\"excludeMatchedPattern\\":true}]]"
```

Output:

```
{
  "webhook": {
    "filterGroups": [
      [
        {
          "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
          PULL_REQUEST_REOPENED",
          "type": "EVENT"
        },
        {
          "excludeMatchedPattern": true,
          "pattern": "refs/heads/myBranch$",
          "type": "HEAD_REF"
        },
        {
          "excludeMatchedPattern": true,
          "pattern": "refs/heads/master$",
```

```
        "type": "BASE_REF"
      }
    ],
    [
      {
        "pattern": "PUSH",
        "type": "EVENT"
      },
      {
        "excludeMatchedPattern": true,
        "pattern": "refs/heads/myBranch$",
        "type": "HEAD_REF"
      }
    ]
  ],
  "lastModifiedSecret": 1556312220.133
}
}
```

Untuk informasi selengkapnya, lihat [Mengubah Setelan \(AWS CLI\) Build Project](#) di Panduan Pengguna AWS CodeBuild

- Untuk detail API, lihat [UpdateWebhook](#) di Referensi AWS CLI Perintah.

## CodeCommit contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CodeCommit.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **associate-approval-rule-template-with-repository**

Contoh kode berikut menunjukkan cara menggunakan `associate-approval-rule-template-with-repository`.

#### AWS CLI

Untuk mengaitkan template aturan persetujuan dengan repositori

`associate-approval-rule-template-with-repository` Contoh berikut mengaitkan template aturan persetujuan yang ditentukan dengan repositori bernama `MyDemoRepo`

```
aws codecommit associate-approval-rule-template-with-repository \  
  --repository-name MyDemoRepo \  
  --approval-rule-template-name 2-approver-rule-for-main
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan Template Aturan Persetujuan dengan Repositori](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [AssociateApprovalRuleTemplateWithRepository](#) di Referensi AWS CLI Perintah.

### **batch-associate-approval-rule-template-with-repositories**

Contoh kode berikut menunjukkan cara menggunakan `batch-associate-approval-rule-template-with-repositories`.

#### AWS CLI

Untuk mengaitkan template aturan persetujuan dengan beberapa repositori dalam satu operasi

`batch-associate-approval-rule-template-with-repositories` Contoh berikut mengaitkan template aturan persetujuan yang ditentukan dengan repositori bernama `MyDemoRepo` dan `MyOtherDemoRepo`

Catatan: Templat aturan persetujuan khusus untuk AWS Wilayah tempat mereka dibuat. Mereka hanya dapat dikaitkan dengan repositori di Wilayah itu AWS .



```
aws codecommit batch-associate-approval-rule-template-with-repositories \  
  --repository-names MyDemoRepo, MyOtherDemoRepo \  
  --approval-rule-template-name 2-approver-rule-for-main
```

Output:

```
{  
  "associatedRepositoryNames": [  
    "MyDemoRepo",  
    "MyOtherDemoRepo"  
  ],  
  "errors": []  
}
```

Untuk informasi selengkapnya, lihat [Mengaitkan Template Aturan Persetujuan dengan Repositori](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [BatchAssociateApprovalRuleTemplateWithRepositories](#) di Referensi AWS CLI Perintah.

## batch-describe-merge-conflicts

Contoh kode berikut menunjukkan cara menggunakan `batch-describe-merge-conflicts`.

### AWS CLI

Untuk mendapatkan informasi tentang konflik gabungan di semua file atau subset file dalam penggabungan antara dua penentu komit

`batch-describe-merge-conflicts` Contoh berikut menentukan konflik gabungan untuk menggabungkan cabang sumber bernama `feature-randomizationfeature` dengan cabang tujuan bernama `main` menggunakan `THREE_WAY_MERGE` strategi dalam repositori bernama `MyDemoRepo`

```
aws codecommit batch-describe-merge-conflicts \  
  --source-commit-specifier feature-randomizationfeature \  
  --destination-commit-specifier main \  
  --merge-option THREE_WAY_MERGE \  
  --repository-name MyDemoRepo
```

Output:

```
{
  "conflicts": [
    {
      "conflictMetadata": {
        "filePath": "readme.md",
        "fileSizes": {
          "source": 139,
          "destination": 230,
          "base": 85
        },
        "fileModes": {
          "source": "NORMAL",
          "destination": "NORMAL",
          "base": "NORMAL"
        },
        "objectTypes": {
          "source": "FILE",
          "destination": "FILE",
          "base": "FILE"
        },
        "numberOfConflicts": 1,
        "isBinaryFile": {
          "source": false,
          "destination": false,
          "base": false
        },
        "contentConflict": true,
        "fileModeConflict": false,
        "objectTypeConflict": false,
        "mergeOperations": {
          "source": "M",
          "destination": "M"
        }
      },
      "mergeHunks": [
        {
          "isConflict": true,
          "source": {
            "startLine": 0,
            "endLine": 3,
            "hunkContent": "VGhpcyBpEXAMPLE=="
          },
          "destination": {
```

```

        "startLine": 0,
        "endLine": 1,
        "hunkContent": "VXNlIHRoEXAMPLE="
      }
    ]
  },
  "errors": [],
  "destinationCommitId": "86958e0aEXAMPLE",
  "sourceCommitId": "6ccd57fdEXAMPLE",
  "baseCommitId": "767b6958EXAMPLE"
}

```

Untuk informasi selengkapnya, lihat [Menyelesaikan Konflik dalam Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [BatchDescribeMergeConflicts](#) di Referensi AWS CLI Perintah.

## batch-disassociate-approval-rule-template-from-repositories

Contoh kode berikut menunjukkan cara menggunakan `batch-disassociate-approval-rule-template-from-repositories`.

### AWS CLI

Untuk memisahkan template aturan persetujuan dari beberapa repositori dalam satu operasi

`batch-disassociate-approval-rule-template-from-repositories` Contoh berikut memisahkan template aturan persetujuan yang ditentukan dari repositori bernama `MyDemoRepo` dan `MyOtherDemoRepo`.

```

aws codecommit batch-disassociate-approval-rule-template-from-repositories \
  --repository-names MyDemoRepo, MyOtherDemoRepo \
  --approval-rule-template-name 1-approval-rule-for-all pull requests

```

### Output:

```

{
  "disassociatedRepositoryNames": [
    "MyDemoRepo",
    "MyOtherDemoRepo"
  ]
}

```

```
  ],  
  "errors": []  
}
```

Untuk informasi selengkapnya, lihat [Memutuskan Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [BatchDisassociateApprovalRuleTemplateFromRepositories](#) di Referensi AWS CLI Perintah.

## batch-get-commits

Contoh kode berikut menunjukkan cara menggunakan `batch-get-commits`.

### AWS CLI

Untuk melihat informasi tentang beberapa komit

`batch-get-commits` Contoh berikut menampilkan rincian tentang commit yang ditentukan.

```
aws codecommit batch-get-commits \  
  --repository-name MyDemoRepo \  
  --commit-ids 317f8570EXAMPLE 4c925148EXAMPLE
```

### Output:

```
{  
  "commits": [  
    {  
      "additionalData": "",  
      "committer": {  
        "date": "1508280564 -0800",  
        "name": "Mary Major",  
        "email": "mary_major@example.com"  
      },  
      "author": {  
        "date": "1508280564 -0800",  
        "name": "Mary Major",  
        "email": "mary_major@example.com"  
      },  
      "commitId": "317f8570EXAMPLE",  
      "treeId": "1f330709EXAMPLE",
```

```
    "parents": [
      "6e147360EXAMPLE"
    ],
    "message": "Change variable name and add new response element"
  },
  {
    "additionalData": "",
    "committer": {
      "date": "1508280542 -0800",
      "name": "Li Juan",
      "email": "li_juan@example.com"
    },
    "author": {
      "date": "1508280542 -0800",
      "name": "Li Juan",
      "email": "li_juan@example.com"
    },
    "commitId": "4c925148EXAMPLE",
    "treeId": "1f330709EXAMPLE",
    "parents": [
      "317f8570EXAMPLE"
    ],
    "message": "Added new class"
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Komit](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [BatchGetCommits](#) di Referensi AWS CLI Perintah.

## batch-get-repositories

Contoh kode berikut menunjukkan cara menggunakan `batch-get-repositories`.

### AWS CLI

Untuk melihat detail tentang beberapa repositori

Contoh ini menunjukkan rincian tentang beberapa AWS CodeCommit repositori.

```
aws codecommit batch-get-repositories \  
  --repository-names MyDemoRepo MyOtherDemoRepo
```

## Output:

```
{
  "repositoriesNotFound": [],
  "repositories": [
    {
      "creationDate": 1429203623.625,
      "defaultBranch": "main",
      "repositoryName": "MyDemoRepo",
      "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
      "lastModifiedDate": 1430783812.0869999,
      "repositoryDescription": "My demonstration repository",
      "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
      "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
      "accountId": "111111111111"
    },
    {
      "creationDate": 1429203623.627,
      "defaultBranch": "main",
      "repositoryName": "MyOtherDemoRepo",
      "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyOtherDemoRepo",
      "lastModifiedDate": 1430783812.0889999,
      "repositoryDescription": "My other demonstration repository",
      "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/MyOtherDemoRepo",
      "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE",
      "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo",
      "accountId": "111111111111"
    }
  ],
  "repositoriesNotFound": []
}
```

- Untuk detail API, lihat [BatchGetRepositories](#) di Referensi AWS CLI Perintah.

## create-approval-rule-template

Contoh kode berikut menunjukkan cara menggunakan `create-approval-rule-template`.

## AWS CLI

Untuk membuat templat aturan persetujuan

`create-approval-rule-template` Contoh berikut membuat template aturan persetujuan bernama `2-approver-rule-for-main`. The template requires two users who assume the role of `CodeCommitReview` untuk menyetujui permintaan tarik sebelum dapat digabungkan ke cabang. `main`

```
aws codecommit create-approval-rule-template \
  --approval-rule-template-name 2-approver-rule-for-main \
  --approval-rule-template-description "Requires two developers from the team to
  approve the pull request if the destination branch is main" \
  --approval-rule-template-content "{\"Version\": \"2018-11-08\",
  \"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
  \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
  [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Output:

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "2-approver-rule-for-main",
    "creationDate": 1571356106.936,
    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",
    \"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
    \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
    [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "approvalRuleTemplateDescription": "Requires two developers from the team to
    approve the pull request if the destination branch is main",
    "lastModifiedDate": 1571356106.936,
    "ruleContentSha256": "4711b576EXAMPLE"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [CreateApprovalRuleTemplate](#) di Referensi AWS CLI Perintah.

## create-branch

Contoh kode berikut menunjukkan cara menggunakan `create-branch`.

### AWS CLI

Untuk membuat cabang

Contoh ini membuat cabang dalam AWS CodeCommit repositori. Perintah ini menghasilkan output hanya jika ada kesalahan.

Perintah:

```
aws codecommit create-branch --repository-name MyDemoRepo --branch-name MyNewBranch
--commit-id 317f8570EXAMPLE
```

Output:

```
None.
```

- Untuk detail API, lihat [CreateBranch](#) di Referensi AWS CLI Perintah.

## create-commit

Contoh kode berikut menunjukkan cara menggunakan `create-commit`.

### AWS CLI

Untuk membuat komit

`create-commit` Contoh berikut menunjukkan cara membuat komit awal untuk repositori yang menambahkan `readme.md` file ke repositori bernama di cabang. `MyDemoRepo` `main`

```
aws codecommit create-commit \
  --repository-name MyDemoRepo \
  --branch-name main \
  --put-files "filePath=readme.md,fileContent='Welcome to our team repository.'"
```

Output:

```
{
```



```

    "filesAdded": [
      {
        "blobId": "5e1c309d-EXAMPLE",
        "absolutePath": "readme.md",
        "fileMode": "NORMAL"
      }
    ],
    "commitId": "4df8b524-EXAMPLE",
    "treeId": "55b57003-EXAMPLE",
    "filesDeleted": [],
    "filesUpdated": []
  }

```

Untuk informasi selengkapnya, lihat [Membuat Komit AWS CodeCommit di](#) Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [CreateCommit](#) di Referensi AWS CLI Perintah.

## create-pull-request-approval-rule

Contoh kode berikut menunjukkan cara menggunakan `create-pull-request-approval-rule`.

### AWS CLI

Untuk membuat aturan persetujuan untuk permintaan tarik

`create-pull-request-approval-rule` Contoh berikut membuat aturan persetujuan bernama `Require two approved approvers` untuk permintaan tarik tertentu. Aturan menetapkan bahwa dua persetujuan diperlukan dari kumpulan persetujuan. Pool mencakup semua pengguna yang mengakses CodeCommit dengan mengambil peran `CodeCommitReview` dalam `123456789012` AWS akun. Ini juga mencakup pengguna IAM atau pengguna federasi bernama `Nikhil_Jayashankar` dari akun yang sama AWS .

```

aws codecommit create-pull-request-approval-rule \
  --approval-rule-name "Require two approved approvers" \
  --approval-rule-content "{\"Version\": \"2018-11-08\", \"Statements\":
  [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers
  \": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
  \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"

```

Output:

```
{
  "approvalRule": {
    "approvalRuleName": "Require two approved approvers",
    "lastModifiedDate": 1570752871.932,
    "ruleContentSha256": "7c44e6ebEXAMPLE",
    "creationDate": 1570752871.932,
    "approvalRuleId": "aac33506-EXAMPLE",
    "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\":
  [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers
  \": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
  \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}}]",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [CreatePullRequestApprovalRule](#) di Referensi AWS CLI Perintah.

## create-pull-request

Contoh kode berikut menunjukkan cara menggunakan `create-pull-request`.

### AWS CLI

Untuk membuat permintaan tarik

`create-pull-request` Contoh berikut membuat permintaan tarik bernama 'Penganalisis kesulitan pengucapan' dengan deskripsi 'Harap tinjau perubahan ini pada hari Selasa' yang menargetkan cabang sumber 'jane-branch' dan akan digabungkan ke cabang default 'main' di repositori bernama ". AWS CodeCommit MyDemoRepo

```
aws codecommit create-pull-request \
  --title "My Pull Request" \
  --description "Please review these changes by Tuesday" \
  --client-request-token 123Example \
  --targets repositoryName=MyDemoRepo,sourceReference=MyNewBranch
```

Output:

```
{
```

```

"pullRequest": {
  "approvalRules": [
    {
      "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\\\"DestinationReferences\\\": [\\\"refs/heads/main\\\"],\\\"Statements\\\": [{\\\"Type
\\\": \\\"Approvers\\\",\\\"NumberOfApprovalsNeeded\\\": 2,\\\"ApprovalPoolMembers\\\":
[\\\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\\\"]}}]",
      "approvalRuleId": "dd8b17fe-EXAMPLE",
      "approvalRuleName": "2-approver-rule-for-main",
      "creationDate": 1571356106.936,
      "lastModifiedDate": 571356106.936,
      "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
      "originApprovalRuleTemplate": {
        "approvalRuleTemplateId": "dd3d22fe-EXAMPLE",
        "approvalRuleTemplateName": "2-approver-rule-for-main"
      },
      "ruleContentSha256": "4711b576EXAMPLE"
    }
  ],
  "authorArn": "arn:aws:iam::111111111111:user/Jane_Doe",
  "description": "Please review these changes by Tuesday",
  "title": "Pronunciation difficulty analyzer",
  "pullRequestTargets": [
    {
      "destinationCommit": "5d036259EXAMPLE",
      "destinationReference": "refs/heads/main",
      "repositoryName": "MyDemoRepo",
      "sourceCommit": "317f8570EXAMPLE",
      "sourceReference": "refs/heads/jane-branch",
      "mergeMetadata": {
        "isMerged": false
      }
    }
  ],
  "lastActivityDate": 1508962823.285,
  "pullRequestId": "42",
  "clientRequestToken": "123Example",
  "pullRequestStatus": "OPEN",
  "creationDate": 1508962823.285
}
}

```

- Untuk detail API, lihat [CreatePullRequest](#) di Referensi AWS CLI Perintah.

## create-repository

Contoh kode berikut menunjukkan cara menggunakan `create-repository`.

### AWS CLI

Untuk membuat repositori

Contoh ini membuat repositori dan mengaitkannya dengan akun pengguna. AWS

Perintah:

```
aws codecommit create-repository --repository-name MyDemoRepo --repository-
description "My demonstration repository"
```

Output:

```
{
  "repositoryMetadata": {
    "repositoryName": "MyDemoRepo",
    "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/
repos/MyDemoRepo",
    "lastModifiedDate": 1444766838.027,
    "repositoryDescription": "My demonstration repository",
    "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/
repos/MyDemoRepo",
    "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    "Arn": "arn:aws:codecommit:us-
east-1:111111111111EXAMPLE:MyDemoRepo",
    "accountId": "111111111111"
  }
}
```

- Untuk detail API, lihat [CreateRepository](#) di Referensi AWS CLI Perintah.

## create-unreferenced-merge-commit

Contoh kode berikut menunjukkan cara menggunakan `create-unreferenced-merge-commit`.

### AWS CLI

Untuk membuat komit yang tidak direferensikan yang mewakili hasil penggabungan dua penentu komit

`create-unreferenced-merge-commit` Contoh berikut membuat komit yang mewakili hasil penggabungan antara cabang sumber bernama `bugfix-1234` dengan cabang tujuan bernama `main` menggunakan strategi `THREE_WAY_MERGE` dalam repositori bernama `MyDemoRepo`

```
aws codecommit create-unreferenced-merge-commit \  
  --source-commit-specifier bugfix-1234 \  
  --destination-commit-specifier main \  
  --merge-option THREE_WAY_MERGE \  
  --repository-name MyDemoRepo \  
  --name "Maria Garcia" \  
  --email "maria_garcia@example.com" \  
  --commit-message "Testing the results of this merge."
```

Output:

```
{  
  "commitId": "4f178133EXAMPLE",  
  "treeId": "389765daEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menyelesaikan Konflik dalam Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [CreateUnreferencedMergeCommit](#) di Referensi AWS CLI Perintah.

## credential-helper

Contoh kode berikut menunjukkan cara menggunakan `credential-helper`.

### AWS CLI

Untuk mengatur pembantu kredensial yang termasuk dalam CLI AWS dengan AWS CodeCommit `credential-helper` Utilitas tidak dirancang untuk dipanggil langsung dari AWS CLI. Sebaliknya itu dimaksudkan untuk digunakan sebagai parameter dengan `git config` perintah untuk mengatur komputer lokal Anda. Ini memungkinkan Git untuk menggunakan HTTPS dan versi yang ditandatangani secara kriptografis dari kredensial pengguna IAM Anda atau peran instans Amazon EC2 setiap kali Git perlu mengautentikasi untuk berinteraksi dengan repositori. AWS CodeCommit

```
git config --global credential.helper '!aws codecommit credential-helper $@'
```

```
git config --global credential.UseHttpPath true
```

Output:

```
[credential]
  helper = !aws codecommit credential-helper $@
  UseHttpPath = true
```

Untuk informasi selengkapnya, lihat [Menyiapkan untuk AWS CodeCommit Menggunakan Metode Lain](#) di Panduan AWS CodeCommit Pengguna. Tinjau konten dengan seksama, lalu ikuti prosedur di salah satu topik berikut: [Untuk Koneksi HTTPS di Linux, macOS, atau Unix](#) atau [Untuk Koneksi HTTPS di Windows](#) di Panduan Pengguna.AWS CodeCommit

- Untuk detail API, lihat [CredentialHelper](#) di Referensi AWS CLI Perintah.

## delete-approval-rule-template

Contoh kode berikut menunjukkan cara menggunakan `delete-approval-rule-template`.

AWS CLI

Untuk menghapus templat aturan persetujuan

`delete-approval-rule-template` Contoh berikut menghapus template aturan persetujuan yang ditentukan.

```
aws codecommit delete-approval-rule-template \
  --approval-rule-template-name 1-approver-for-all-pull-requests
```

Output:

```
{
  "approvalRuleTemplateId": "41de97b7-EXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menghapus Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [DeleteApprovalRuleTemplate](#) di Referensi AWS CLI Perintah.

## delete-branch

Contoh kode berikut menunjukkan cara menggunakan `delete-branch`.

### AWS CLI

Untuk menghapus cabang

Contoh ini menunjukkan cara menghapus cabang di AWS CodeCommit repositori.

Perintah:

```
aws codecommit delete-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Output:

```
{
  "branch": {
    "commitId": "317f8570EXAMPLE",
    "branchName": "MyNewBranch"
  }
}
```

- Untuk detail API, lihat [DeleteBranch](#) di Referensi AWS CLI Perintah.

## delete-comment-content

Contoh kode berikut menunjukkan cara menggunakan `delete-comment-content`.

### AWS CLI

Untuk menghapus konten komentar

Anda hanya dapat menghapus konten komentar jika Anda membuat komentar. Contoh ini menunjukkan cara menghapus konten komentar dengan ID yang dihasilkan sistem dari `ff30b348EXAMPLEb9aa670f`

```
aws codecommit delete-comment-content \
  --comment-id ff30b348EXAMPLEb9aa670f
```

Output:

```
{
  "comment": {
    "creationDate": 1508369768.142,
    "deleted": true,
    "lastModifiedDate": 1508369842.278,
    "clientRequestToken": "123Example",
    "commentId": "ff30b348EXAMPLEb9aa670f",
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "callerReactions": [],
    "reactionCounts":
    {
      "CLAP" : 1
    }
  }
}
```

- Untuk detail API, lihat [DeleteCommentContent](#) di Referensi AWS CLI Perintah.

## delete-file

Contoh kode berikut menunjukkan cara menggunakan `delete-file`.

### AWS CLI

Untuk menghapus file

`delete-file` Contoh berikut menunjukkan cara menghapus file bernama `README.md` dari cabang bernama `main` dengan ID komit terbaru dari `c5709475EXAMPLE` dalam repositori bernama `MyDemoRepo`

```
aws codecommit delete-file \
  --repository-name MyDemoRepo \
  --branch-name main \
  --file-path README.md \
  --parent-commit-id c5709475EXAMPLE
```

Output:

```
{
  "blobId": "559b44fEXAMPLE",
  "commitId": "353cf655EXAMPLE",
```



```
"filePath":"README.md",  
"treeId":"6bc824cEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Mengedit atau Menghapus File AWS CodeCommit dalam panduan Referensi AWS CodeCommit API](#).

- Untuk detail API, lihat [DeleteFile](#) di Referensi AWS CLI Perintah.

## **delete-pull-request-approval-rule**

Contoh kode berikut menunjukkan cara menggunakan `delete-pull-request-approval-rule`.

### AWS CLI

Untuk menghapus aturan persetujuan untuk permintaan tarik

`delete-pull-request-approval-rule` Contoh berikut menghapus aturan persetujuan yang dinamai `My Approval Rule` untuk permintaan tarik yang ditentukan.

```
aws codecommit delete-pull-request-approval-rule \  
  --approval-rule-name "My Approval Rule" \  
  --pull-request-id 15
```

### Output:

```
{  
  "approvalRuleId": "077d8e8a8-EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Mengedit atau Menghapus Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [DeletePullRequestApprovalRule](#) di Referensi AWS CLI Perintah.

## **delete-repository**

Contoh kode berikut menunjukkan cara menggunakan `delete-repository`.

### AWS CLI

Untuk menghapus repositori

Contoh ini menunjukkan cara menghapus AWS CodeCommit repositori.

Perintah:

```
aws codecommit delete-repository --repository-name MyDemoRepo
```

Output:

```
{
  "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE"
}
```

- Untuk detail API, lihat [DeleteRepository](#) di Referensi AWS CLI Perintah.

## describe-merge-conflicts

Contoh kode berikut menunjukkan cara menggunakan `describe-merge-conflicts`.

AWS CLI

Untuk mendapatkan informasi rinci tentang konflik gabungan

`describe-merge-conflicts` Contoh berikut menentukan konflik gabungan untuk file bernama `readme.md` di cabang sumber tertentu dan cabang tujuan menggunakan strategi `THREE_WAY_MERGE`.

```
aws codecommit describe-merge-conflicts \
  --source-commit-specifier feature-randomizationfeature \
  --destination-commit-specifier main \
  --merge-option THREE_WAY_MERGE \
  --file-path readme.md \
  --repository-name MyDemoRepo
```

Output:

```
{
  "conflictMetadata": {
    "filePath": "readme.md",
    "fileSizes": {
      "source": 139,
      "destination": 230,
      "base": 85
    }
  }
}
```

```
    },
    "fileModes": {
      "source": "NORMAL",
      "destination": "NORMAL",
      "base": "NORMAL"
    },
    "objectTypes": {
      "source": "FILE",
      "destination": "FILE",
      "base": "FILE"
    },
    "numberOfConflicts": 1,
    "isBinaryFile": {
      "source": false,
      "destination": false,
      "base": false
    },
    "contentConflict": true,
    "fileModeConflict": false,
    "objectTypeConflict": false,
    "mergeOperations": {
      "source": "M",
      "destination": "M"
    }
  },
  "mergeHunks": [
    {
      "isConflict": true,
      "source": {
        "startLine": 0,
        "endLine": 3,
        "hunkContent": "VGhpcyBpEXAMPLE="
      },
      "destination": {
        "startLine": 0,
        "endLine": 1,
        "hunkContent": "VXNlIHRoEXAMPLE="
      }
    }
  ],
  "destinationCommitId": "86958e0aEXAMPLE",
  "sourceCommitId": "6ccd57fdEXAMPLE",
  "baseCommitId": "767b69580EXAMPLE"
```

```
}
```

Untuk informasi selengkapnya, lihat [Menyelesaikan Konflik dalam Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [DescribeMergeConflicts](#) di Referensi AWS CLI Perintah.

## describe-pull-request-events

Contoh kode berikut menunjukkan cara menggunakan `describe-pull-request-events`.

### AWS CLI

Untuk melihat peristiwa dalam permintaan tarik

`describe-pull-request-events` Contoh berikut mengambil peristiwa untuk permintaan tarik dengan ID '8'.

```
aws codecommit describe-pull-request-events --pull-request-id 8
```

Output:

```
{
  "pullRequestEvents": [
    {
      "pullRequestId": "8",
      "pullRequestEventType": "PULL_REQUEST_CREATED",
      "eventDate": 1510341779.53,
      "actor": "arn:aws:iam::111111111111:user/Zhang_Wei"
    },
    {
      "pullRequestStatusChangedEventMetadata": {
        "pullRequestStatus": "CLOSED"
      },
      "pullRequestId": "8",
      "pullRequestEventType": "PULL_REQUEST_STATUS_CHANGED",
      "eventDate": 1510341930.72,
      "actor": "arn:aws:iam::111111111111:user/Jane_Doe"
    }
  ]
}
```

- Untuk detail API, lihat [DescribePullRequestEvents](#) di Referensi AWS CLI Perintah.

## **disassociate-approval-rule-template-from-repository**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-approval-rule-template-from-repository`.

### AWS CLI

Untuk memisahkan template aturan persetujuan dari repositori

`disassociate-approval-rule-template-from-repository` Contoh berikut memisahkan template aturan persetujuan yang ditentukan dari repositori bernama `MyDemoRepo`

```
aws codecommit disassociate-approval-rule-template-from-repository \  
  --repository-name MyDemoRepo \  
  --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memutuskan Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [DisassociateApprovalRuleTemplateFromRepository](#) di Referensi AWS CLI Perintah.

## **evaluate-pull-request-approval-rules**

Contoh kode berikut menunjukkan cara menggunakan `evaluate-pull-request-approval-rules`.

### AWS CLI

Untuk mengevaluasi apakah permintaan tarik telah memenuhi semua aturan persetujuannya

`evaluate-pull-request-approval-rules` Contoh berikut mengevaluasi status aturan persetujuan pada permintaan tarik yang ditentukan. Dalam contoh ini, aturan persetujuan belum dipenuhi untuk permintaan tarik, sehingga output dari perintah menunjukkan `approved` nilai `false`.

```
aws codecommit evaluate-pull-request-approval-rules \  
  --pull-request-id 1-approver-rule-for-all-pull-requests
```

```
--pull-request-id 27 \
--revision-id 9f29d167EXAMPLE
```

**Output:**

```
{
  "evaluation": {
    "approved": false,
    "approvalRulesNotSatisfied": [
      "Require two approved approvers"
    ],
    "overridden": false,
    "approvalRulesSatisfied": []
  }
}
```

Untuk informasi selengkapnya, lihat [Menggabungkan Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [EvaluatePullRequestApprovalRules](#) di Referensi AWS CLI Perintah.

**get-approval-rule-template**

Contoh kode berikut menunjukkan cara menggunakan `get-approval-rule-template`.

**AWS CLI**

Untuk mendapatkan konten template aturan persetujuan

`get-approval-rule-template` Contoh berikut mendapatkan isi dari template aturan persetujuan bernama `1-approver-rule-for-all-pull-requests`.

```
aws codecommit get-approval-rule-template \
--approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

**Output:**

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
    {\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [
    \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",

```

```
"ruleContentSha256": "621181bbEXAMPLE",
"lastModifiedDate": 1571356106.936,
"creationDate": 1571356106.936,
"approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
"lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan",
"approvalRuleTemplateId": "a29abb15-EXAMPLE",
"approvalRuleTemplateDescription": "All pull requests must be approved by
one developer on the team."
}
}
```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [GetApprovalRuleTemplate](#) di Referensi AWS CLI Perintah.

## get-blob

Contoh kode berikut menunjukkan cara menggunakan `get-blob`.

### AWS CLI

Untuk melihat informasi tentang objek gumpalan Git

`get-blob` Contoh berikut mengambil informasi tentang gumpalan Git dengan ID '2eb4AF3BEXample' dalam repositori bernama ". AWS CodeCommit MyDemoRepo

```
aws codecommit get-blob --repository-name MyDemoRepo --blob-id 2eb4af3bEXAMPLE
```

Output:

```
{
  "content": "QSBcCaW5hcnkgTGFyToEXAMPLE="
}
```

- Untuk detail API, lihat [GetBlob](#) di Referensi AWS CLI Perintah.

## get-branch

Contoh kode berikut menunjukkan cara menggunakan `get-branch`.

## AWS CLI

Untuk mendapatkan informasi tentang cabang

Contoh ini mendapatkan informasi tentang cabang dalam AWS CodeCommit repositori.

Perintah:

```
aws codecommit get-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Output:

```
{
  "BranchInfo": {
    "commitID": "317f8570EXAMPLE",
    "branchName": "MyNewBranch"
  }
}
```

- Untuk detail API, lihat [GetBranch](#) di Referensi AWS CLI Perintah.

## get-comment-reactions

Contoh kode berikut menunjukkan cara menggunakan `get-comment-reactions`.

## AWS CLI

Untuk melihat reaksi emoji terhadap komentar

`get-comment-reactions` Contoh berikut mencantumkan semua reaksi emoji terhadap komentar dengan ID dari `abcd1234EXAMPLEb5678efgh`. Jika font untuk shell Anda mendukung tampilan Emoji Versi 1.0, maka dalam output untuk emoji emoji ditampilkan.

```
aws codecommit get-comment-reactions \
  --comment-id abcd1234EXAMPLEb5678efgh
```

Output:

```
{
  "reactionsForComment": {
    [
      {
```



```

    "reaction": {
      "emoji": "??",
      "shortCode": "thumbsup",
      "unicode": "U+1F44D"
    },
    "users": [
      "arn:aws:iam::123456789012:user/Li_Juan",
      "arn:aws:iam::123456789012:user/Mary_Major",
      "arn:aws:iam::123456789012:user/Jorge_Souza"
    ]
  },
  {
    "reaction": {
      "emoji": "??",
      "shortCode": "thumbsdown",
      "unicode": "U+1F44E"
    },
    "users": [
      "arn:aws:iam::123456789012:user/Nikhil_Jayashankar"
    ]
  },
  {
    "reaction": {
      "emoji": "??",
      "shortCode": "confused",
      "unicode": "U+1F615"
    },
    "users": [
      "arn:aws:iam::123456789012:user/Saanvi_Sarkar"
    ]
  }
]
}
}
}

```

Untuk informasi selengkapnya, lihat [Mengomentari komit AWS CodeCommit di](#) Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [GetCommentReactions](#) di Referensi AWS CLI Perintah.

## get-comment

Contoh kode berikut menunjukkan cara menggunakan `get-comment`.

## AWS CLI

Untuk melihat detail komentar

Contoh ini menunjukkan cara melihat detail komentar dengan ID komentar yang dihasilkan sistem dari. `ff30b348EXAMPLEb9aa670f`

```
aws codecommit get-comment \  
  --comment-id ff30b348EXAMPLEb9aa670f
```

Output:

```
{  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "123Example",  
    "commentId": "ff30b348EXAMPLEb9aa670f",  
    "content": "Whoops - I meant to add this comment to the line, but I don't  
see how to delete it.",  
    "creationDate": 1508369768.142,  
    "deleted": false,  
    "commentId": "",  
    "lastModifiedDate": 1508369842.278,  
    "callerReactions": [],  
    "reactionCounts":  
    {  
      "SMILE" : 6,  
      "THUMBSUP" : 1  
    }  
  }  
}
```

- Untuk detail API, lihat [GetComment](#) di Referensi AWS CLI Perintah.

### **get-comments-for-compared-commit**

Contoh kode berikut menunjukkan cara menggunakan `get-comments-for-compared-commit`.

## AWS CLI

Untuk melihat komentar pada komit

Contoh ini menunjukkan cara melihat komentar tampilan yang dibuat pada perbandingan antara dua komit dalam repositori bernama. MyDemoRepo

```
aws codecommit get-comments-for-compared-commit \  
--repository-name MyDemoRepo \  
--before-commit-ID 6e147360EXAMPLE \  
--after-commit-id 317f8570EXAMPLE
```

Output:

```
{  
  "commentsForComparedCommitData": [  
    {  
      "afterBlobId": "1f330709EXAMPLE",  
      "afterCommitId": "317f8570EXAMPLE",  
      "beforeBlobId": "80906a4cEXAMPLE",  
      "beforeCommitId": "6e147360EXAMPLE",  
      "comments": [  
        {  
          "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
          "clientRequestToken": "123Example",  
          "commentId": "ff30b348EXAMPLEb9aa670f",  
          "content": "Whoops - I meant to add this comment to the line,  
not the file, but I don't see how to delete it.",  
          "creationDate": 1508369768.142,  
          "deleted": false,  
          "CommentId": "123abc-EXAMPLE",  
          "lastModifiedDate": 1508369842.278,  
          "callerReactions": [],  
          "reactionCounts":  
            {  
              "SMILE" : 6,  
              "THUMBSUP" : 1  
            }  
        },  
        {  
          "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
          "clientRequestToken": "123Example",  
          "commentId": "553b509bEXAMPLE56198325",  
          "content": "Can you add a test case for this?",  
          "creationDate": 1508369612.240,  
          "deleted": false,  
          "commentId": "456def-EXAMPLE",
```

```

        "lastModifiedDate": 1508369612.240,
        "callerReactions": [],
        "reactionCounts":
        {
            "THUMBSUP" : 2
        }
    ],
    "location": {
        "filePath": "cl_sample.js",
        "filePosition": 1232,
        "relativeFileVersion": "after"
    },
    "repositoryName": "MyDemoRepo"
}
],
"nextToken": "exampleToken"
}

```

- Untuk detail API, lihat [GetCommentsForComparedCommit](#) di Referensi AWS CLI Perintah.

## get-comments-for-pull-request

Contoh kode berikut menunjukkan cara menggunakan `get-comments-for-pull-request`.

### AWS CLI

Untuk melihat komentar untuk permintaan tarik

Contoh ini menunjukkan cara melihat komentar untuk permintaan tarik dalam repositori bernama `MyDemoRepo`

```

aws codecommit get-comments-for-pull-request \
  --repository-name MyDemoRepo \
  --before-commit-ID 317f8570EXAMPLE \
  --after-commit-id 5d036259EXAMPLE

```

Output:

```

{
  "commentsForPullRequestData": [
    {

```

```
    "afterBlobId": "1f330709EXAMPLE",
    "afterCommitId": "5d036259EXAMPLE",
    "beforeBlobId": "80906a4cEXAMPLE",
    "beforeCommitId": "317f8570EXAMPLE",
    "comments": [
      {
        "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
        "clientRequestToken": "",
        "commentId": "abcd1234EXAMPLEb5678efgh",
        "content": "These don't appear to be used anywhere. Can we
remove them?",
        "creationDate": 1508369622.123,
        "deleted": false,
        "lastModifiedDate": 1508369622.123,
        "callerReactions": [],
        "reactionCounts":
        {
          "THUMBSUP" : 6,
          "CONFUSED" : 1
        }
      },
      {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "",
        "commentId": "442b498bEXAMPLE5756813",
        "content": "Good catch. I'll remove them.",
        "creationDate": 1508369829.104,
        "deleted": false,
        "lastModifiedDate": 150836912.273,
        "callerReactions": ["THUMBSUP"]
        "reactionCounts":
        {
          "THUMBSUP" : 14
        }
      }
    ],
    "location": {
      "filePath": "ahs_count.py",
      "filePosition": 367,
      "relativeFileVersion": "AFTER"
    },
    "repositoryName": "MyDemoRepo",
    "pullRequestId": "42"
  }
```

```
  ],
  "nextToken": "exampleToken"
}
```

- Untuk detail API, lihat [GetCommentsForPullRequest](#) di Referensi AWS CLI Perintah.

## get-commit

Contoh kode berikut menunjukkan cara menggunakan `get-commit`.

### AWS CLI

Untuk melihat informasi tentang komit dalam repositori

Contoh ini menunjukkan detail tentang komit dengan ID yang dihasilkan sistem dari '7e9fd3091thisisanexamplethisisanexample1' dalam repositori bernama ". AWS CodeCommit MyDemoRepo

Perintah:

```
aws codecommit get-commit --repository-name MyDemoRepo --commit-id
7e9fd3091thisisanexamplethisisanexample1
```

Output:

```
{
  "commit": {
    "additionalData": "",
    "committer": {
      "date": "1484167798 -0800",
      "name": "Mary Major",
      "email": "mary_major@example.com"
    },
    "author": {
      "date": "1484167798 -0800",
      "name": "Mary Major",
      "email": "mary_major@example.com"
    },
    "treeId": "347a3408thisisanexampletreeidexample",
    "parents": [
      "7aa87a031thisisanexamplethisisanexample1"
    ]
  }
}
```

```

    ],
    "message": "Fix incorrect variable name"
  }
}

```

- Untuk detail API, lihat [GetCommit](#) di Referensi AWS CLI Perintah.

## get-differences

Contoh kode berikut menunjukkan cara menggunakan `get-differences`.

### AWS CLI

Untuk mendapatkan informasi tentang perbedaan untuk specifier komit dalam repositori

Contoh ini menunjukkan tampilan informasi metadata tentang perubahan antara dua penentu komit (cabang, tag, HEAD, atau referensi lain yang memenuhi syarat sepenuhnya, seperti ID komit) dalam folder yang diganti namanya dalam repositori bernama. AWS CodeCommit MyDemoRepo Contoh ini mencakup beberapa opsi yang tidak diperlukan, termasuk `--before-commit-specifier`, `--before-path`, dan `--after-path`, untuk lebih menggambarkan bagaimana Anda dapat menggunakan opsi ini untuk membatasi hasil. Respons termasuk izin mode file.

Perintah:

```

aws codecommit get-differences --repository-name MyDemoRepo --before-commit-specifier 955bba12thisisanexamplethisisanexample --after-commit-specifier 14a95463thisisanexamplethisisanexample --before-path tmp/example-folder --after-path tmp/renamed-folder

```

Output:

```

{
  "differences": [
    {
      "afterBlob": {
        "path": "blob.txt",
        "blobId": "2eb4af3b1thisisanexamplethisisanexample1",
        "mode": "100644"
      },
      "changeType": "M",
      "beforeBlob": {

```

```

        "path": "blob.txt",
        "blobId": "bf7fcf281thisisanexamplethisisanexample1",
        "mode": "100644"
    }
}
]
}

```

- Untuk detail API, lihat [GetDifferences](#) di Referensi AWS CLI Perintah.

## get-file

Contoh kode berikut menunjukkan cara menggunakan `get-file`.

### AWS CLI

Untuk mendapatkan isi berkas yang dikodekan basis-64 dalam sebuah repositori AWS CodeCommit

`get-file` Contoh berikut menunjukkan bagaimana untuk mendapatkan isi basis-64 dikodekan dari file bernama `README.md` dari cabang bernama dalam repositori bernama `main`. `MyDemoRepo`

```

aws codecommit get-file \
  --repository-name MyDemoRepo \
  --commit-specifier main \
  --file-path README.md

```

Output:

```

{
  "blobId": "559b44fEXAMPLE",
  "commitId": "c5709475EXAMPLE",
  "fileContent": "IyBQaHVzEXAMPLE",
  "filePath": "README.md",
  "fileMode": "NORMAL",
  "fileSize": 1563
}

```

Untuk informasi selengkapnya, lihat [GetFile](#) di panduan Referensi AWS CodeCommit API.

- Untuk detail API, lihat [GetFile](#) di Referensi AWS CLI Perintah.



## get-folder

Contoh kode berikut menunjukkan cara menggunakan `get-folder`.

### AWS CLI

Untuk mendapatkan isi folder dalam AWS CodeCommit repositori

`get-folder` Contoh berikut menunjukkan bagaimana untuk mendapatkan isi dari folder tingkat atas dari repositori bernama. `MyDemoRepo`

```
aws codecommit get-folder --repository-name MyDemoRepo --folder-path ""
```

Output:

```
{
  "commitId": "c5709475EXAMPLE",
  "files": [
    {
      "absolutePath": ".gitignore",
      "blobId": "74094e8bEXAMPLE",
      "fileMode": "NORMAL",
      "relativePath": ".gitignore"
    },
    {
      "absolutePath": "Gemfile",
      "blobId": "9ceb72f6EXAMPLE",
      "fileMode": "NORMAL",
      "relativePath": "Gemfile"
    },
    {
      "absolutePath": "Gemfile.lock",
      "blobId": "795c4a2aEXAMPLE",
      "fileMode": "NORMAL",
      "relativePath": "Gemfile.lock"
    },
    {
      "absolutePath": "LICENSE.txt",
      "blobId": "0c7932c8EXAMPLE",
      "fileMode": "NORMAL",
      "relativePath": "LICENSE.txt"
    },
    {
```

```

        "absolutePath":"README.md",
        "blobId":"559b44feEXAMPLE",
        "fileMode":"NORMAL",
        "relativePath":"README.md"
    }
],
"folderPath":"",
"subFolders":[
    {
        "absolutePath":"public",
        "relativePath":"public",
        "treeId":"d5e92ae3aEXAMPLE"
    },
    {
        "absolutePath":"tmp",
        "relativePath":"tmp",
        "treeId":"d564d0bcEXAMPLE"
    }
],
"subModules":[],
"symbolicLinks":[],
"treeId":"7b3c4dadEXAMPLE"
}

```

Untuk informasi selengkapnya, lihat `GetFolder` di panduan Referensi AWS CodeCommit API.

- Untuk detail API, lihat [GetFolder](#) di Referensi AWS CLI Perintah.

## get-merge-commit

Contoh kode berikut menunjukkan cara menggunakan `get-merge-commit`.

### AWS CLI

Untuk mendapatkan informasi rinci tentang komit gabungan

`get-merge-commit` Contoh berikut menampilkan rincian tentang komit gabungan untuk cabang sumber bernama `bugfix-bug1234` dengan cabang tujuan bernama `main` menggunakan strategi `THREE_WAY_MERGE` dalam repositori bernama `MyDemoRepo`

```
aws codecommit get-merge-commit \
  --source-commit-specifier bugfix-bug1234 \
```

```
--destination-commit-specifier main \  
--merge-option THREE_WAY_MERGE \  
--repository-name MyDemoRepo
```

Output:

```
{  
  "sourceCommitId": "c5709475EXAMPLE",  
  "destinationCommitId": "317f8570EXAMPLE",  
  "baseCommitId": "fb12a539EXAMPLE",  
  "mergeCommitId": "ffc4d608eEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Komit](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [GetMergeCommit](#) di Referensi AWS CLI Perintah.

## get-merge-conflicts

Contoh kode berikut menunjukkan cara menggunakan `get-merge-conflicts`.

### AWS CLI

Untuk melihat apakah ada konflik gabungan untuk permintaan tarik

`get-merge-conflicts` Contoh berikut menampilkan apakah ada konflik gabungan antara ujung cabang sumber bernama `feature-randomizationfeature` dan cabang tujuan bernama `'main'` dalam repositori bernama `MyDemoRepo`

```
aws codecommit get-merge-conflicts \  
  --repository-name MyDemoRepo \  
  --source-commit-specifier feature-randomizationfeature \  
  --destination-commit-specifier main \  
  --merge-option THREE_WAY_MERGE
```

Output:

```
{  
  "mergeable": false,
```

```

"destinationCommitId": "86958e0aEXAMPLE",
"sourceCommitId": "6ccd57fdEXAMPLE",
"baseCommitId": "767b6958EXAMPLE",
"conflictMetadataList": [
  {
    "filePath": "readme.md",
    "fileSizes": {
      "source": 139,
      "destination": 230,
      "base": 85
    },
    "fileModes": {
      "source": "NORMAL",
      "destination": "NORMAL",
      "base": "NORMAL"
    },
    "objectTypes": {
      "source": "FILE",
      "destination": "FILE",
      "base": "FILE"
    },
    "numberOfConflicts": 1,
    "isBinaryFile": {
      "source": false,
      "destination": false,
      "base": false
    },
    "contentConflict": true,
    "fileModeConflict": false,
    "objectTypeConflict": false,
    "mergeOperations": {
      "source": "M",
      "destination": "M"
    }
  }
]
}

```

- Untuk detail API, lihat [GetMergeConflicts](#) di Referensi AWS CLI Perintah.

## get-merge-options

Contoh kode berikut menunjukkan cara menggunakan `get-merge-options`.

## AWS CLI

Untuk mendapatkan informasi tentang opsi penggabungan yang tersedia untuk menggabungkan dua cabang yang ditentukan

`get-merge-options` Contoh berikut menentukan opsi gabungan yang tersedia untuk menggabungkan cabang sumber bernama `bugfix-bug1234` dengan cabang tujuan bernama `main` dalam repositori bernama `MyDemoRepo`

```
aws codecommit get-merge-options \
  --source-commit-specifier bugfix-bug1234 \
  --destination-commit-specifier main \
  --repository-name MyDemoRepo
```

Output:

```
{
  "mergeOptions": [
    "FAST_FORWARD_MERGE",
    "SQUASH_MERGE",
    "THREE_WAY_MERGE"
  ],
  "sourceCommitId": "18059494EXAMPLE",
  "destinationCommitId": "ffd3311dEXAMPLE",
  "baseCommitId": "ffd3311dEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menyelesaikan Konflik dalam Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [GetMergeOptions](#) di Referensi AWS CLI Perintah.

## **get-pull-request-approval-states**

Contoh kode berikut menunjukkan cara menggunakan `get-pull-request-approval-states`.

## AWS CLI

Untuk melihat persetujuan pada permintaan tarik

`get-pull-request-approval-states` Contoh berikut mengembalikan persetujuan untuk permintaan tarik tertentu.

```
aws codecommit get-pull-request-approval-states \  
  --pull-request-id 8 \  
  --revision-id 9f29d167EXAMPLE
```

Output:

```
{  
  "approvals": [  
    {  
      "userArn": "arn:aws:iam::123456789012:user/Mary_Major",  
      "approvalState": "APPROVE"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [GetPullRequestApprovalStates](#) di Referensi AWS CLI Perintah.

## get-pull-request-override-state

Contoh kode berikut menunjukkan cara menggunakan `get-pull-request-override-state`.

AWS CLI

Untuk mendapatkan informasi tentang status penggantian permintaan tarik

`get-pull-request-override-state` Contoh berikut mengembalikan state override untuk pull request yang ditentukan. Dalam contoh ini, aturan persetujuan untuk permintaan tarik diganti oleh pengguna bernama Mary Major, sehingga output mengembalikan nilai `true` :

```
aws codecommit get-pull-request-override-state \  
  --pull-request-id 34 \  
  --revision-id 9f29d167EXAMPLE
```

Output:

```
{  
  "overridden": true,
```

```
"override": "arn:aws:iam::123456789012:user/Mary_Major"
}
```

Untuk informasi selengkapnya, lihat [Mengganti Aturan Persetujuan pada Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [GetPullRequestOverrideState](#) di Referensi AWS CLI Perintah.

## get-pull-request

Contoh kode berikut menunjukkan cara menggunakan `get-pull-request`.

### AWS CLI

Untuk melihat detail permintaan tarik

Contoh ini menunjukkan cara melihat informasi tentang permintaan tarik dengan ID. 27

```
aws codecommit get-pull-request \
  --pull-request-id 27
```

Output:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
          [
            {
              \"Type\": \"Approvers\",
              \"NumberOfApprovalsNeeded\": 2,
              \"ApprovalPoolMembers\": [
                \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"
              ]
            }
          ]
        }\",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "lastActivityDate": 1562619583.565,
    "pullRequestTargets": [
      {
        "sourceCommit": "ca45e279EXAMPLE",
```

```

        "sourceReference": "refs/heads/bugfix-1234",
        "mergeBase": "a99f5ddbEXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
            "isMerged": false
        },
        "destinationCommit": "2abfc6beEXAMPLE",
        "repositoryName": "MyDemoRepo"
    }
],
"revisionId": "e47def21EXAMPLE",
"title": "Quick fix for bug 1234",
"authorArn": "arn:aws:iam::123456789012:user/Nikhil_Jayashankar",
"clientRequestToken": "d8d7612e-EXAMPLE",
"creationDate": 1562619583.565,
"pullRequestId": "27",
"pullRequestStatus": "OPEN"
}
}

```

- Untuk detail API, lihat [GetPullRequest](#) di Referensi AWS CLI Perintah.

## get-repository-triggers

Contoh kode berikut menunjukkan cara menggunakan `get-repository-triggers`.

### AWS CLI

Untuk mendapatkan informasi tentang pemacu di repositori

Contoh ini menunjukkan detail tentang pemacu yang dikonfigurasi untuk AWS CodeCommit repositori bernama `MyDemoRepo`

```
aws codecommit get-repository-triggers \
  --repository-name MyDemoRepo
```

### Output:

```
{
  "configurationId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
  "triggers": [
    {
```



```

        "destinationArn": "arn:aws:sns:us-
east-1:111111111111:MyCodeCommitTopic",
        "branches": [
            "main",
            "preprod"
        ],
        "name": "MyFirstTrigger",
        "customData": "",
        "events": [
            "all"
        ]
    },
    {
        "destinationArn": "arn:aws:lambda:us-
east-1:111111111111:function:MyCodeCommitPythonFunction",
        "branches": [],
        "name": "MySecondTrigger",
        "customData": "EXAMPLE",
        "events": [
            "all"
        ]
    }
]
}

```

- Untuk detail API, lihat [GetRepositoryTriggers](#) di Referensi AWS CLI Perintah.

## get-repository

Contoh kode berikut menunjukkan cara menggunakan `get-repository`.

### AWS CLI

Untuk mendapatkan informasi tentang repositori

Contoh ini menunjukkan rincian tentang AWS CodeCommit repositori.

```
aws codecommit get-repository \
  --repository-name MyDemoRepo
```

Output:

```
{
```

```

    "repositoryMetadata": {
      "creationDate": 1429203623.625,
      "defaultBranch": "main",
      "repositoryName": "MyDemoRepo",
      "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/v1/
repos/MyDemoRepo",
      "lastModifiedDate": 1430783812.0869999,
      "repositoryDescription": "My demonstration repository",
      "cloneUrlHttp": "https://codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
      "Arn": "arn:aws:codecommit:us-east-1:80398EXAMPLE:MyDemoRepo",
      "accountId": "111111111111"
    }
  }
}

```

- Untuk detail API, lihat [GetRepository](#) di Referensi AWS CLI Perintah.

## list-approval-rule-templates

Contoh kode berikut menunjukkan cara menggunakan `list-approval-rule-templates`.

### AWS CLI

Untuk mencantumkan semua templat aturan persetujuan di AWS Wilayah

`list-approval-rule-templates` Contoh berikut mencantumkan semua templat aturan persetujuan di Wilayah tertentu. Jika tidak ada AWS Wilayah yang ditentukan sebagai parameter, perintah mengembalikan templat aturan persetujuan untuk wilayah yang ditentukan dalam profil AWS CLI yang digunakan untuk menjalankan perintah.

```
aws codecommit list-approval-rule-templates \
  --region us-east-2
```

Output:

```
{
  "approvalRuleTemplateName": [
    "2-approver-rule-for-main",
    "1-approver-rule-for-all-pull-requests"
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [ListApprovalRuleTemplates](#) di Referensi AWS CLI Perintah.

## **list-associated-approval-rule-templates-for-repository**

Contoh kode berikut menunjukkan cara menggunakan `list-associated-approval-rule-templates-for-repository`.

### AWS CLI

Untuk daftar semua template yang terkait dengan repositori

`list-associated-approval-rule-templates-for-repository` Contoh berikut mencantumkan semua template aturan persetujuan yang terkait dengan repositori bernama `MyDemoRepo`

```
aws codecommit list-associated-approval-rule-templates-for-repository \  
  --repository-name MyDemoRepo
```

### Output:

```
{  
  "approvalRuleTemplateName": [  
    "2-approver-rule-for-main",  
    "1-approver-rule-for-all-pull-requests"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [ListAssociatedApprovalRuleTemplatesForRepository](#) di Referensi AWS CLI Perintah.

## **list-branches**

Contoh kode berikut menunjukkan cara menggunakan `list-branches`.

## AWS CLI

Untuk melihat daftar nama cabang

Contoh ini mencantumkan semua nama cabang dalam AWS CodeCommit repositori.

```
aws codecommit list-branches \  
  --repository-name MyDemoRepo
```

Output:

```
{  
  "branches": [  
    "MyNewBranch",  
    "main"  
  ]  
}
```

- Untuk detail API, lihat [ListBranches](#) di Referensi AWS CLI Perintah.

## list-pull-requests

Contoh kode berikut menunjukkan cara menggunakan `list-pull-requests`.

## AWS CLI

Untuk melihat daftar permintaan tarik dalam repositori

Contoh ini menunjukkan cara membuat daftar permintaan tarik yang dibuat oleh pengguna IAM dengan ARN 'arn:aws:iam: :111111111111:user/LI\_JUAN' dan status 'CLOSED' di repositori bernama "": AWS CodeCommit MyDemoRepo

```
aws codecommit list-pull-requests --author-arn arn:aws:iam::111111111111:user/  
Li_Juan --pull-request-status CLOSED --repository-name MyDemoRepo
```

Output:

```
{  
  "nextToken": "",  
  "pullRequestIds": ["2", "12", "16", "22", "23", "35", "30", "39", "47"]  
}
```

- Untuk detail API, lihat [ListPullRequests](#) di Referensi AWS CLI Perintah.

## **list-repositories-for-approval-rule-template**

Contoh kode berikut menunjukkan cara menggunakan `list-repositories-for-approval-rule-template`.

### AWS CLI

Untuk daftar semua repositori yang terkait dengan template

`list-repositories-for-approval-rule-template` Contoh berikut mencantumkan semua repositori yang terkait dengan template aturan persetujuan yang ditentukan.

```
aws codecommit list-repositories-for-approval-rule-template \
  --approval-rule-template-name 2-approver-rule-for-main
```

Output:

```
{
  "repositoryNames": [
    "MyDemoRepo",
    "MyClonedRepo"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [ListRepositoriesForApprovalRuleTemplate](#) di Referensi AWS CLI Perintah.

## **list-repositories**

Contoh kode berikut menunjukkan cara menggunakan `list-repositories`.

### AWS CLI

Untuk melihat daftar repositori

Contoh ini mencantumkan semua AWS CodeCommit repositori yang terkait dengan akun pengguna. AWS

**Perintah:**

```
aws codecommit list-repositories
```

**Output:**

```
{
  "repositories": [
    {
      "repositoryName": "MyDemoRepo"
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    },
    {
      "repositoryName": "MyOtherDemoRepo"
      "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE"
    }
  ]
}
```

- Untuk detail API, lihat [ListRepositories](#) di Referensi AWS CLI Perintah.

**list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

**AWS CLI**

Untuk melihat AWS tag untuk repositori

`list-tags-for-resource` Contoh berikut mencantumkan kunci tag dan nilai tag untuk repositori tertentu.

```
aws codecommit list-tags-for-resource \
  --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo
```

**Output:**

```
{
  "tags": {
    "Status": "Secret",
    "Team": "Saanvi"
  }
}
```

```
}
```

Untuk informasi selengkapnya, [lihat Melihat Tag untuk Repositori](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **merge-branches-by-fast-forward**

Contoh kode berikut menunjukkan cara menggunakan `merge-branches-by-fast-forward`.

### AWS CLI

Untuk menggabungkan dua cabang menggunakan strategi penggabungan maju cepat

`merge-branches-by-fast-forward` Contoh berikut menggabungkan cabang sumber yang ditentukan dengan cabang tujuan tertentu dalam repositori bernama `MyDemoRepo`

```
aws codecommit merge-branches-by-fast-forward \  
  --source-commit-specifier bugfix-bug1234 \  
  --destination-commit-specifier bugfix-bug1233 \  
  --repository-name MyDemoRepo
```

Output:

```
{  
  "commitId": "4f178133EXAMPLE",  
  "treeId": "389765daEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Membandingkan dan Menggabungkan Cabang](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [MergeBranchesByFastForward](#) di Referensi AWS CLI Perintah.

## **merge-branches-by-squash**

Contoh kode berikut menunjukkan cara menggunakan `merge-branches-by-squash`.

### AWS CLI

Untuk menggabungkan dua cabang menggunakan strategi penggabungan squash

`merge-branches-by-squash` Contoh berikut menggabungkan cabang sumber yang ditentukan dengan cabang tujuan tertentu dalam repositori bernama. MyDemoRepo

```
aws codecommit merge-branches-by-squash \  
  --source-commit-specifier bugfix-bug1234 \  
  --destination-commit-specifier bugfix-bug1233 \  
  --author-name "Maria Garcia" \  
  --email "maria_garcia@example.com" \  
  --commit-message "Merging two fix branches to prepare for a general patch." \  
  --repository-name MyDemoRepo
```

Output:

```
{  
  "commitId": "4f178133EXAMPLE",  
  "treeId": "389765daEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Membandingkan dan Menggabungkan Cabang](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [MergeBranchesBySquash](#) di Referensi AWS CLI Perintah.

## **merge-branches-by-three-way**

Contoh kode berikut menunjukkan cara menggunakan `merge-branches-by-three-way`.

AWS CLI

Untuk menggabungkan dua cabang menggunakan strategi penggabungan tiga arah

`merge-branches-by-three-way` Contoh berikut menggabungkan cabang sumber yang ditentukan dengan cabang tujuan tertentu dalam repositori bernama. MyDemoRepo

```
aws codecommit merge-branches-by-three-way \  
  --source-commit-specifier main \  
  --destination-commit-specifier bugfix-bug1234 \  
  --author-name "Jorge Souza" --email "jorge_souza@example.com" \  
  --commit-message "Merging changes from main to bugfix branch before additional  
testing." \  
  --repository-name MyDemoRepo
```



**Output:**

```
{
  "commitId": "4f178133EXAMPLE",
  "treeId": "389765daEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Membandingkan dan Menggabungkan Cabang](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [MergeBranchesByThreeWay](#) di Referensi AWS CLI Perintah.

**merge-pull-request-by-fast-forward**

Contoh kode berikut menunjukkan cara menggunakan `merge-pull-request-by-fast-forward`.

**AWS CLI**

Untuk menggabungkan dan menutup permintaan tarik

Contoh ini menunjukkan cara menggabungkan dan menutup permintaan tarik dengan ID '47' dan ID komit sumber '99132AB0Example' dalam repositori bernama. MyDemoRepo

```
aws codecommit merge-pull-request-by-fast-forward \
  --pull-request-id 47 \
  --source-commit-id 99132ab0EXAMPLE \
  --repository-name MyDemoRepo
```

**Output:**

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
          {\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [
            {\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"}]}]}\",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "I want one approver for this pull request",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
      }
    ]
  }
}
```

```

        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
    }
],
"authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
"clientRequestToken": "",
"creationDate": 1508530823.142,
"description": "Review the latest changes and updates to the global
variables",
"lastActivityDate": 1508887223.155,
"pullRequestId": "47",
"pullRequestStatus": "CLOSED",
"pullRequestTargets": [
    {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
            "isMerged": true,
            "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
    }
],
"title": "Consolidation of global variables"
}
}

```

Untuk informasi selengkapnya, lihat [Menggabungkan Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [MergePullRequestByFastForward](#) di Referensi AWS CLI Perintah.

## merge-pull-request-by-squash

Contoh kode berikut menunjukkan cara menggunakan `merge-pull-request-by-squash`.

### AWS CLI

Untuk menggabungkan permintaan tarik menggunakan strategi penggabungan squash

`merge-pull-request-by-squash` Contoh berikut menggabungkan dan menutup permintaan tarik yang ditentukan menggunakan strategi resolusi konflik `ACCEPT_SOURCE` dalam repositori bernama `MyDemoRepo`

```
aws codecommit merge-pull-request-by-squash \
  --pull-request-id 47 \
  --source-commit-id 99132ab0EXAMPLE \
  --repository-name MyDemoRepo \
  --conflict-detail-level LINE_LEVEL \
  --conflict-resolution-strategy ACCEPT_SOURCE \
  --name "Jorge Souza" --email "jorge_souza@example.com" \
  --commit-message "Merging pull request 47 by squash and accepting source in
merge conflicts"
```

Output:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.142,
    "description": "Review the latest changes and updates to the global
variables",
    "lastActivityDate": 1508887223.155,
    "pullRequestId": "47",
```

```

    "pullRequestStatus": "CLOSED",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": true,
          "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ],
    "title": "Consolidation of global variables"
  }
}

```

Untuk informasi selengkapnya, lihat [Menggabungkan Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [MergePullRequestBySquash](#) di Referensi AWS CLI Perintah.

## merge-pull-request-by-three-way

Contoh kode berikut menunjukkan cara menggunakan `merge-pull-request-by-three-way`.

### AWS CLI

Untuk menggabungkan permintaan tarik menggunakan strategi penggabungan tiga arah

`merge-pull-request-by-three-way` Contoh berikut menggabungkan dan menutup permintaan tarik yang ditentukan menggunakan opsi default untuk detail konflik dan strategi resolusi konflik dalam repositori bernama. `MyDemoRepo`

```

aws codecommit merge-pull-request-by-three-way \
  --pull-request-id 47 \
  --source-commit-id 99132ab0EXAMPLE \
  --repository-name MyDemoRepo \
  --name "Maria Garcia" \
  --email "maria_garcia@example.com" \
  --commit-message "Merging pull request 47 by three-way with default options"

```

## Output:

```

{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
        \"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
        \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
        [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.142,
    "description": "Review the latest changes and updates to the global
    variables",
    "lastActivityDate": 1508887223.155,
    "pullRequestId": "47",
    "pullRequestStatus": "CLOSED",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": true,
          "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ],
    "title": "Consolidation of global variables"
  }
}

```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Menggabungkan Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [MergePullRequestByThreeWay](#) di Referensi AWS CLI Perintah.

## **override-pull-request-approval-rules**

Contoh kode berikut menunjukkan cara menggunakan `override-pull-request-approval-rules`.

### AWS CLI

Untuk mengganti persyaratan aturan persetujuan pada permintaan tarik

`override-pull-request-approval-rules` Contoh berikut mengesampingkan aturan persetujuan pada permintaan tarik yang ditentukan. Untuk mencabut penggantian sebagai gantinya, setel nilai `--override-status` parameter ke. `REVOKE`

```
aws codecommit override-pull-request-approval-rules \  
  --pull-request-id 34 \  
  --revision-id 927df8d8EXAMPLE \  
  --override-status OVERRIDE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti Aturan Persetujuan pada Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [OverridePullRequestApprovalRules](#) di Referensi AWS CLI Perintah.

## **post-comment-for-compared-commit**

Contoh kode berikut menunjukkan cara menggunakan `post-comment-for-compared-commit`.

### AWS CLI

Untuk membuat komentar pada komit

Contoh ini menunjukkan cara menambahkan komentar "Can you add a test case for this?" pada perubahan ke `cl_sample.js` file dalam perbandingan antara dua komit dalam repositori bernama `MyDemoRepo`

```
aws codecommit post-comment-for-compared-commit \  
  --repository-name MyDemoRepo \  
  --before-commit-id 317f8570EXAMPLE \  
  --after-commit-id 5d036259EXAMPLE \  
  --client-request-token 123Example \  
  --content "Can you add a test case for this?" \  
  --location filePath=cl_sample.js,filePosition=1232,relativeFileVersion=AFTER
```

Output:

```
{  
  "afterBlobId": "1f330709EXAMPLE",  
  "afterCommitId": "317f8570EXAMPLE",  
  "beforeBlobId": "80906a4cEXAMPLE",  
  "beforeCommitId": "6e147360EXAMPLE",  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "",  
    "commentId": "553b509bEXAMPLE56198325",  
    "content": "Can you add a test case for this?",  
    "creationDate": 1508369612.203,  
    "deleted": false,  
    "commentId": "abc123-EXAMPLE",  
    "lastModifiedDate": 1508369612.203,  
    "callerReactions": [],  
    "reactionCounts": []  
  },  
  "location": {  
    "filePath": "cl_sample.js",  
    "filePosition": 1232,  
    "relativeFileVersion": "AFTER"  
  },  
  "repositoryName": "MyDemoRepo"  
}
```

- Untuk detail API, lihat [PostCommentForComparedCommit](#) di Referensi AWS CLI Perintah.

## post-comment-for-pull-request

Contoh kode berikut menunjukkan cara menggunakan `post-comment-for-pull-request`.

### AWS CLI

Untuk menambahkan komentar ke permintaan tarik

`post-comment-for-pull-request` Contoh berikut menambahkan komentar "Ini tampaknya tidak digunakan di mana pun. Bisakah kita menghapusnya?" pada perubahan ke `ahs_count.py` file dalam permintaan tarik dengan ID 47 dalam repositori bernama `MyDemoRepo`

```
aws codecommit post-comment-for-pull-request \
  --pull-request-id "47" \
  --repository-name MyDemoRepo \
  --before-commit-id 317f8570EXAMPLE \
  --after-commit-id 5d036259EXAMPLE \
  --client-request-token 123Example \
  --content "These don't appear to be used anywhere. Can we remove them?" \
  --location filePath=ahs_count.py,filePosition=367,relativeFileVersion=AFTER
```

Output:

```
{
  "afterBlobId": "1f330709EXAMPLE",
  "afterCommitId": "5d036259EXAMPLE",
  "beforeBlobId": "80906a4cEXAMPLE",
  "beforeCommitId": "317f8570EXAMPLE",
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
    "clientRequestToken": "123Example",
    "commentId": "abcd1234EXAMPLEb5678efgh",
    "content": "These don't appear to be used anywhere. Can we remove
them?",
    "creationDate": 1508369622.123,
    "deleted": false,
    "CommentId": "",
    "lastModifiedDate": 1508369622.123,
    "callerReactions": [],
    "reactionCounts": []
  },
  "location": {
    "filePath": "ahs_count.py",
```



```
        "filePosition": 367,  
        "relativeFileVersion": "AFTER"  
    },  
    "repositoryName": "MyDemoRepo",  
    "pullRequestId": "47"  
}
```

- Untuk detail API, lihat [PostCommentForPullRequest](#) di Referensi AWS CLI Perintah.

## post-comment-reply

Contoh kode berikut menunjukkan cara menggunakan `post-comment-reply`.

### AWS CLI

Untuk membalas komentar pada komit atau permintaan tarik

Contoh ini menunjukkan cara menambahkan balasan "Good catch. I'll remove them." ke komentar dengan ID yang dihasilkan sistem dari. `abcd1234EXAMPLEb5678efgh`

```
aws codecommit post-comment-reply \  
  --in-reply-to abcd1234EXAMPLEb5678efgh \  
  --content "Good catch. I'll remove them." \  
  --client-request-token 123Example
```

Output:

```
{  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "123Example",  
    "commentId": "442b498bEXAMPLE5756813",  
    "content": "Good catch. I'll remove them.",  
    "creationDate": 1508369829.136,  
    "deleted": false,  
    "CommentId": "abcd1234EXAMPLEb5678efgh",  
    "lastModifiedDate": 150836912.221,  
    "callerReactions": [],  
    "reactionCounts": []  
  }  
}
```

- Untuk detail API, lihat [PostCommentReply](#) di Referensi AWS CLI Perintah.

## put-comment-reaction

Contoh kode berikut menunjukkan cara menggunakan `put-comment-reaction`.

### AWS CLI

Untuk membalas komentar pada komit dengan emoji

`put-comment-reaction` Contoh berikut membalas komentar dengan ID dari `abcd1234EXAMPLEb5678efgh` dengan nilai reaksi emoji. `:thumbsup:`

```
aws codecommit put-comment-reaction \  
  --comment-id abcd1234EXAMPLEb5678efgh \  
  --reaction-value :thumbsup:
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengomentari komit AWS CodeCommit di Panduan AWS CodeCommit Pengguna](#).

- Untuk detail API, lihat [PutCommentReaction](#) di Referensi AWS CLI Perintah.

## put-file

Contoh kode berikut menunjukkan cara menggunakan `put-file`.

### AWS CLI

Untuk menambahkan file ke repositori

`put-file` Contoh berikut menambahkan file bernama 'ExampleSolution.py' ke repositori bernama 'ke cabang bernama' `feature-randomizationfeature` MyDemoRepo 'yang komit terbarunya memiliki ID '4C925148Example'.

```
aws codecommit put-file \  
  --repository-name MyDemoRepo \  
  --branch-name feature-randomizationfeature \  
  --file-content file://MyDirectory/ExampleSolution.py \  
  --file-path /solutions/ExampleSolution.py \  
  --file-permissions filemode::rwxr-xr-x
```

```
--parent-commit-id 4c925148EXAMPLE \  
--name "Maria Garcia" \  
--email "maria_garcia@example.com" \  
--commit-message "I added a third randomization routine."
```

Output:

```
{  
  "blobId": "2eb4af3bEXAMPLE",  
  "commitId": "317f8570EXAMPLE",  
  "treeId": "347a3408EXAMPLE"  
}
```

- Untuk detail API, lihat [PutFile](#) di Referensi AWS CLI Perintah.

## put-repository-triggers

Contoh kode berikut menunjukkan cara menggunakan `put-repository-triggers`.

### AWS CLI

Untuk menambah atau memperbarui pemicu di repositori

Contoh ini menunjukkan cara memperbarui pemicu bernama " dan 'MyFirstTriggerMySecondTrigger' menggunakan file JSON yang sudah dibuat (di sini bernama MyTriggers .json) yang berisi struktur semua pemicu untuk repositori bernama MyDemoRepo. Untuk mempelajari cara mendapatkan JSON untuk pemicu yang ada, lihat perintahnya. `get-repository-triggers`

```
aws codecommit put-repository-triggers \  
  --repository-name MyDemoRepo file://MyTriggers.json
```

Isi dari MyTriggers .json:

```
{  
  "repositoryName": "MyDemoRepo",  
  "triggers": [  
    {  
      "destinationArn": "arn:aws:sns:us-  
east-1:80398EXAMPLE:MyCodeCommitTopic",  
      "branches": [  

```

```

        "main",
        "preprod"
    ],
    "name": "MyFirstTrigger",
    "customData": "",
    "events": [
        "all"
    ]
},
{
    "destinationArn": "arn:aws:lambda:us-
east-1:111111111111:function:MyCodeCommitPythonFunction",
    "branches": [],
    "name": "MySecondTrigger",
    "customData": "EXAMPLE",
    "events": [
        "all"
    ]
}
]
}

```

Output:

```

{
  "configurationId": "6fa51cd8-35c1-EXAMPLE"
}

```

- Untuk detail API, lihat [PutRepositoryTriggers](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan AWS tag ke repositori yang ada

`tag-resource` Contoh berikut tag repositori yang ditentukan dengan dua tag.

```

aws codecommit tag-resource \
  --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo \

```

```
--tags Status=Secret,Team=Saarvi
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan Tag ke Repositori](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## test-repository-triggers

Contoh kode berikut menunjukkan cara menggunakan `test-repository-triggers`.

### AWS CLI

Untuk menguji pemicu dalam repositori

Contoh ini menunjukkan bagaimana untuk menguji pemicu bernama 'MyFirstTrigger' dalam AWS CodeCommit repositori bernama MyDemoRepo. Dalam contoh ini, peristiwa di repositori memicu pemberitahuan dari topik Simple Notification Service Amazon (Amazon SNS).

Perintah:

```
aws codecommit test-repository-triggers --repository-name MyDemoRepo
--triggers name=MyFirstTrigger,destinationArn=arn:aws:sns:us-
east-1:111111111111:MyCodeCommitTopic,branches=mainline,preprod,events=all
```

Output:

```
{
  "successfulExecutions": [
    "MyFirstTrigger"
  ],
  "failedExecutions": []
}
```

- Untuk detail API, lihat [TestRepositoryTriggers](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus AWS tag dari repositori

`untag-resource` Contoh berikut menghapus tag dengan kunci tertentu dari repositori bernama `MyDemoRepo`

```
aws codecommit untag-resource \  
  --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo \  
  --tag-keys Status
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Tag dari Repositori](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-approval-rule-template-content

Contoh kode berikut menunjukkan cara menggunakan `update-approval-rule-template-content`.

### AWS CLI

Untuk memperbarui konten templat aturan persetujuan

`update-approval-rule-template-content` Contoh berikut mengubah konten templat aturan persetujuan yang ditentukan untuk mendefinisikan ulang kumpulan persetujuan kepada pengguna yang berperan. `CodeCommitReview`

```
aws codecommit update-approval-rule-template-content \  
  --approval-rule-template-name 1-approver-rule \  
  --new-rule-content "{\"Version\": \"2018-11-08\", \"DestinationReferences\  
\": [\"refs/heads/main\"], \"Statements\": [{\"Type\": \"Approvers\  
\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":  
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Output:

```
{  
  "approvalRuleTemplate": {
```

```

    "creationDate": 1571352720.773,
    "approvalRuleTemplateDescription": "Requires 1 approval for all pull
requests from the CodeCommitReview pool",
    "lastModifiedDate": 1571358728.41,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\":
[{\ \"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\":
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "ruleContentSha256": "2f6c21a5EXAMPLE",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [UpdateApprovalRuleTemplateContent](#) di Referensi AWS CLI Perintah.

## update-approval-rule-template-description

Contoh kode berikut menunjukkan cara menggunakan `update-approval-rule-template-description`.

### AWS CLI

Untuk memperbarui deskripsi templat aturan persetujuan

`update-approval-rule-template-description` Contoh berikut mengubah deskripsi template aturan persetujuan yang ditentukan menjadi `Requires 1 approval for all pull requests from the CodeCommitReview pool`.

```

aws codecommit update-approval-rule-template-description \
  --approval-rule-template-name 1-approver-rule-for-all-pull-requests \
  --approval-rule-template-description "Requires 1 approval for all pull requests
from the CodeCommitReview pool"

```

Output:

```

{
  "approvalRuleTemplate": {
    "creationDate": 1571352720.773,

```

```

    "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests
    from the CodeCommitReview pool",
    "lastModifiedDate": 1571358728.41,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\":
    [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\":
    [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "ruleContentSha256": "2f6c21a5EXAMPLE",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [UpdateApprovalRuleTemplateDescription](#) di Referensi AWS CLI Perintah.

## update-approval-rule-template-name

Contoh kode berikut menunjukkan cara menggunakan `update-approval-rule-template-name`.

### AWS CLI

Untuk memperbarui nama templat aturan persetujuan

`update-approval-rule-template-name` Contoh berikut mengubah nama template aturan persetujuan dari `1-1-approver-rule` `approver-rule-for-all-pull-requests`.

```

aws codecommit update-approval-rule-template-name \
  --old-approval-rule-template-name 1-approver-rule \
  --new-approval-rule-template-name 1-approver-rule-for-all-pull-requests

```

Output:

```

{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "lastModifiedDate": 1571358241.619,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\":
    [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\":
    [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",

```



```
"creationDate": 1571352720.773,  
"lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",  
"approvalRuleTemplateDescription": "All pull requests must be approved by one  
developer on the team.",  
"ruleContentSha256": "2f6c21a5cEXAMPLE"  
}  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Template Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [UpdateApprovalRuleTemplateNamed](#) di Referensi AWS CLI Perintah.

## update-comment

Contoh kode berikut menunjukkan cara menggunakan `update-comment`.

### AWS CLI

Untuk memperbarui komentar pada komit

Contoh ini menunjukkan cara menambahkan konten "Fixed as requested. I'll update the pull request." ke komentar dengan ID dari 442b498bEXAMPLE5756813

```
aws codecommit update-comment \  
--comment-id 442b498bEXAMPLE5756813 \  
--content "Fixed as requested. I'll update the pull request."
```

### Output:

```
{  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "",  
    "commentId": "442b498bEXAMPLE5756813",  
    "content": "Fixed as requested. I'll update the pull request.",  
    "creationDate": 1508369929.783,  
    "deleted": false,  
    "lastModifiedDate": 1508369929.287,  
    "callerReactions": [],  
    "reactionCounts":  
      {  

```

```
        "THUMBSUP" : 2
      }
    }
  }
```

- Untuk detail API, lihat [UpdateComment](#) di Referensi AWS CLI Perintah.

## update-default-branch

Contoh kode berikut menunjukkan cara menggunakan `update-default-branch`.

### AWS CLI

Untuk mengubah cabang default untuk repositori

Contoh ini mengubah cabang default untuk AWS CodeCommit repositori. Perintah ini menghasilkan output hanya jika ada kesalahan.

Perintah:

```
aws codecommit update-default-branch --repository-name MyDemoRepo --default-branch-name MyNewBranch
```

Output:

```
None.
```

- Untuk detail API, lihat [UpdateDefaultBranch](#) di Referensi AWS CLI Perintah.

## update-pull-request-approval-rule-content

Contoh kode berikut menunjukkan cara menggunakan `update-pull-request-approval-rule-content`.

### AWS CLI

Untuk mengedit aturan persetujuan untuk permintaan tarik

`update-pull-request-approval-rule-content` Contoh berikut memperbarui dia menetapkan aturan persetujuan untuk meminta satu persetujuan pengguna dari kumpulan persetujuan yang mencakup setiap pengguna IAM dalam 123456789012 AWS akun.

```
aws codecommit update-pull-request-approval-rule-content \
  --pull-request-id 27 \
  --approval-rule-name "Require two approved approvers" \
  --approval-rule-content "{Version: 2018-11-08, Statements: [{Type:
  \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers:
  [\"CodeCommitApprovers:123456789012:user/*\"]}]}"
```

Output:

```
{
  "approvalRule": {
    "approvalRuleContent": "{Version: 2018-11-08, Statements:
    [{Type: \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers:
    [\"CodeCommitApprovers:123456789012:user/*\"]}]}",
    "approvalRuleId": "aac33506-EXAMPLE",
    "originApprovalRuleTemplate": {},
    "creationDate": 1570752871.932,
    "lastModifiedDate": 1570754058.333,
    "approvalRuleName": "Require two approved approvers",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "ruleContentSha256": "cd93921cEXAMPLE",
  }
}
```

Untuk informasi selengkapnya, lihat [Mengedit atau Menghapus Aturan Persetujuan](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [UpdatePullRequestApprovalRuleContent](#) di Referensi AWS CLI Perintah.

## update-pull-request-approval-state

Contoh kode berikut menunjukkan cara menggunakan `update-pull-request-approval-state`.

### AWS CLI

Untuk menyetujui atau mencabut persetujuan untuk permintaan tarik

`update-pull-request-approval-state` Contoh berikut menyetujui permintaan tarik dengan ID 27 dan ID revisi dari 9f29d167EXAMPLE. Jika Anda ingin mencabut persetujuan sebagai gantinya, maka atur nilai `--approval-state` parameter ke `REVOKE`.

```
aws codecommit update-pull-request-approval-state \
```

```
--pull-request-id 27 \  
--revision-id 9f29d167EXAMPLE \  
--approval-state "APPROVE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Meninjau Permintaan Tarik](#) di Panduan AWS CodeCommit Pengguna.

- Untuk detail API, lihat [UpdatePullRequestApprovalState](#) di Referensi AWS CLI Perintah.

## update-pull-request-description

Contoh kode berikut menunjukkan cara menggunakan `update-pull-request-description`.

AWS CLI

Untuk mengubah deskripsi permintaan tarik

Contoh ini menunjukkan bagaimana mengubah deskripsi permintaan tarik dengan ID dari 47

```
aws codecommit update-pull-request-description \  
--pull-request-id 47 \  
--description "Updated the pull request to remove unused global variable."
```

Output:

```
{  
  "pullRequest": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "",  
    "creationDate": 1508530823.155,  
    "description": "Updated the pull request to remove unused global variable.",  
    "lastActivityDate": 1508372423.204,  
    "pullRequestId": "47",  
    "pullRequestStatus": "OPEN",  
    "pullRequestTargets": [  
      {  
        "destinationCommit": "9f31c968EXAMPLE",  
        "destinationReference": "refs/heads/main",  
        "mergeMetadata": {
```

```

        "isMerged": false,
      },
      "repositoryName": "MyDemoRepo",
      "sourceCommit": "99132ab0EXAMPLE",
      "sourceReference": "refs/heads/variables-branch"
    }
  ],
  "title": "Consolidation of global variables"
}
}

```

- Untuk detail API, lihat [UpdatePullRequestDescription](#) di Referensi AWS CLI Perintah.

## update-pull-request-status

Contoh kode berikut menunjukkan cara menggunakan `update-pull-request-status`.

### AWS CLI

Untuk mengubah status permintaan tarik

Contoh ini menunjukkan bagaimana mengubah status permintaan tarik dengan ID dari 42 ke status CLOSED dalam AWS CodeCommit repositori bernama. MyDemoRepo

```

aws codecommit update-pull-request-status \
  --pull-request-id 42 \
  --pull-request-status CLOSED

```

### Output:

```

{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
          {\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [
            {\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"}]}]}\",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approvers-needed-for-this-change",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,

```

```

        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
    }
],
"authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
"clientRequestToken": "",
"creationDate": 1508530823.165,
"description": "Updated the pull request to remove unused global variable.",
"lastActivityDate": 1508372423.12,
"pullRequestId": "47",
"pullRequestStatus": "CLOSED",
"pullRequestTargets": [
    {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
            "isMerged": false,
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
    }
],
"title": "Consolidation of global variables"
}
}

```

- Untuk detail API, lihat [UpdatePullRequestStatus](#) di Referensi AWS CLI Perintah.

## update-pull-request-title

Contoh kode berikut menunjukkan cara menggunakan `update-pull-request-title`.

### AWS CLI

Untuk mengubah judul permintaan tarik

Contoh ini menunjukkan bagaimana mengubah judul permintaan tarik dengan ID dari. 47

```

aws codecommit update-pull-request-title \
  --pull-request-id 47 \
  --title "Consolidation of global variables - updated review"

```

## Output:

```

{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\\\"DestinationReferences\": [\\\"refs/heads/main\\\"],\\\"Statements\": [{\\\"Type
\\\": \\\"Approvers\\\",\\\"NumberOfApprovalsNeeded\": 2,\\\"ApprovalPoolMembers\":
[\\\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\\\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b26gr-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.12,
    "description": "Review the latest changes and updates to the global
variables. I have updated this request with some changes, including removing some
unused variables.",
    "lastActivityDate": 1508372657.188,
    "pullRequestId": "47",
    "pullRequestStatus": "OPEN",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": false,
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ],
    "title": "Consolidation of global variables - updated review"
  }
}

```

```
}  
}
```

- Untuk detail API, lihat [UpdatePullRequestTitle](#) di Referensi AWS CLI Perintah.

## **update-repository-description**

Contoh kode berikut menunjukkan cara menggunakan `update-repository-description`.

### AWS CLI

Untuk mengubah deskripsi untuk repositori

Contoh ini mengubah deskripsi untuk AWS CodeCommit repositori. Perintah ini menghasilkan output hanya jika ada kesalahan.

Perintah:

```
aws codecommit update-repository-description --repository-name MyDemoRepo --  
repository-description "This description was changed"
```

Output:

```
None .
```

- Untuk detail API, lihat [UpdateRepositoryDescription](#) di Referensi AWS CLI Perintah.

## **update-repository-name**

Contoh kode berikut menunjukkan cara menggunakan `update-repository-name`.

### AWS CLI

Untuk mengubah nama repositori

Contoh ini mengubah nama AWS CodeCommit repositori. Perintah ini menghasilkan output hanya jika ada kesalahan. Mengubah nama AWS CodeCommit repositori akan mengubah URL SSH dan HTTPS yang harus dihubungkan pengguna ke repositori. Pengguna tidak akan dapat terhubung ke repositori ini sampai mereka memperbarui pengaturan koneksi mereka. Selain itu, karena ARN repositori akan berubah, yang mengubah nama repositori akan membatalkan setiap kebijakan pengguna IAM yang bergantung pada ARN repositori ini.



**Perintah:**

```
aws codecommit update-repository-name --old-name MyDemoRepo --new-name
MyRenamedDemoRepo
```

**Output:**

```
None .
```

- Untuk detail API, lihat [UpdateRepositoryName](#) di Referensi AWS CLI Perintah.

## CodeDeploy contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CodeDeploy.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

**Topik**

- [Tindakan](#)

**Tindakan****add-tags-to-on-premises-instances**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-on-premises-instances`.

**AWS CLI**

Untuk menambahkan tag ke instance lokal

`add-tags-to-on-premises-instances` Contoh berikut mengaitkan tag instans lokal yang sama ke dua instance lokal. AWS CodeDeploy itu tidak mendaftarkan instance lokal dengan AWS CodeDeploy

```
aws deploy add-tags-to-on-premises-instances \
  --instance-names AssetTag12010298EX AssetTag23121309EX \
  --tags Key=Name,Value=CodeDeployDemo-OnPrem
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AddTagsToOnPremisesInstances](#) di Referensi AWS CLI Perintah.

## `batch-get-application-revisions`

Contoh kode berikut menunjukkan cara menggunakan `batch-get-application-revisions`.

### AWS CLI

Untuk mengambil informasi tentang revisi aplikasi

`batch-get-application-revisions` Contoh berikut mengambil informasi tentang revisi tertentu yang disimpan dalam repositori. GitHub

```
aws deploy batch-get-application-revisions \
  --application-name my-codedeploy-application \
  --revisions "[{\\"githubLocation\\": {\\"commitId\\": \\"fa85936EXAMPLEe31736c051f10d77297EXAMPLE\\", \\"repository\\": \\"my-github-token/my-repository\\"}, \\"revisionType\\": \\"GitHub\\"}]"
```

Output:

```
{
  "revisions": [
    {
      "genericRevisionInfo": {
        "description": "Application revision registered by Deployment ID: d-A1B2C3111",
        "lastUsedTime": 1556912355.884,
        "registerTime": 1556912355.884,
        "firstUsedTime": 1556912355.884,
        "deploymentGroups": []
      },
    },
  ],
}
```

```

        "revisionLocation": {
            "revisionType": "GitHub",
            "gitHubLocation": {
                "commitId": "fa85936EXAMPLEa31736c051f10d77297EXAMPLE",
                "repository": "my-github-token/my-repository"
            }
        }
    ],
    "applicationName": "my-codedeploy-application",
    "errorMessage": ""
}

```

Untuk informasi selengkapnya, lihat [BatchGetApplicationRevisions](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [BatchGetApplicationRevisions](#) di Referensi AWS CLI Perintah.

## batch-get-applications

Contoh kode berikut menunjukkan cara menggunakan `batch-get-applications`.

### AWS CLI

Untuk mendapatkan informasi tentang beberapa aplikasi

`batch-get-applications` Contoh berikut menampilkan informasi tentang beberapa aplikasi yang terkait dengan AWS akun pengguna.

```
aws deploy batch-get-applications --application-names WordPress_App MyOther_App
```

Output:

```

{
  "applicationsInfo": [
    {
      "applicationName": "WordPress_App",
      "applicationId": "d9dd6993-f171-44fa-a811-211e4EXAMPLE",
      "createTime": 1407878168.078,
      "linkedToGitHub": false
    },
    {
      "applicationName": "MyOther_App",

```

```

    "applicationId": "8ca57519-31da-42b2-9194-8bb16EXAMPLE",
    "createTime": 1407453571.63,
    "linkedToGitHub": false
  }
]
}

```

- Untuk detail API, lihat [BatchGetApplications](#) di Referensi AWS CLI Perintah.

## batch-get-deployment-groups

Contoh kode berikut menunjukkan cara menggunakan `batch-get-deployment-groups`.

### AWS CLI

Untuk mengambil informasi tentang satu atau beberapa grup penyebaran

`batch-get-deployment-groups` Contoh berikut mengambil informasi tentang dua kelompok penyebaran yang terkait dengan aplikasi yang ditentukan CodeDeploy .

```

aws deploy batch-get-deployment-groups \
  --application-name my-codedeploy-application \
  --deployment-group-names ["my-deployment-group-1","my-deployment-group-2"]

```

### Output:

```

{
  "deploymentGroupsInfo": [
    {
      "deploymentStyle": {
        "deploymentOption": "WITHOUT_TRAFFIC_CONTROL",
        "deploymentType": "IN_PLACE"
      },
      "autoRollbackConfiguration": {
        "enabled": false
      },
      "onPremisesTagSet": {
        "onPremisesTagSetList": []
      },
      "serviceRoleArn": "arn:aws:iam::123456789012:role/CodeDeployServiceRole",
      "lastAttemptedDeployment": {

```

```

        "endTime": 1556912366.415,
        "status": "Failed",
        "createTime": 1556912355.884,
        "deploymentId": "d-A1B2C3111"
    },
    "autoScalingGroups": [],
    "deploymentGroupName": "my-deployment-group-1",
    "ec2TagSet": {
        "ec2TagSetList": [
            [
                {
                    "Type": "KEY_AND_VALUE",
                    "Value": "my-EC2-instance",
                    "Key": "Name"
                }
            ]
        ]
    },
    "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111example",
    "triggerConfigurations": [],
    "applicationName": "my-codedeploy-application",
    "computePlatform": "Server",
    "deploymentConfigName": "CodeDeployDefault.AllAtOnce"
},
{
    "deploymentStyle": {
        "deploymentOption": "WITHOUT_TRAFFIC_CONTROL",
        "deploymentType": "IN_PLACE"
    },
    "autoRollbackConfiguration": {
        "enabled": false
    },
    "onPremisesTagSet": {
        "onPremisesTagSetList": []
    },
    "serviceRoleArn": "arn:aws:iam::123456789012:role/
CodeDeployServiceRole",
    "autoScalingGroups": [],
    "deploymentGroupName": "my-deployment-group-2",
    "ec2TagSet": {
        "ec2TagSetList": [
            [
                {
                    "Type": "KEY_AND_VALUE",

```

```

        "Value": "my-EC2-instance",
        "Key": "Name"
      }
    ]
  ],
  "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-22222example",
  "triggerConfigurations": [],
  "applicationName": "my-codedeploy-application",
  "computePlatform": "Server",
  "deploymentConfigName": "CodeDeployDefault.AllAtOnce"
}
],
"errorMessage": ""
}

```

Untuk informasi selengkapnya, lihat [BatchGetDeploymentGroups](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [BatchGetDeploymentGroups](#) di Referensi AWS CLI Perintah.

## batch-get-deployment-targets

Contoh kode berikut menunjukkan cara menggunakan `batch-get-deployment-targets`.

### AWS CLI

Untuk mengambil target yang terkait dengan penerapan

`batch-get-deployment-targets` Contoh berikut mengembalikan informasi tentang salah satu target yang terkait dengan penyebaran yang ditentukan.

```

aws deploy batch-get-deployment-targets \
  --deployment-id "d-1A2B3C4D5" \
  --target-ids "i-01a2b3c4d5e6f1111"

```

### Output:

```

{
  "deploymentTargets": [
    {
      "deploymentTargetType": "InstanceTarget",
      "instanceTarget": {

```

```
    "lifecycleEvents": [
      {
        "startTime": 1556918592.162,
        "lifecycleEventName": "ApplicationStop",
        "status": "Succeeded",
        "endTime": 1556918592.247,
        "diagnostics": {
          "scriptName": "",
          "errorCode": "Success",
          "logTail": "",
          "message": "Succeeded"
        }
      },
      {
        "startTime": 1556918593.193,
        "lifecycleEventName": "DownloadBundle",
        "status": "Succeeded",
        "endTime": 1556918593.981,
        "diagnostics": {
          "scriptName": "",
          "errorCode": "Success",
          "logTail": "",
          "message": "Succeeded"
        }
      },
      {
        "startTime": 1556918594.805,
        "lifecycleEventName": "BeforeInstall",
        "status": "Succeeded",
        "endTime": 1556918681.807,
        "diagnostics": {
          "scriptName": "",
          "errorCode": "Success",
          "logTail": "",
          "message": "Succeeded"
        }
      }
    ],
    "targetArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-01a2b3c4d5e6f1111",
    "deploymentId": "d-1A2B3C4D5",
    "lastUpdatedAt": 1556918687.504,
    "targetId": "i-01a2b3c4d5e6f1111",
    "status": "Succeeded"
```

```

    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [BatchGetDeploymentTargets](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [BatchGetDeploymentTargets](#) di Referensi AWS CLI Perintah.

## batch-get-deployments

Contoh kode berikut menunjukkan cara menggunakan `batch-get-deployments`.

### AWS CLI

Untuk mendapatkan informasi tentang beberapa penerapan

`batch-get-deployments` Contoh berikut menampilkan informasi tentang beberapa penerapan yang terkait dengan akun pengguna. AWS

```
aws deploy batch-get-deployments --deployment-ids d-A1B2C3111 d-A1B2C3222
```

### Output:

```

{
  "deploymentsInfo": [
    {
      "applicationName": "WordPress_App",
      "status": "Failed",
      "deploymentOverview": {
        "Failed": 0,
        "InProgress": 0,
        "Skipped": 0,
        "Succeeded": 1,
        "Pending": 0
      },
      "deploymentConfigName": "CodeDeployDefault.OneAtATime",
      "creator": "user",
      "deploymentGroupName": "WordPress_DG",
      "revision": {
        "revisionType": "S3",
        "s3Location": {

```



```
        "bundleType": "zip",
        "version": "uTecLusEXAMPLEFXtfUcyfV8bEXAMPLE",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPressApp.zip"
    }
},
"deploymentId": "d-A1B2C3111",
"createTime": 1408480721.9,
"completeTime": 1408480741.822
},
{
    "applicationName": "MyOther_App",
    "status": "Failed",
    "deploymentOverview": {
        "Failed": 1,
        "InProgress": 0,
        "Skipped": 0,
        "Succeeded": 0,
        "Pending": 0
    },
    "deploymentConfigName": "CodeDeployDefault.OneAtATime",
    "creator": "user",
    "errorInformation": {
        "message": "Deployment failed: Constraint default violated: No hosts
succeeded.",
        "code": "HEALTH_CONSTRAINTS"
    },
    "deploymentGroupName": "MyOther_DG",
    "revision": {
        "revisionType": "S3",
        "s3Location": {
            "bundleType": "zip",
            "eTag": "\"dd56cfdEXAMPLE8e768f9d77fEXAMPLE\"",
            "bucket": "CodeDeployDemoBucket",
            "key": "MyOtherApp.zip"
        }
    },
    "deploymentId": "d-A1B2C3222",
    "createTime": 1409764576.589,
    "completeTime": 1409764596.101
}
]
}
```

- Untuk detail API, lihat [BatchGetDeployments](#) di Referensi AWS CLI Perintah.

## batch-get-on-premises-instances

Contoh kode berikut menunjukkan cara menggunakan `batch-get-on-premises-instances`.

### AWS CLI

Untuk mendapatkan informasi tentang satu atau beberapa instans lokal

`batch-get-on-premises-instances` Contoh berikut mendapatkan informasi tentang dua instance lokal.

```
aws deploy batch-get-on-premises-instances --instance-names AssetTag12010298EX
AssetTag23121309EX
```

### Output:

```
{
  "instanceInfos": [
    {
      "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag12010298EX",
      "tags": [
        {
          "Value": "CodeDeployDemo-OnPrem",
          "Key": "Name"
        }
      ],
      "instanceName": "AssetTag12010298EX",
      "registerTime": 1425579465.228,
      "instanceArn": "arn:aws:codedeploy:us-west-2:123456789012:instance/
AssetTag12010298EX_4IwLNI2Alh"
    },
    {
      "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag23121309EX",
      "tags": [
        {
          "Value": "CodeDeployDemo-OnPrem",
          "Key": "Name"
        }
      ]
    }
  ]
}
```

```
    ],
    "instanceName": "AssetTag23121309EX",
    "registerTime": 1425595585.988,
    "instanceArn": "arn:aws:codedeploy:us-west-2:80398EXAMPLE:instance/
AssetTag23121309EX_PomUy64Was"
  }
]
}
```

- Untuk detail API, lihat [BatchGetOnPremisesInstances](#) di Referensi AWS CLI Perintah.

## continue-deployment

Contoh kode berikut menunjukkan cara menggunakan `continue-deployment`.

### AWS CLI

Untuk mulai mengalihkan lalu lintas tanpa menunggu waktu tunggu yang ditentukan berlalu.

`continue-deployment` Contoh berikut mulai mengalihkan lalu lintas dari instance di lingkungan asli yang siap untuk mulai mengalihkan lalu lintas ke instance di lingkungan pengganti.

```
aws deploy continue-deployment \
  --deployment-id "d-A1B2C3111" \
  --deployment-wait-type "READY_WAIT"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ContinueDeployment](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [ContinueDeployment](#) di Referensi AWS CLI Perintah.

## create-application

Contoh kode berikut menunjukkan cara menggunakan `create-application`.

### AWS CLI

Untuk membuat aplikasi

`create-application` Contoh berikut membuat aplikasi dan mengaitkannya dengan AWS akun pengguna.

```
aws deploy create-application --application-name MyOther_App
```

Output:

```
{
  "applicationId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Untuk detail API, lihat [CreateApplication](#) di Referensi AWS CLI Perintah.

## create-deployment-config

Contoh kode berikut menunjukkan cara menggunakan `create-deployment-config`.

### AWS CLI

Untuk membuat konfigurasi penerapan kustom

`create-deployment-config` Contoh berikut membuat konfigurasi penerapan kustom dan mengaitkannya dengan akun pengguna. AWS

```
aws deploy create-deployment-config \
  --deployment-config-name ThreeQuartersHealthy \
  --minimum-healthy-hosts type=FLEET_PERCENT,value=75
```

Output:

```
{
  "deploymentConfigId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Untuk detail API, lihat [CreateDeploymentConfig](#) di Referensi AWS CLI Perintah.

## create-deployment-group

Contoh kode berikut menunjukkan cara menggunakan `create-deployment-group`.

### AWS CLI

Untuk membuat grup penyebaran

`create-deployment-group` Contoh berikut membuat grup penyebaran dan mengaitkannya dengan aplikasi yang ditentukan dan akun pengguna. AWS

```
aws deploy create-deployment-group \
  --application-name WordPress_App \
  --auto-scaling-groups CodeDeployDemo-ASG \
  --deployment-config-name CodeDeployDefault.OneAtATime \
  --deployment-group-name WordPress_DG \
  --ec2-tag-filters Key=Name,Value=CodeDeployDemo,Type=KEY_AND_VALUE \
  --service-role-arn arn:aws:iam::123456789012:role/CodeDeployDemoRole
```

Output:

```
{
  "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Untuk detail API, lihat [CreateDeploymentGroup](#) di Referensi AWS CLI Perintah.

## create-deployment

Contoh kode berikut menunjukkan cara menggunakan `create-deployment`.

### AWS CLI

Contoh 1: Untuk membuat CodeDeploy penerapan menggunakan platform komputasi EC2/On-premise

`create-deployment` Contoh berikut membuat penyebaran dan mengaitkannya dengan akun pengguna. AWS

```
aws deploy create-deployment \
  --application-name WordPress_App \
  --deployment-config-name CodeDeployDefault.OneAtATime \
  --deployment-group-name WordPress_DG \
  --description "My demo deployment" \
  --s3-location
bucket=CodeDeployDemoBucket,bundleType=zip,eTag=dd56cfdEXAMPLE8e768f9d77fEXAMPLE,key=WordPr
```

Output:

```
{
  "deploymentId": "d-A1B2C3111"
}
```

Contoh 2: Untuk membuat CodeDeploy penerapan menggunakan platform komputasi Amazon ECS

`create-deployment` Contoh berikut menggunakan dua file berikut untuk menyebarkan layanan Amazon ECS.

Isi `create-deployment.json` file:

```
{
  "applicationName": "ecs-deployment",
  "deploymentGroupName": "ecs-deployment-dg",
  "revision": {
    "revisionType": "S3",
    "s3Location": {
      "bucket": "ecs-deployment-bucket",
      "key": "appspec.yaml",
      "bundleType": "YAML"
    }
  }
}
```

File itu, pada gilirannya, mengambil file berikut `appspec.yaml` dari bucket S3 yang disebut `ecs-deployment-bucket`

```
version: 0.0
Resources:
  - TargetService:
    Type: AWS::ECS::Service
    Properties:
      TaskDefinition: "arn:aws:ecs:region:123456789012:task-definition/ecs-task-def:2"
      LoadBalancerInfo:
        ContainerName: "sample-app"
        ContainerPort: 80
        PlatformVersion: "LATEST"
```

Perintah:

```
aws deploy create-deployment \  
  --cli-input-json file://create-deployment.json \  
  --region us-east-1
```

Output:

```
{  
  "deploymentId": "d-1234ABCDE"  
}
```

Untuk informasi selengkapnya, lihat [CreateDeployment](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [CreateDeployment](#) di Referensi AWS CLI Perintah.

## **delete-application**

Contoh kode berikut menunjukkan cara menggunakan `delete-application`.

### AWS CLI

Untuk menghapus aplikasi

`delete-application` Contoh berikut menghapus aplikasi yang ditentukan yang terkait dengan AWS akun pengguna.

```
aws deploy delete-application --application-name WordPress_App
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteApplication](#) di Referensi AWS CLI Perintah.

## **delete-deployment-config**

Contoh kode berikut menunjukkan cara menggunakan `delete-deployment-config`.

### AWS CLI

Untuk menghapus konfigurasi penerapan

`delete-deployment-config` Contoh berikut menghapus konfigurasi penerapan kustom yang dikaitkan dengan akun pengguna. AWS

```
aws deploy delete-deployment-config --deployment-config-name ThreeQuartersHealthy
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteDeploymentConfig](#) di Referensi AWS CLI Perintah.

## **delete-deployment-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-deployment-group`.

### AWS CLI

Untuk menghapus grup penyebaran

`delete-deployment-group` Contoh berikut menghapus grup penyebaran yang terkait dengan aplikasi yang ditentukan.

```
aws deploy delete-deployment-group \  
  --application-name WordPress_App \  
  --deployment-group-name WordPress_DG
```

Output:

```
{  
  "hooksNotCleanedUp": []  
}
```

- Untuk detail API, lihat [DeleteDeploymentGroup](#) di Referensi AWS CLI Perintah.

## **delete-git-hub-account-token**

Contoh kode berikut menunjukkan cara menggunakan `delete-git-hub-account-token`.

### AWS CLI

Untuk menghapus koneksi GitHub akun

`delete-git-hub-account-token` Contoh berikut menghapus koneksi GitHub akun yang ditentukan.

```
aws deploy delete-git-hub-account-token --token-name my-github-account
```



**Output:**

```
{
  "tokenName": "my-github-account"
}
```

Untuk informasi selengkapnya, lihat [DeleteGitHubAccountToken](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [DeleteGitHubAccountToken](#) di Referensi AWS CLI Perintah.

**deregister-on-premises-instance**

Contoh kode berikut menunjukkan cara menggunakan `deregister-on-premises-instance`.

**AWS CLI**

Untuk membatalkan pendaftaran instans lokal

`deregister-on-premises-instance` Contoh berikut membatalkan pendaftaran instans lokal dengan AWS CodeDeploy, tetapi tidak menghapus pengguna IAM yang terkait dengan instance, juga tidak memisahkan tag instans lokal AWS CodeDeploy dari instance. Itu juga tidak menghapus Instalasi AWS CodeDeploy Agen dari instance atau menghapus file konfigurasi lokal dari instance.

```
aws deploy deregister-on-premises-instance --instance-name AssetTag12010298EX
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeregisterOnPremisesInstance](#) di Referensi AWS CLI Perintah.

**deregister**

Contoh kode berikut menunjukkan cara menggunakan `deregister`.

**AWS CLI**

Untuk membatalkan pendaftaran instans lokal

`deregister` Contoh berikut membatalkan pendaftaran instance lokal dengan AWS CodeDeploy. Itu tidak menghapus pengguna IAM yang terkait dengan instance. Ini terputus dalam AWS

CodeDeploy tag lokal dari instance. Itu tidak menghapus instalasi AWS CodeDeploy Agen dari instance atau menghapus file konfigurasi lokal dari instance.

```
aws deploy deregister \  
  --instance-name AssetTag12010298EX \  
  --no-delete-iam-user \  
  --region us-west-2
```

Output:

```
Retrieving on-premises instance information... DONE  
IamUserArn: arn:aws:iam::80398EXAMPLE:user/AWS/CodeDeploy/AssetTag12010298EX  
Tags: Key=Name,Value=CodeDeployDemo-OnPrem  
Removing tags from the on-premises instance... DONE  
Deregistering the on-premises instance... DONE  
Run the following command on the on-premises instance to uninstall the codedeploy-  
agent:  
aws deploy uninstall
```

- Untuk detail API, lihat [Deregister](#) di AWS CLI Referensi Perintah.

## get-application-revision

Contoh kode berikut menunjukkan cara menggunakan `get-application-revision`.

### AWS CLI

Untuk mendapatkan informasi tentang revisi aplikasi

`get-application-revision` Contoh berikut menampilkan informasi tentang revisi aplikasi yang terkait dengan aplikasi yang ditentukan.

```
aws deploy get-application-revision \  
  --application-name WordPress_App \  
  --s3-location  
bucket=CodeDeployDemoBucket,bundleType=zip,eTag=dd56cfdEXAMPLE8e768f9d77fEXAMPLE,key=WordPress
```

Output:

```
{  
  "applicationName": "WordPress_App",
```

```

    "revisionInfo": {
      "description": "Application revision registered by Deployment ID: d-
A1B2C3111",
      "registerTime": 1411076520.009,
      "deploymentGroups": "WordPress_DG",
      "lastUsedTime": 1411076520.009,
      "firstUsedTime": 1411076520.009
    },
    "revision": {
      "revisionType": "S3",
      "s3Location": {
        "bundleType": "zip",
        "eTag": "dd56cfdEXAMPLE8e768f9d77fEXAMPLE",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPressApp.zip"
      }
    }
  }
}

```

- Untuk detail API, lihat [GetApplicationRevision](#) di Referensi AWS CLI Perintah.

## get-application

Contoh kode berikut menunjukkan cara menggunakan `get-application`.

### AWS CLI

Untuk mendapatkan informasi tentang aplikasi

`get-application` Contoh berikut menampilkan informasi tentang aplikasi yang terkait dengan AWS akun pengguna.

```
aws deploy get-application --application-name WordPress_App
```

Output:

```

{
  "application": {
    "applicationName": "WordPress_App",
    "applicationId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "createTime": 1407878168.078,
    "linkedToGitHub": false
  }
}

```

```
}  
}
```

- Untuk detail API, lihat [GetApplication](#) di Referensi AWS CLI Perintah.

## get-deployment-config

Contoh kode berikut menunjukkan cara menggunakan `get-deployment-config`.

### AWS CLI

Untuk mendapatkan informasi tentang konfigurasi penerapan

`get-deployment-config` Contoh berikut menampilkan informasi tentang konfigurasi penyebaran yang terkait dengan AWS akun pengguna.

```
aws deploy get-deployment-config --deployment-config-name ThreeQuartersHealthy
```

Output:

```
{  
  "deploymentConfigInfo": {  
    "deploymentConfigId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "minimumHealthyHosts": {  
      "type": "FLEET_PERCENT",  
      "value": 75  
    },  
    "createTime": 1411081164.379,  
    "deploymentConfigName": "ThreeQuartersHealthy"  
  }  
}
```

- Untuk detail API, lihat [GetDeploymentConfig](#) di Referensi AWS CLI Perintah.

## get-deployment-group

Contoh kode berikut menunjukkan cara menggunakan `get-deployment-group`.

### AWS CLI

Untuk melihat informasi tentang grup penyebaran

`get-deployment-group` Contoh berikut menampilkan informasi tentang grup penyebaran yang terkait dengan aplikasi yang ditentukan.

```
aws deploy get-deployment-group \  
  --application-name WordPress_App \  
  --deployment-group-name WordPress_DG
```

Output:

```
{  
  "deploymentGroupInfo": {  
    "applicationName": "WordPress_App",  
    "autoScalingGroups": [  
      "CodeDeployDemo-ASG"  
    ],  
    "deploymentConfigName": "CodeDeployDefault.OneAtATime",  
    "ec2TagFilters": [  
      {  
        "Type": "KEY_AND_VALUE",  
        "Value": "CodeDeployDemo",  
        "Key": "Name"  
      }  
    ],  
    "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "serviceRoleArn": "arn:aws:iam::123456789012:role/CodeDeployDemoRole",  
    "deploymentGroupName": "WordPress_DG"  
  }  
}
```

- Untuk detail API, lihat [GetDeploymentGroup](#) di Referensi AWS CLI Perintah.

## **get-deployment-instance**

Contoh kode berikut menunjukkan cara menggunakan `get-deployment-instance`.

### AWS CLI

Untuk mendapatkan informasi tentang instance penerapan

`get-deployment-instance` Contoh berikut menampilkan informasi tentang instance penyebaran yang terkait dengan penerapan yang ditentukan.

```
aws deploy get-deployment-instance --deployment-id d-QA4G4F9EX --instance-id
i-902e9fEX
```

### Output:

```
{
  "instanceSummary": {
    "instanceId": "arn:aws:ec2:us-east-1:80398EXAMPLE:instance/i-902e9fEX",
    "lifecycleEvents": [
      {
        "status": "Succeeded",
        "endTime": 1408480726.569,
        "startTime": 1408480726.437,
        "lifecycleEventName": "ApplicationStop"
      },
      {
        "status": "Succeeded",
        "endTime": 1408480728.016,
        "startTime": 1408480727.665,
        "lifecycleEventName": "DownloadBundle"
      },
      {
        "status": "Succeeded",
        "endTime": 1408480729.744,
        "startTime": 1408480729.125,
        "lifecycleEventName": "BeforeInstall"
      },
      {
        "status": "Succeeded",
        "endTime": 1408480730.979,
        "startTime": 1408480730.844,
        "lifecycleEventName": "Install"
      },
      {
        "status": "Failed",
        "endTime": 1408480732.603,
        "startTime": 1408480732.1,
        "lifecycleEventName": "AfterInstall"
      },
      {
        "status": "Skipped",
        "endTime": 1408480732.606,
        "lifecycleEventName": "ApplicationStart"
      }
    ]
  }
}
```

```

    },
    {
      "status": "Skipped",
      "endTime": 1408480732.606,
      "lifecycleEventName": "ValidateService"
    }
  ],
  "deploymentId": "d-QA4G4F9EX",
  "lastUpdatedAt": 1408480733.152,
  "status": "Failed"
}
}

```

- Untuk detail API, lihat [GetDeploymentInstance](#) di Referensi AWS CLI Perintah.

## get-deployment-target

Contoh kode berikut menunjukkan cara menggunakan `get-deployment-target`.

### AWS CLI

Untuk mengembalikan informasi tentang target penyebaran

`get-deployment-target` Contoh berikut mengembalikan informasi tentang target penyebaran yang terkait dengan penerapan yang ditentukan.

```

aws deploy get-deployment-target \
  --deployment-id "d-A1B2C3111" \
  --target-id "i-a1b2c3d4e5f611111"

```

Output:

```

{
  "deploymentTarget": {
    "deploymentTargetType": "InstanceTarget",
    "instanceTarget": {
      "lastUpdatedAt": 1556918687.504,
      "targetId": "i-a1b2c3d4e5f611111",
      "targetArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-a1b2c3d4e5f611111",
      "status": "Succeeded",
      "lifecycleEvents": [

```

```
{
  "status": "Succeeded",
  "diagnostics": {
    "errorCode": "Success",
    "message": "Succeeded",
    "logTail": "",
    "scriptName": ""
  },
  "lifecycleEventName": "ApplicationStop",
  "startTime": 1556918592.162,
  "endTime": 1556918592.247
},
{
  "status": "Succeeded",
  "diagnostics": {
    "errorCode": "Success",
    "message": "Succeeded",
    "logTail": "",
    "scriptName": ""
  },
  "lifecycleEventName": "DownloadBundle",
  "startTime": 1556918593.193,
  "endTime": 1556918593.981
},
{
  "status": "Succeeded",
  "diagnostics": {
    "errorCode": "Success",
    "message": "Succeeded",
    "logTail": "",
    "scriptName": ""
  },
  "lifecycleEventName": "BeforeInstall",
  "startTime": 1556918594.805,
  "endTime": 1556918681.807
},
{
  "status": "Succeeded",
  "diagnostics": {
    "errorCode": "Success",
    "message": "Succeeded",
    "logTail": "",
    "scriptName": ""
  },
}
```



```
        "lifecycleEventName": "Install",
        "startTime": 1556918682.696,
        "endTime": 1556918683.005
    },
    {
        "status": "Succeeded",
        "diagnostics": {
            "errorCode": "Success",
            "message": "Succeeded",
            "logTail": "",
            "scriptName": ""
        },
        "lifecycleEventName": "AfterInstall",
        "startTime": 1556918684.135,
        "endTime": 1556918684.216
    },
    {
        "status": "Succeeded",
        "diagnostics": {
            "errorCode": "Success",
            "message": "Succeeded",
            "logTail": "",
            "scriptName": ""
        },
        "lifecycleEventName": "ApplicationStart",
        "startTime": 1556918685.211,
        "endTime": 1556918685.295
    },
    {
        "status": "Succeeded",
        "diagnostics": {
            "errorCode": "Success",
            "message": "Succeeded",
            "logTail": "",
            "scriptName": ""
        },
        "lifecycleEventName": "ValidateService",
        "startTime": 1556918686.65,
        "endTime": 1556918686.747
    }
],
"deploymentId": "d-A1B2C3111"
}
```

```
}
```

Untuk informasi selengkapnya, lihat [GetDeploymentTarget](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [GetDeploymentTarget](#) di Referensi AWS CLI Perintah.

## get-deployment

Contoh kode berikut menunjukkan cara menggunakan `get-deployment`.

### AWS CLI

Untuk mendapatkan informasi tentang penerapan

`get-deployment` Contoh berikut menampilkan informasi tentang penyebaran yang terkait dengan AWS akun pengguna.

```
aws deploy get-deployment --deployment-id d-A1B2C3123
```

Output:

```
{
  "deploymentInfo": {
    "applicationName": "WordPress_App",
    "status": "Succeeded",
    "deploymentOverview": {
      "Failed": 0,
      "InProgress": 0,
      "Skipped": 0,
      "Succeeded": 1,
      "Pending": 0
    },
    "deploymentConfigName": "CodeDeployDefault.OneAtATime",
    "creator": "user",
    "description": "My WordPress app deployment",
    "revision": {
      "revisionType": "S3",
      "s3Location": {
        "bundleType": "zip",
        "eTag": "\"dd56cfdEXAMPLE8e768f9d77fEXAMPLE\"",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPressApp.zip"
      }
    }
  }
}
```

```

    }
  },
  "deploymentId": "d-A1B2C3123",
  "deploymentGroupName": "WordPress_DG",
  "createTime": 1409764576.589,
  "completeTime": 1409764596.101,
  "ignoreApplicationStopFailures": false
}
}

```

- Untuk detail API, lihat [GetDeployment](#) di Referensi AWS CLI Perintah.

## get-on-premises-instance

Contoh kode berikut menunjukkan cara menggunakan `get-on-premises-instance`.

### AWS CLI

Untuk mendapatkan informasi tentang instans lokal

`get-on-premises-instance` Contoh berikut mengambil informasi tentang instans lokal yang ditentukan.

```
aws deploy get-on-premises-instance --instance-name AssetTag12010298EX
```

Output:

```

{
  "instanceInfo": {
    "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/AssetTag12010298EX",
    "tags": [
      {
        "Value": "CodeDeployDemo-OnPrem",
        "Key": "Name"
      }
    ],
    "instanceName": "AssetTag12010298EX",
    "registerTime": 1425579465.228,
    "instanceArn": "arn:aws:codedeploy:us-east-1:123456789012:instance/AssetTag12010298EX_4IwLNI2Alh"
  }
}

```

```
}
```

- Untuk detail API, lihat [GetOnPremisesInstance](#) di Referensi AWS CLI Perintah.

## install

Contoh kode berikut menunjukkan cara menggunakan `install`.

### AWS CLI

Untuk menginstal instans lokal

`install` Contoh berikut menyalin file konfigurasi lokal dari lokasi yang ditentukan pada instance ke lokasi pada instance yang diharapkan AWS CodeDeploy Agen untuk menemukannya. Itu juga menginstal AWS CodeDeploy Agen pada instance. Itu tidak membuat pengguna IAM apa pun, atau mendaftarkan instans lokal dengan AWS CodeDeploy, atau mengaitkan tag instans lokal apa pun AWS CodeDeploy untuk instance tersebut.

```
aws deploy install \  
  --override-config \  
  --config-file C:\temp\codedeploy.onpremises.yml \  
  --region us-west-2 \  
  --agent-installer s3://aws-codedeploy-us-west-2/latest/codedeploy-agent.msi
```

Output:

```
Creating the on-premises instance configuration file... DONE  
Installing the AWS CodeDeploy Agent... DONE
```

- Untuk detail API, lihat [Menginstal](#) di Referensi AWS CLI Perintah.

## list-application-revisions

Contoh kode berikut menunjukkan cara menggunakan `list-application-revisions`.

### AWS CLI

Untuk mendapatkan informasi tentang revisi aplikasi

`list-application-revisions` Contoh berikut menampilkan informasi tentang semua revisi aplikasi yang terkait dengan aplikasi yang ditentukan.

```
aws deploy list-application-revisions \  
  --application-name WordPress_App \  
  --s3-bucket CodeDeployDemoBucket \  
  --deployed exclude \  
  --s3-key-prefix WordPress_ \  
  --sort-by lastUsedTime \  
  --sort-order descending
```

Output:

```
{  
  "revisions": [  
    {  
      "revisionType": "S3",  
      "s3Location": {  
        "version": "uTecLusvCB_JqHFXtfUcyfV8bEXAMPLE",  
        "bucket": "CodeDeployDemoBucket",  
        "key": "WordPress_App.zip",  
        "bundleType": "zip"  
      }  
    },  
    {  
      "revisionType": "S3",  
      "s3Location": {  
        "version": "tMk.UxgDpMEVb7V187ZM6wVAWEXAMPLE",  
        "bucket": "CodeDeployDemoBucket",  
        "key": "WordPress_App_2-0.zip",  
        "bundleType": "zip"  
      }  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListApplicationRevisions](#) di Referensi AWS CLI Perintah.

## list-applications

Contoh kode berikut menunjukkan cara menggunakan `list-applications`.

### AWS CLI

Untuk mendapatkan informasi tentang aplikasi

`list-applications` Contoh berikut menampilkan informasi tentang semua aplikasi yang terkait dengan AWS akun pengguna.

```
aws deploy list-applications
```

Output:

```
{
  "applications": [
    "WordPress_App",
    "MyOther_App"
  ]
}
```

- Untuk detail API, lihat [ListApplications](#) di Referensi AWS CLI Perintah.

## **list-deployment-configs**

Contoh kode berikut menunjukkan cara menggunakan `list-deployment-configs`.

AWS CLI

Untuk mendapatkan informasi tentang konfigurasi penerapan

`list-deployment-configs` Contoh berikut menampilkan informasi tentang semua konfigurasi penyebaran yang terkait dengan akun pengguna. AWS

```
aws deploy list-deployment-configs
```

Output:

```
{
  "deploymentConfigsList": [
    "ThreeQuartersHealthy",
    "CodeDeployDefault.AllAtOnce",
    "CodeDeployDefault.HalfAtATime",
    "CodeDeployDefault.OneAtATime"
  ]
}
```

- Untuk detail API, lihat [ListDeploymentConfigs](#) di Referensi AWS CLI Perintah.

## list-deployment-groups

Contoh kode berikut menunjukkan cara menggunakan `list-deployment-groups`.

### AWS CLI

Untuk mendapatkan informasi tentang grup penyebaran

`list-deployment-groups` Contoh berikut menampilkan informasi tentang semua kelompok penyebaran yang terkait dengan aplikasi yang ditentukan.

```
aws deploy list-deployment-groups --application-name WordPress_App
```

Output:

```
{
  "applicationName": "WordPress_App",
  "deploymentGroups": [
    "WordPress_DG",
    "WordPress_Beta_DG"
  ]
}
```

- Untuk detail API, lihat [ListDeploymentGroups](#) di Referensi AWS CLI Perintah.

## list-deployment-instances

Contoh kode berikut menunjukkan cara menggunakan `list-deployment-instances`.

### AWS CLI

Untuk mendapatkan informasi tentang instance penerapan

`list-deployment-instances` Contoh berikut menampilkan informasi tentang semua instance penerapan yang terkait dengan penerapan yang ditentukan.

```
aws deploy list-deployment-instances \
```

```
--deployment-id d-A1B2C3111 \  
--instance-status-filter Succeeded
```

Output:

```
{  
  "instancesList": [  
    "i-EXAMPLE11",  
    "i-EXAMPLE22"  
  ]  
}
```

- Untuk detail API, lihat [ListDeploymentInstances](#) di Referensi AWS CLI Perintah.

## list-deployment-targets

Contoh kode berikut menunjukkan cara menggunakan `list-deployment-targets`.

AWS CLI

Untuk mengambil daftar ID target yang terkait dengan penerapan

`list-deployment-targets` Contoh berikut mengambil daftar ID target yang terkait dengan penerapan yang memiliki status “Gagal” atau “.” InProgress

```
aws deploy list-deployment-targets \  
  --deployment-id "d-A1B2C3111" \  
  --target-filters "{\"TargetStatus\": [\"Failed\", \"InProgress\"]}"
```

Output:

```
{  
  "targetIds": [  
    "i-0f1558aaf90e5f1f9"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListDeploymentTargets](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [ListDeploymentTargets](#) di Referensi AWS CLI Perintah.



## list-deployments

Contoh kode berikut menunjukkan cara menggunakan `list-deployments`.

### AWS CLI

Untuk mendapatkan informasi tentang penerapan

`list-deployments` Contoh berikut menampilkan informasi tentang semua penerapan yang terkait dengan kelompok aplikasi dan penyebaran yang ditentukan.

```
aws deploy list-deployments \
  --application-name WordPress_App \
  --create-time-range start=2014-08-19T00:00:00,end=2014-08-20T00:00:00 \
  --deployment-group-name WordPress_DG \
  --include-only-statuses Failed
```

Output:

```
{
  "deployments": [
    "d-EXAMPLE11",
    "d-EXAMPLE22",
    "d-EXAMPLE33"
  ]
}
```

- Untuk detail API, lihat [ListDeployments](#) di Referensi AWS CLI Perintah.

## list-git-hub-account-token-names

Contoh kode berikut menunjukkan cara menggunakan `list-git-hub-account-token-names`.

### AWS CLI

Untuk mencantumkan nama koneksi tersimpan ke GitHub akun

`list-git-hub-account-token-names` Contoh berikut mencantumkan nama-nama koneksi yang disimpan ke GitHub akun untuk AWS pengguna saat ini.

```
aws deploy list-git-hub-account-token-names
```

**Output:**

```
{
  "tokenNameList": [
    "my-first-token",
    "my-second-token",
    "my-third-token"
  ]
}
```

Untuk informasi selengkapnya, lihat [ListGitHubAccountTokenNames](#) di Referensi AWS CodeDeploy API.

- Untuk detail API, lihat [ListGitHubAccountTokenNames](#) di Referensi AWS CLI Perintah.

**list-on-premises-instances**

Contoh kode berikut menunjukkan cara menggunakan `list-on-premises-instances`.

**AWS CLI**

Untuk mendapatkan informasi tentang satu atau beberapa instans lokal

`list-on-premises-instances` Contoh berikut mengambil daftar nama instans lokal yang tersedia untuk instans yang terdaftar AWS CodeDeploy dan juga memiliki tag instans lokal tertentu yang terkait dengan instans. AWS CodeDeploy

```
aws deploy list-on-premises-instances \
  --registration-status Registered \
  --tag-filters Key=Name,Value=CodeDeployDemo-OnPrem,Type=KEY_AND_VALUE
```

**Output:**

```
{
  "instanceNames": [
    "AssetTag12010298EX"
  ]
}
```

- Untuk detail API, lihat [ListOnPremisesInstances](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk sumber daya (aplikasi)

`list-tags-for-resource` Contoh berikut mencantumkan tag yang diterapkan ke aplikasi bernama TestApp di CodeDeploy.

```
aws deploy list-tags-for-resource \
  --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp
```

Output:

```
{
  "Tags": [
    {
      "Key": "Type",
      "Value": "testType"
    },
    {
      "Key": "Name",
      "Value": "testName"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menandai instance untuk grup penerapan CodeDeploy di Panduan Pengguna](#).AWS CodeDeploy

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## push

Contoh kode berikut menunjukkan cara menggunakan `push`.

### AWS CLI

Untuk menggabungkan dan menerapkan revisi aplikasi yang AWS CodeDeploy kompatibel ke Amazon S3

pushContoh berikut bundel dan menyebarkan revisi aplikasi ke Amazon S3 dan kemudian mengaitkan revisi aplikasi dengan aplikasi yang ditentukan.

```
aws deploy push \
  --application-name WordPress_App \
  --description "This is my deployment" \
  --ignore-hidden-files \
  --s3-location s3://CodeDeployDemoBucket/WordPressApp.zip \
  --source /tmp/MyLocalDeploymentFolder/
```

Output menjelaskan cara menggunakan create-deployment perintah untuk membuat penyebaran yang menggunakan revisi aplikasi yang diunggah.

```
To deploy with this revision, run:
aws deploy create-deployment --application-name WordPress_App
  --deployment-config-name <deployment-config-name> --
deployment-group-name <deployment-group-name> --s3-location
  bucket=CodeDeployDemoBucket,key=WordPressApp.zip,bundleType=zip,eTag="cecc9b8EXAMPLE50a6e71"
```

- Untuk detail API, lihat [Push](#) in AWS CLI Command Reference.

## register-application-revision

Contoh kode berikut menunjukkan cara menggunakanregister-application-revision.

### AWS CLI

Untuk mendaftarkan informasi tentang revisi aplikasi yang sudah diunggah

register-application-revisionContoh berikut mendaftarkan informasi tentang revisi aplikasi yang sudah diunggah yang disimpan di Amazon S3 dengan. AWS CodeDeploy

```
aws deploy register-application-revision \
  --application-name WordPress_App \
  --description "Revised WordPress application" \
  --s3-location
  bucket=CodeDeployDemoBucket,key=RevisedWordPressApp.zip,bundleType=zip,eTag=cecc9b8a08eac65
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RegisterApplicationRevision](#)di Referensi AWS CLI Perintah.

## register-on-premises-instance

Contoh kode berikut menunjukkan cara menggunakan `register-on-premises-instance`.

### AWS CLI

Untuk mendaftarkan instans lokal

`register-on-premises-instance` Contoh berikut mendaftarkan instance lokal dengan AWS CodeDeploy. Itu tidak membuat pengguna IAM yang ditentukan, juga tidak mengaitkan tag instance lokal dengan instance terdaftar. AWS CodeDeploy

```
aws deploy register-on-premises-instance \  
  --instance-name AssetTag12010298EX \  
  --iam-user-arn arn:aws:iam::80398EXAMPLE:user/CodeDeployDemoUser-OnPrem
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RegisterOnPremisesInstance](#) di Referensi AWS CLI Perintah.

## register

Contoh kode berikut menunjukkan cara menggunakan `register`.

### AWS CLI

Untuk mendaftarkan instans lokal

`register` Contoh berikut mendaftarkan instance lokal dengan AWS CodeDeploy, mengaitkan tag instans lokal AWS CodeDeploy yang ditentukan dengan instans terdaftar, dan membuat file konfigurasi lokal yang dapat disalin ke instans. Itu tidak membuat pengguna IAM, juga tidak menginstal AWS CodeDeploy Agen pada instance.

```
aws deploy register \  
  --instance-name AssetTag12010298EX \  
  --iam-user-arn arn:aws:iam::80398EXAMPLE:user/CodeDeployUser-OnPrem \  
  --tags Key=Name,Value=CodeDeployDemo-OnPrem \  
  --region us-west-2
```

Output:

```
Registering the on-premises instance... DONE
```

```
Adding tags to the on-premises instance... DONE
Copy the on-premises configuration file named codedeploy.onpremises.yml to the on-
premises instance, and run the following command on the on-premises instance to
install and configure the AWS CodeDeploy Agent:
aws deploy install --config-file codedeploy.onpremises.yml
```

- Untuk detail API, lihat [Mendaftar](#) di Referensi AWS CLI Perintah.

## remove-tags-from-on-premises-instances

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-on-premises-instances`.

### AWS CLI

Untuk menghapus tag dari satu atau beberapa instance lokal

`remove-tags-from-on-premises-instances` Contoh berikut memisahkan tag lokal yang ditentukan AWS CodeDeploy dari instance lokal. Itu tidak membatalkan pendaftaran instans lokal di AWS CodeDeploy, atau menghapus Instans dari instans, AWS CodeDeploy atau menghapus file konfigurasi lokal dari instance, atau menghapus pengguna IAM yang terkait dengan instans.

```
aws deploy remove-tags-from-on-premises-instances \
  --instance-names AssetTag12010298EX AssetTag23121309EX \
  --tags Key=Name,Value=CodeDeployDemo-OnPrem
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RemoveTagsFromOnPremisesInstances](#) di Referensi AWS CLI Perintah.

## stop-deployment

Contoh kode berikut menunjukkan cara menggunakan `stop-deployment`.

### AWS CLI

Untuk mencoba menghentikan penerapan

`stop-deployment` Contoh berikut mencoba menghentikan penerapan yang sedang berlangsung yang terkait dengan akun pengguna. AWS

```
aws menyebarkan stop-deployment --deployment-id d-A1b2c3111
```

## Output:

```
{
  "status": "Succeeded",
  "statusMessage": "No more commands will be scheduled for execution in the
deployment instances"
}
```

- Untuk detail API, lihat [StopDeployment](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya (aplikasi)

`tag-resource` Contoh berikut menambahkan dua tag dengan kunci Nama dan Jenis, dan nilai `testName` dan `testType` ke aplikasi bernama `TestApp` di `CodeDeploy` :

```
aws deploy tag-resource \
  --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp \
  --tags Key=Name,Value=testName Key=Type,Value=testType
```

Jika berhasil, perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai instance untuk grup penerapan CodeDeploy di Panduan Pengguna AWS CodeDeploy](#)

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## uninstall

Contoh kode berikut menunjukkan cara menggunakan `uninstall`.

### AWS CLI

Untuk menghapus instans lokal

`uninstall` Contoh berikut menghapus instalasi AWS CodeDeploy Agen dari instans lokal dan menghapus file konfigurasi lokal dari instance. Itu tidak membatalkan pendaftaran instance AWS

CodeDeploy, atau memisahkan tag instans lokal apa pun AWS CodeDeploy dari instance, atau menghapus pengguna IAM yang terkait dengan instance.

```
aws deploy uninstall
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [Menghapus instalasi](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya (aplikasi)

`untag-resource` Contoh berikut menghapus dua tag dengan kunci Nama dan Jenis dari aplikasi bernama TestApp di CodeDeploy.

```
aws deploy untag-resource \
  --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp \
  --tag-keys Name Type
```

Jika berhasil, perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai instance untuk grup penerapan CodeDeploy di Panduan Pengguna AWS CodeDeploy](#).

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-application

Contoh kode berikut menunjukkan cara menggunakan `update-application`.

### AWS CLI

Untuk mengubah detail aplikasi

`update-application` Contoh berikut mengubah nama aplikasi yang dikaitkan dengan AWS akun pengguna.



```
aws deploy update-application \  
  --application-name WordPress_App \  
  --new-application-name My_WordPress_App
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateApplication](#) di Referensi AWS CLI Perintah.

## update-deployment-group

Contoh kode berikut menunjukkan cara menggunakan `update-deployment-group`.

### AWS CLI

Untuk mengubah informasi tentang grup penyebaran

`update-deployment-group` Contoh berikut mengubah pengaturan grup penyebaran yang terkait dengan aplikasi yang ditentukan.

```
aws deploy update-deployment-group \  
  --application-name WordPress_App \  
  --auto-scaling-groups My_CodeDeployDemo_ASG \  
  --current-deployment-group-name WordPress_DG \  
  --deployment-config-name CodeDeployDefault.AllAtOnce \  
  --ec2-tag-filters Key=Name,Type=KEY_AND_VALUE,Value=My_CodeDeployDemo \  
  --new-deployment-group-name My_WordPress_DepGroup \  
  --service-role-arn arn:aws:iam::80398EXAMPLE:role/CodeDeployDemo-2
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateDeploymentGroup](#) di Referensi AWS CLI Perintah.

## CodeGuru Contoh reviewer menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With CodeGuru Reviewer.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **associate-repository**

Contoh kode berikut menunjukkan cara menggunakan `associate-repository`.

AWS CLI

Contoh 1: Untuk membuat asosiasi repositori Bitbucket

`associate-repository` Contoh berikut membuat asosiasi repositori menggunakan repositori Bitbucket yang ada.

```
aws codeguru-reviewer associate-repository \
  --repository 'Bitbucket={Owner=sample-owner, Name=mySampleRepo,
  ConnectionArn=arn:aws:codestar-connections:us-west-2:123456789012:connection/
  a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 }'
```

Output:

```
{
  "RepositoryAssociation": {
    "ProviderType": "Bitbucket",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1596216896.979,
    "AssociationId": "association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "State": "Associating",
    "StateReason": "Pending Repository Association",
```

```

    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Owner": "sample-owner"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat asosiasi repositori Bitbucket di Amazon CodeGuru Reviewer di Panduan Pengguna](#) Amazon CodeGuru Reviewer.

Contoh 2: Untuk membuat asosiasi repositori GitHub Enterprise

`associate-repository` Contoh berikut membuat asosiasi repositori menggunakan repositori GitHub Enterprise yang ada.

```

aws codeguru-reviewer associate-repository \
  --repository 'GitHubEnterpriseServer={Owner=sample-owner, Name=mySampleRepo,
  ConnectionArn=arn:aws:codestar-connections:us-west-2:123456789012:connection/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 }'

```

Output:

```

{
  "RepositoryAssociation": {
    "ProviderType": "GitHubEnterpriseServer",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1596216896.979,
    "AssociationId": "association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "State": "Associating",
    "StateReason": "Pending Repository Association",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Owner": "sample-owner"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat asosiasi repositori Server GitHub Perusahaan di Amazon CodeGuru Reviewer](#) di Panduan Pengguna Amazon Codeguru Reviewer.

Contoh 3: Untuk membuat asosiasi AWS CodeCommit repositori

`associate-repository` Contoh berikut membuat asosiasi repositori menggunakan repositori yang ada AWS CodeCommit .

```
aws codeguru-reviewer associate-repository \  
  --repository CodeCommit={Name=mySampleRepo}
```

Output:

```
{  
  "RepositoryAssociation": {  
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "Name": "My-ecs-beta-repo",  
    "LastUpdatedTimeStamp": 1595634764.029,  
    "ProviderType": "CodeCommit",  
    "CreatedTimeStamp": 1595634764.029,  
    "Owner": "544120495673",  
    "State": "Associating",  
    "StateReason": "Pending Repository Association",  
    "AssociationArn": "arn:aws:codeguru-reviewer:us-  
west-2:544120495673:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat asosiasi AWS CodeCommit repositori di Amazon CodeGuru Reviewer di Panduan](#) Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [AssociateRepository](#) di Referensi AWS CLI Perintah.

## **create-code-review**

Contoh kode berikut menunjukkan cara menggunakan `create-code-review`.

### AWS CLI

Untuk membuat tinjauan kode.

Berikut ini `create-code-review` membuat review kode di mainline cabang AWS CodeCommit repositori yang diberi nama. `my-repository-name`

```
aws codeguru-reviewer create-code-review \  
  --name my-code-review \  
  --repository CodeCommit={Name=mySampleRepo}
```

```
--repository-association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
--type '{"RepositoryAnalysis": {"RepositoryHead": {"BranchName": "mainline"}}}'
```

### Output:

```
{
  "CodeReview": {
    "Name": "my-code-review",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222:code-
review:RepositoryAnalysis-my-code-review",
    "RepositoryName": "my-repository-name",
    "Owner": "123456789012",
    "ProviderType": "CodeCommit",
    "State": "Pending",
    "StateReason": "CodeGuru Reviewer has received the request, and a code
review is scheduled.",
    "CreatedTimeStamp": 1618873489.195,
    "LastUpdatedTimeStamp": 1618873489.195,
    "Type": "RepositoryAnalysis",
    "SourceCodeType": {
      "RepositoryHead": {
        "BranchName": "mainline"
      }
    },
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat ulasan kode di Amazon CodeGuru Reviewer](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [CreateCodeReview](#) di Referensi AWS CLI Perintah.

## describe-code-review

Contoh kode berikut menunjukkan cara menggunakan `describe-code-review`.

### AWS CLI

Daftar detail tentang tinjauan kode.

Berikut ini `describe-code-review` mencantumkan informasi tentang peninjauan kode di cabang "mainline" dari AWS CodeCommit repositori yang diberi nama "". my-repo-name

```
aws codeguru-reviewer put-recommendation-feedback \
  --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
  --recommendation-id
3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb \
  --reactions ThumbsUp
```

## Output

```
{
  "CodeReview": {
    "Name": "My-ecs-beta-repo-master-xs6di4kfd4j269dz",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222:code-
review:RepositoryAnalysis-my-repo-name",
    "RepositoryName": "My-ecs-beta-repo",
    "Owner": "123456789012",
    "ProviderType": "CodeCommit",
    "State": "Pending",
    "StateReason": "CodeGuru Reviewer is reviewing the source code.",
    "CreatedTimeStamp": 1618874226.226,
    "LastUpdatedTimeStamp": 1618874233.689,
    "Type": "RepositoryAnalysis",
    "SourceCodeType": {
      "RepositoryHead": {
        "BranchName": "mainline"
      }
    },
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Untuk informasi selengkapnya, [lihat Melihat detail ulasan kode](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [DescribeCodeReview](#) di Referensi AWS CLI Perintah.

## describe-recommendation-feedback

Contoh kode berikut menunjukkan cara menggunakan `describe-recommendation-feedback`.

### AWS CLI

Untuk melihat informasi tentang umpan balik pada rekomendasi

Berikut ini `describe-recommendation-feedback` menampilkan informasi tentang umpan balik pada rekomendasi. Rekomendasi ini memiliki satu ThumbsUp reaksi.

```
aws codeguru-reviewer describe-recommendation-feedback \
  --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
  --recommendation-id
3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb
```

Output:

```
{
  "RecommendationFeedback": {
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678",
    "RecommendationId":
"3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb",
    "Reactions": [
      "ThumbsUp"
    ],
    "UserId": "aws-user-id",
    "CreatedTimeStamp": 1618877070.313,
    "LastUpdatedTimeStamp": 1618877948.881
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat rekomendasi dan memberikan umpan balik](#) dan [Langkah 4: Memberikan umpan balik](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [DescribeRecommendationFeedback](#) di Referensi AWS CLI Perintah.

## describe-repository-association

Contoh kode berikut menunjukkan cara menggunakan `describe-repository-association`.

### AWS CLI

Contoh 1: Untuk mengembalikan informasi tentang asosiasi GitHub repositori

`describe-repository-association` Contoh berikut mengembalikan informasi tentang asosiasi repositori yang menggunakan repositori GitHub Enterprise dan berada dalam keadaan `Associated`

```
aws codeguru-reviewer describe-repository-association \
  --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
  "RepositoryAssociation": {
    "AssociationId": "b822717e-0711-4e8a-bada-0e738289c75e",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1588102637.649,
    "ProviderType": "GitHub",
    "CreatedTimeStamp": 1588102615.636,
    "Owner": "sample-owner",
    "State": "Associated",
    "StateReason": "Pull Request Notification configuration successful",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat asosiasi repositori Server GitHub Perusahaan di Amazon CodeGuru Reviewer di Panduan Pengguna Amazon CodeGuru Reviewer](#).

Contoh 2: Untuk mengembalikan informasi tentang asosiasi repositori yang gagal

`describe-repository-association` Contoh berikut mengembalikan informasi tentang asosiasi repositori yang menggunakan repositori GitHub Enterprise dan berada dalam keadaan `Failed`



```
aws codeguru-reviewer describe-repository-association \  
  --association-arn arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

### Output:

```
{  
  "RepositoryAssociation": {  
    "ProviderType": "GitHubEnterpriseServer",  
    "Name": "mySampleRepo",  
    "LastUpdatedTimeStamp": 1596217036.892,  
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "CreatedTimeStamp": 1596216896.979,  
    "ConnectionArn": "arn:aws:codestar-connections:us-  
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
    "State": "Failed",  
    "StateReason": "Failed, Please retry.",  
    "AssociationArn": "arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
    "Owner": "sample-owner"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat asosiasi repositori Server GitHub Perusahaan di Amazon CodeGuru Reviewer di Panduan](#) Pengguna Amazon CodeGuru Reviewer.

Contoh 3: Untuk mengembalikan informasi tentang asosiasi repositori disosiasi

`describe-repository-association` Contoh berikut mengembalikan informasi tentang asosiasi repositori yang menggunakan repositori GitHub Enterprise dan berada dalam keadaan `Disassociating`

```
aws codeguru-reviewer describe-repository-association \  
  --association-arn arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

### Output:

```
{  
  "RepositoryAssociation": {  
    "ProviderType": "GitHubEnterpriseServer",  
    "Name": "mySampleRepo",
```

```

    "LastUpdatedTimeStamp": 1596217036.892,
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "State": "Disassociating",
    "StateReason": "Source code access removal in progress",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "sample-owner"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat asosiasi repositori Server GitHub Perusahaan di Amazon CodeGuru Reviewer di Panduan](#) Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [DescribeRepositoryAssociation](#) di Referensi AWS CLI Perintah.

## disassociate-repository

Contoh kode berikut menunjukkan cara menggunakan `disassociate-repository`.

### AWS CLI

Untuk memisahkan asosiasi repositori

Berikut ini `disassociate-repository` memisahkan asosiasi repositori yang menggunakan repositori. AWS CodeCommit

```

aws codeguru-reviewer disassociate-repository \
  --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Output:

```

{
  "RepositoryAssociation": {
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Name": "my-repository",
    "Owner": "123456789012",
    "ProviderType": "CodeCommit",
  }
}

```

```

    "State": "Disassociating",
    "LastUpdatedTimeStamp": 1618939174.759,
    "CreatedTimeStamp": 1595636947.096
  },
  "Tags": {
    "Status": "Secret",
    "Team": "Saanvi"
  }
}

```

Untuk informasi selengkapnya, lihat [Memutuskan repositori di CodeGuru Reviewer di Panduan Pengguna Amazon CodeGuru Reviewer](#).

- Untuk detail API, lihat [DisassociateRepository](#) di Referensi AWS CLI Perintah.

## list-code-reviews

Contoh kode berikut menunjukkan cara menggunakan `list-code-reviews`.

### AWS CLI

Untuk membuat daftar ulasan kode yang dibuat di AWS akun Anda dalam 90 hari terakhir.

`list-code-reviews` Contoh berikut mencantumkan ulasan kode yang dibuat dalam 90 hari terakhir menggunakan permintaan tarik.

```

aws codeguru-reviewer list-code-reviews \
  --type PullRequest

```

### Output:

```

{
  "CodeReviewSummaries": [
    {
      "LastUpdatedTimeStamp": 1588897288.054,
      "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProviderType": "GitHub",
      "PullRequestId": "5",
      "MetricsSummary": {
        "MeteredLinesOfCodeCount": 24,
        "FindingsCount": 1
      },
      "CreatedTimeStamp": 1588897068.512,

```

```
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588869793.263,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "ProviderType": "GitHub",
    "PullRequestId": "4",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 29,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588869575.949,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588870511.211,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "ProviderType": "GitHub",
    "PullRequestId": "4",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 2,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588870292.425,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588118522.452,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
```

```
    "ProviderType": "GitHub",
    "PullRequestId": "3",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 29,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588118301.131,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588112205.207,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "ProviderType": "GitHub",
    "PullRequestId": "2",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 25,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588111987.443,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588104489.981,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "ProviderType": "GitHub",
    "PullRequestId": "1",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 25,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588104270.223,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
```

```

        "Owner": "sample-owner",
        "RepositoryName": "sample-repository-name",
        "Type": "PullRequest"
    }
]
}

```

Untuk informasi selengkapnya, [lihat Melihat semua ulasan kode](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [ListCodeReviews](#) di Referensi AWS CLI Perintah.

## list-recommendation-feedback

Contoh kode berikut menunjukkan cara menggunakan `list-recommendation-feedback`.

### AWS CLI

Untuk membuat daftar umpan balik rekomendasi pelanggan untuk rekomendasi pada repositori terkait

Berikut ini `list-recommendation-feedback` mencantumkan umpan balik pelanggan tentang semua rekomendasi pada tinjauan kode. Ulasan kode ini memiliki satu umpan balik, "ThumbsUp", dari pelanggan.

```

aws codeguru-reviewer list-recommendation-feedback \
  --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678

```

### Output:

```

{
  "RecommendationFeedbackSummaries": [
    {
      "RecommendationId":
"3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb",
      "Reactions": [
        "ThumbsUp"
      ],
      "UserId": "aws-user-id"
    }
  ]
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Langkah 4: Memberikan masukan](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [ListRecommendationFeedback](#) di Referensi AWS CLI Perintah.

## list-recommendations

Contoh kode berikut menunjukkan cara menggunakan `list-recommendations`.

### AWS CLI

Untuk membuat daftar rekomendasi untuk peninjauan kode yang lengkap

`list-recommendations` Contoh berikut mencantumkan rekomendasi untuk tinjauan kode lengkap. Ulasan kode ini memiliki satu rekomendasi.

```
aws codeguru-reviewer list-recommendations \
  --code-review-arn arn:aws:codeguru-reviewer:us-west-2:544120495673:code-
  review:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

### Output:

```
{
  "RecommendationSummaries": [
    {
      "Description": "\n\nProblem \n You are using a `ConcurrentHashMap`,
but your usage of `containsKey()` and `get()` may not be thread-safe at lines: **63
and 64**. In between the check and the `get()` another thread can remove the key
and the `get()` will return `null`. The remove that can remove the key is at line:
**59**.\n\nFix \n Consider calling `get()`, checking instead of your current
check if the returned object is `null`, and then using that object only, without
calling `get()` again.\n\nMore info \n [View an example on GitHub](https://
github.com/apache/hadoop/blob/f16cf877e565084c66bc63605659b157c4394dc8/hadoop-tools/
hadoop-aws/src/main/java/org/apache/hadoop/fs/s3a/s3guard/S3Guard.java#L302-L304)
(external link).",
      "RecommendationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "StartLine": 63,
      "EndLine": 64,
      "FilePath": "src/main/java/com/company/sample/application/
CreateOrderThread.java"
```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Langkah 4: Memberikan masukan](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [ListRecommendations](#) di Referensi AWS CLI Perintah.

## list-repository-associations

Contoh kode berikut menunjukkan cara menggunakan `list-repository-associations`.

### AWS CLI

Untuk membuat daftar asosiasi repositori di akun Anda AWS

`list-repository-associations` Contoh berikut mengembalikan daftar objek ringkasan asosiasi repositori di akun Anda. Anda dapat memfilter daftar yang dikembalikan dengan `ProviderType`, `Name`, `State`, dan `Owner`.

```
aws codeguru-reviewer list-repository-associations
```

### Output:

```

{
  "RepositoryAssociationSummaries": [
    {
      "LastUpdatedTimeStamp": 1595886609.616,
      "Name": "test",
      "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Owner": "sample-owner",
      "State": "Associated",
      "AssociationArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProviderType": "Bitbucket"
    },
    {
      "LastUpdatedTimeStamp": 1595636969.035,
      "Name": "CodeDeploy-CodePipeline-ECS-Tutorial",
      "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "Owner": "123456789012",
      "State": "Associated",
    }
  ]
}

```



```
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "ProviderType": "CodeCommit"
  },
  {
    "LastUpdatedTimeStamp": 1595634785.983,
    "Name": "My-ecs-beta-repo",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "123456789012",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "ProviderType": "CodeCommit"
  },
  {
    "LastUpdatedTimeStamp": 1590712811.77,
    "Name": "MyTestCodeCommit",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "Owner": "123456789012",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "ProviderType": "CodeCommit"
  },
  {
    "LastUpdatedTimeStamp": 1588102637.649,
    "Name": "aws-codeguru-profiler-sample-application",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "Owner": "sample-owner",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "ProviderType": "GitHub"
  },
  {
    "LastUpdatedTimeStamp": 1588028233.995,
    "Name": "codeguru-profiler-demo-app",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "Owner": "sample-owner",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "ProviderType": "GitHub"
  }
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Melihat semua asosiasi repositori di CodeGuru Reviewer di Panduan Pengguna Amazon CodeGuru Reviewer](#).

- Untuk detail API, lihat [ListRepositoryAssociations](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag pada repositori terkait

Berikut ini `list-tags-for-resource` mencantumkan tag pada repositori terkait. Repositori terkait ini memiliki dua tag.

```
aws codeguru-reviewer list-tags-for-resource \
  --resource-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
  "Tags": {
    "Status": "Secret",
    "Team": "Saanvi"
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat tag untuk CodeGuru repositori terkait Reviewer \(AWS CLI\) di Panduan Pengguna Amazon Reviewer](#). CodeGuru

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-recommendation-feedback

Contoh kode berikut menunjukkan cara menggunakan `put-recommendation-feedback`.

## AWS CLI

Untuk menambahkan rekomendasi ke tinjauan kode

Berikut ini `put-recommendation-feedback` menempatkan ThumbsUp rekomendasi pada tinjauan kode.

```
aws codeguru-reviewer put-recommendation-feedback \  
  --code-review-arn \arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-  
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \  
  --recommendation-id  
3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb \  
  --reactions ThumbsUp
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 4: Memberikan masukan](#) di Panduan Pengguna Amazon CodeGuru Reviewer.

- Untuk detail API, lihat [PutRecommendationFeedback](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menambahkan tag ke repositori terkait

Berikut ini `tag-resource` menambahkan dua tag ke repositori terkait

```
aws codeguru-reviewer tag-resource \  
  --resource-arn arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --tags Status=Secret,Team=Saarvi
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan tag ke CodeGuru repositori terkait Reviewer \(AWS CLI\)](#) dan [Menambahkan atau memperbarui tag untuk repositori terkait CodeGuru Reviewer \(AWS CLI\)](#) di [Panduan Pengguna Amazon Reviewer](#). CodeGuru

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag repositori terkait

Berikut ini `untag-resource` menghapus dua tag dengan kunci “Rahasia” dan “Tim” dari repositori terkait.

```
aws codeguru-reviewer untag-resource \  
  --resource-arn arn:aws:codeguru-reviewer:us-\  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --tag-keys Status Team
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus tag dari CodeGuru repositori terkait Reviewer \(AWS CLI\) di](#) Panduan Pengguna Amazon Reviewer. CodeGuru

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## CodePipeline contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CodePipeline.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **acknowledge-job**

Contoh kode berikut menunjukkan cara menggunakan `acknowledge-job`.

#### AWS CLI

Untuk mengambil informasi tentang pekerjaan tertentu

Contoh ini mengembalikan informasi tentang pekerjaan tertentu, termasuk status pekerjaan itu jika ada. Ini hanya digunakan untuk pekerja kerja dan tindakan khusus. Untuk menentukan nilai nonce dan ID pekerjaan, gunakan `aws poll-for-jobs codepipeline`.

Perintah:

```
aws codepipeline acknowledge-job --job-id f4f4ff82-2d11-EXAMPLE --nonce 3
```

Output:

```
{
  "status": "InProgress"
}
```

- Untuk detail API, lihat [AcknowledgeJob](#) di Referensi AWS CLI Perintah.

### **create-custom-action-type**

Contoh kode berikut menunjukkan cara menggunakan `create-custom-action-type`.

#### AWS CLI

Untuk membuat tindakan kustom

Contoh ini membuat tindakan kustom untuk AWS CodePipeline menggunakan file JSON yang sudah dibuat (di sini bernama `MyCustomAction.json`) yang berisi struktur tindakan kustom. Untuk informasi selengkapnya tentang persyaratan untuk membuat tindakan kustom, termasuk struktur file, lihat Panduan AWS CodePipeline Pengguna.

```
aws codepipeline create-custom-action-type --cli-input-json file://  
MyCustomAction.json
```

Isi file `MyCustomAction.json` JSON:

```
{  
  "category": "Build",  
  "provider": "MyJenkinsProviderName",  
  "version": "1",  
  "settings": {  
    "entityUrlTemplate": "https://192.0.2.4/job/{Config:ProjectName}/",  
    "executionUrlTemplate": "https://192.0.2.4/job/{Config:ProjectName}/  
lastSuccessfulBuild/{ExternalExecutionId}/"  
  },  
  "configurationProperties": [  
    {  
      "name": "MyJenkinsExampleBuildProject",  
      "required": true,  
      "key": true,  
      "secret": false,  
      "queryable": false,  
      "description": "The name of the build project must be provided when this  
action is added to the pipeline.",  
      "type": "String"  
    }  
  ],  
  "inputArtifactDetails": {  
    "maximumCount": 1,  
    "minimumCount": 0  
  },  
  "outputArtifactDetails": {  
    "maximumCount": 1,  
    "minimumCount": 0  
  }  
}
```

Perintah ini mengembalikan struktur tindakan kustom.

- Untuk detail API, lihat [CreateCustomActionType](#) di Referensi AWS CLI Perintah.

## create-pipeline

Contoh kode berikut menunjukkan cara menggunakan `create-pipeline`.

### AWS CLI

Untuk membuat pipa

Contoh ini membuat pipeline dalam AWS CodePipeline menggunakan file JSON yang sudah dibuat (di sini bernama `MySecondPipeline.json`) yang berisi struktur pipeline. Untuk informasi selengkapnya tentang persyaratan untuk membuat pipeline, termasuk struktur file, lihat [Panduan AWS CodePipeline Pengguna](#).

Perintah:

```
aws codepipeline create-pipeline --cli-input-json file://MySecondPipeline.json
```

Isi sampel file JSON:

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "MyApp"
              }
            ],
            "configuration": {
              "S3Bucket": "awscodepipeline-demo-bucket",
              "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
            }
          }
        ]
      }
    ]
  }
}
```

```
    },
    "runOrder": 1
  }
]
},
{
  "name": "Beta",
  "actions": [
    {
      "inputArtifacts": [
        {
          "name": "MyApp"
        }
      ],
      "name": "CodePipelineDemoFleet",
      "actionTypeId": {
        "category": "Deploy",
        "owner": "AWS",
        "version": "1",
        "provider": "CodeDeploy"
      },
      "outputArtifacts": [],
      "configuration": {
        "ApplicationName": "CodePipelineDemoApplication",
        "DeploymentGroupName": "CodePipelineDemoFleet"
      },
      "runOrder": 1
    }
  ]
}
],
"artifactStore": {
  "type": "S3",
  "location": "codepipeline-us-east-1-11EXAMPLE11"
},
"name": "MySecondPipeline",
"version": 1
}
}
```

### Output:

This command returns the structure of the pipeline.



- Untuk detail API, lihat [CreatePipeline](#) di Referensi AWS CLI Perintah.

## delete-custom-action-type

Contoh kode berikut menunjukkan cara menggunakan `delete-custom-action-type`.

### AWS CLI

Untuk menghapus tindakan kustom

Contoh ini menghapus tindakan kustom AWS CodePipeline dengan menggunakan file JSON yang sudah dibuat (di sini bernama `DeleteMyCustomAction.json`) yang berisi tipe tindakan, nama penyedia, dan nomor versi tindakan yang akan dihapus. Gunakan `list-action-types` perintah untuk melihat nilai yang benar untuk kategori, versi, dan penyedia.

Perintah:

```
aws codepipeline delete-custom-action-type --cli-input-json file://DeleteMyCustomAction.json
```

Isi sampel file JSON:

```
{
  "category": "Build",
  "version": "1",
  "provider": "MyJenkinsProviderName"
}
```

Output:

```
None.
```

- Untuk detail API, lihat [DeleteCustomActionType](#) di Referensi AWS CLI Perintah.

## delete-pipeline

Contoh kode berikut menunjukkan cara menggunakan `delete-pipeline`.

### AWS CLI

Untuk menghapus pipa

Contoh ini menghapus pipeline bernama MySecondPipeline from AWS CodePipeline. Gunakan perintah list-pipelines untuk melihat daftar pipeline yang terkait dengan akun Anda. AWS

Perintah:

```
aws codepipeline delete-pipeline --name MySecondPipeline
```

Output:

```
None.
```

- Untuk detail API, lihat [DeletePipeline](#) di Referensi AWS CLI Perintah.

## delete-webhook

Contoh kode berikut menunjukkan cara menggunakan delete-webhook.

### AWS CLI

Untuk menghapus webhook

delete-webhook Contoh berikut menghapus webhook untuk tindakan sumber GitHub versi 1. Anda harus menggunakan deregister-webhook-with-third-party perintah untuk membatalkan pendaftaran webhook sebelum Anda menghapusnya.

```
aws codepipeline delete-webhook \  
  --name my-webhook
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus webhook untuk GitHub sumber Anda](#) di Panduan AWS CodePipeline Pengguna.

- Untuk detail API, lihat [DeleteWebhook](#) di Referensi AWS CLI Perintah.

## deregister-webhook-with-third-party

Contoh kode berikut menunjukkan cara menggunakan deregister-webhook-with-third-party.

## AWS CLI

Untuk membatalkan pendaftaran webhook

`deregister-webhook-with-third-party` Contoh berikut menghapus webhook untuk tindakan sumber GitHub versi 1. Anda harus membatalkan pendaftaran webhook sebelum Anda menghapusnya.

```
aws codepipeline deregister-webhook-with-third-party \  
  --webhook-name my-webhook
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus webhook untuk GitHub sumber Anda](#) di Panduan AWS CodePipeline Pengguna.

- Untuk detail API, lihat [DeregisterWebhookWithThirdParty](#) di Referensi AWS CLI Perintah.

## **disable-stage-transition**

Contoh kode berikut menunjukkan cara menggunakan `disable-stage-transition`.

## AWS CLI

Untuk menonaktifkan transisi ke tahap dalam pipa

Contoh ini menonaktifkan transisi ke tahap Beta dari MyFirstPipeline pipeline di AWS CodePipeline

Perintah:

```
aws codepipeline disable-stage-transition --pipeline-name MyFirstPipeline --stage-  
name Beta --transition-type Inbound
```

Output:

```
None .
```

- Untuk detail API, lihat [DisableStageTransition](#) di Referensi AWS CLI Perintah.

## enable-stage-transition

Contoh kode berikut menunjukkan cara menggunakan `enable-stage-transition`.

### AWS CLI

Untuk mengaktifkan transisi ke tahap dalam pipa

Contoh ini memungkinkan transisi ke tahap Beta dari MyFirstPipeline pipeline di AWS CodePipeline.

Perintah:

```
aws codepipeline enable-stage-transition --pipeline-name MyFirstPipeline --stage-name Beta --transition-type Inbound
```

Output:

```
None .
```

- Untuk detail API, lihat [EnableStageTransition](#) di Referensi AWS CLI Perintah.

## get-job-details

Contoh kode berikut menunjukkan cara menggunakan `get-job-details`.

### AWS CLI

Untuk mendapatkan detail pekerjaan

Contoh ini mengembalikan rincian tentang pekerjaan yang ID diwakili oleh F4f4FF82-2D11-Example. Perintah ini hanya digunakan untuk tindakan kustom. Saat perintah ini dipanggil, AWS CodePipeline mengembalikan kredensi sementara untuk bucket Amazon S3 yang digunakan untuk menyimpan artefak untuk pipeline, jika diperlukan untuk tindakan kustom. Perintah ini juga akan mengembalikan nilai rahasia yang ditentukan untuk tindakan, jika ada yang ditentukan.

Perintah:

```
aws codepipeline get-job-details --job-id f4f4ff82-2d11-EXAMPLE
```

## Output:

```

{
  "jobDetails": {
    "accountId": "111111111111",
    "data": {
      "actionConfiguration": {
        "__type": "ActionConfiguration",
        "configuration": {
          "ProjectName": "MyJenkinsExampleTestProject"
        }
      },
      "actionTypeId": {
        "__type": "ActionTypeId",
        "category": "Test",
        "owner": "Custom",
        "provider": "MyJenkinsProviderName",
        "version": "1"
      },
      "artifactCredentials": {
        "__type": "AWSSessionCredentials",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFfiCYEXAMPLEKEY",
        "sessionToken":
          "fICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMakGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwd
          +a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/
          f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/
          MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpEibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQ
          +auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs
      },
      "inputArtifacts": [
        {
          "__type": "Artifact",
          "location": {
            "s3Location": {
              "bucketName": "codepipeline-us-east-1-11EXAMPLE11",
              "objectKey": "MySecondPipeline/MyAppBuild/EXAMPLE"
            },
            "type": "S3"
          },
          "name": "MyAppBuild"
        }
      ],
      "outputArtifacts": [],

```

```
"pipelineContext": {
  "__type": "PipelineContext",
  "action": {
    "name": "MyJenkinsTest-Action"
  },
  "pipelineName": "MySecondPipeline",
  "stage": {
    "name": "Testing"
  }
}
},
"id": "f4f4ff82-2d11-EXAMPLE"
}
}
```

- Untuk detail API, lihat [GetJobDetails](#) di Referensi AWS CLI Perintah.

## get-pipeline-state

Contoh kode berikut menunjukkan cara menggunakan `get-pipeline-state`.

### AWS CLI

Untuk mendapatkan informasi tentang keadaan pipa

Contoh ini mengembalikan status terbaru dari pipeline bernama `MyFirstPipeline`.

Perintah:

```
aws codepipeline get-pipeline-state --name MyFirstPipeline
```

Output:

```
{
  "created": 1446137312.204,
  "pipelineName": "MyFirstPipeline",
  "pipelineVersion": 1,
  "stageStates": [
    {
      "actionStates": [
        {
          "actionName": "Source",
```

```

    "entityUrl": "https://console.aws.amazon.com/s3/home?#",
    "latestExecution": {
      "lastStatusChange": 1446137358.328,
      "status": "Succeeded"
    }
  ],
  "stageName": "Source"
},
{
  "actionStates": [
    {
      "actionName": "CodePipelineDemoFleet",
      "entityUrl": "https://console.aws.amazon.com/codedeploy/home?#/applications/CodePipelineDemoApplication/deployment-groups/CodePipelineDemoFleet",
      "latestExecution": {
        "externalExecutionId": "d-EXAMPLE",
        "externalExecutionUrl": "https://console.aws.amazon.com/codedeploy/home?#/deployments/d-EXAMPLE",
        "lastStatusChange": 1446137493.131,
        "status": "Succeeded",
        "summary": "Deployment Succeeded"
      }
    }
  ],
  "inboundTransitionState": {
    "enabled": true
  },
  "stageName": "Beta"
}
],
"updated": 1446137312.204
}

```

- Untuk detail API, lihat [GetPipelineState](#) di Referensi AWS CLI Perintah.

## get-pipeline

Contoh kode berikut menunjukkan cara menggunakan `get-pipeline`.

### AWS CLI

Untuk melihat struktur pipa

Contoh ini mengembalikan struktur pipa bernama MyFirstPipeline.

Perintah:

```
aws codepipeline get-pipeline --name MyFirstPipeline
```

Output:

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "MyApp"
              }
            ],
            "configuration": {
              "S3Bucket": "awscodepipeline-demo-bucket",
              "S3ObjectKey": "aws-codepipeline-s3-aws-
codedeploy_linux.zip"
            },
            "runOrder": 1
          }
        ]
      },
      {
        "name": "Beta",
        "actions": [
          {
            "inputArtifacts": [
```



```

        {
            "name": "MyApp"
        }
    ],
    "name": "CodePipelineDemoFleet",
    "actionTypeId": {
        "category": "Deploy",
        "owner": "AWS",
        "version": "1",
        "provider": "CodeDeploy"
    },
    "outputArtifacts": [],
    "configuration": {
        "ApplicationName": "CodePipelineDemoApplication",
        "DeploymentGroupName": "CodePipelineDemoFleet"
    },
    "runOrder": 1
}
]
}
],
"artifactStore": {
    "type": "S3",
    "location": "codepipeline-us-east-1-11EXAMPLE11"
},
"name": "MyFirstPipeline",
"version": 1
}
}

```

- Untuk detail API, lihat [GetPipeline](#) di Referensi AWS CLI Perintah.

## list-action-executions

Contoh kode berikut menunjukkan cara menggunakan `list-action-executions`.

### AWS CLI

Untuk membuat daftar eksekusi tindakan

`list-action-executions` Contoh berikut menampilkan detail eksekusi tindakan untuk pipeline, seperti ID eksekusi tindakan, artefak input, artefak keluaran, hasil eksekusi, dan status.

```
aws codepipeline list-action-executions \  
  --pipeline-name myPipeline
```

### Output:

```
{  
  "actionExecutionDetails": [  
    {  
      "pipelineExecutionId": "EXAMPLE0-adfc-488e-bf4c-1111111720d3",  
      "actionExecutionId": "EXAMPLE4-2ee8-4853-bd6a-111111158148",  
      "pipelineVersion": 12,  
      "stageName": "Deploy",  
      "actionName": "Deploy",  
      "startTime": 1598572628.6,  
      "lastUpdateTime": 1598572661.255,  
      "status": "Succeeded",  
      "input": {  
        "actionTypeId": {  
          "category": "Deploy",  
          "owner": "AWS",  
          "provider": "CodeDeploy",  
          "version": "1"  
        },  
        "configuration": {  
          "ApplicationName": "my-application",  
          "DeploymentGroupName": "my-deployment-group"  
        },  
        "resolvedConfiguration": {  
          "ApplicationName": "my-application",  
          "DeploymentGroupName": "my-deployment-group"  
        },  
        "region": "us-east-1",  
        "inputArtifacts": [  
          {  
            "name": "SourceArtifact",  
            "s3location": {  
              "bucket": "artifact-bucket",  
              "key": "myPipeline/SourceArti/key"  
            }  
          }  
        ],  
        "namespace": "DeployVariables"  
      },  
    ],  
  },  
}
```

```
    "output": {
      "outputArtifacts": [],
      "executionResult": {
        "externalExecutionId": "d-EXAMPLEE5",
        "externalExecutionSummary": "Deployment Succeeded",
        "externalExecutionUrl": "https://myaddress.com"
      },
      "outputVariables": {}
    }
  },
  {
    "pipelineExecutionId": "EXAMPLE0-adfc-488e-bf4c-1111111720d3",
    "actionExecutionId": "EXAMPLE5-abb4-4192-9031-11111113a7b0",
    "pipelineVersion": 12,
    "stageName": "Source",
    "actionName": "Source",
    "startTime": 1598572624.387,
    "lastUpdateTime": 1598572628.16,
    "status": "Succeeded",
    "input": {
      "actionTypeId": {
        "category": "Source",
        "owner": "AWS",
        "provider": "CodeCommit",
        "version": "1"
      },
      "configuration": {
        "BranchName": "production",
        "PollForSourceChanges": "false",
        "RepositoryName": "my-repo"
      },
      "resolvedConfiguration": {
        "BranchName": "production",
        "PollForSourceChanges": "false",
        "RepositoryName": "my-repo"
      },
      "region": "us-east-1",
      "inputArtifacts": [],
      "namespace": "SourceVariables"
    },
    "output": {
      "outputArtifacts": [
        {
          "name": "SourceArtifact",
```

```

        "s3location": {
            "bucket": "my-bucket",
            "key": "myPipeline/SourceArti/key"
        }
    },
    "executionResult": {
        "externalExecutionId":
"11111111ad99dcd35914c00b7fbea13995EXAMPLE",
        "externalExecutionSummary": "Edited template.yml",
        "externalExecutionUrl": "https://myaddress.com"
    },
    "outputVariables": {
        "AuthorDate": "2020-05-08T17:45:43Z",
        "BranchName": "production",
        "CommitId": "EXAMPLEad99dcd35914c00b7fbea139951111111",
        "CommitMessage": "Edited template.yml",
        "CommitterDate": "2020-05-08T17:45:43Z",
        "RepositoryName": "my-repo"
    }
},
. . . .

```

Untuk informasi selengkapnya, [lihat Melihat eksekusi tindakan \(CLI\)](#) di Panduan Pengguna AWS CodePipeline .

- Untuk detail API, lihat [ListActionExecutions](#) di Referensi AWS CLI Perintah.

## list-action-types

Contoh kode berikut menunjukkan cara menggunakan `list-action-types`.

### AWS CLI

Untuk melihat jenis tindakan yang tersedia

Digunakan dengan sendirinya, `list-action-types` perintah mengembalikan struktur semua tindakan yang tersedia untuk AWS akun Anda. Contoh ini menggunakan `action-owner-filter` opsi `--` untuk mengembalikan hanya tindakan khusus.

Perintah:

```
aws codepipeline list-action-types --action-owner-filter Custom
```

**Output:**

```
{
  "actionTypes": [
    {
      "inputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
      },
      "actionConfigurationProperties": [
        {
          "secret": false,
          "required": true,
          "name": "MyJenkinsExampleBuildProject",
          "key": true,
          "queryable": true
        }
      ],
      "outputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
      },
      "id": {
        "category": "Build",
        "owner": "Custom",
        "version": "1",
        "provider": "MyJenkinsProviderName"
      },
      "settings": {
        "entityUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}",
        "executionUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}/
{ExternalExecutionId}"
      }
    },
    {
      "inputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
      },
      "actionConfigurationProperties": [
        {
```

```

        "secret": false,
        "required": true,
        "name": "MyJenkinsExampleTestProject",
        "key": true,
        "queryable": true
    }
],
"outputArtifactDetails": {
    "maximumCount": 5,
    "minimumCount": 0
},
"id": {
    "category": "Test",
    "owner": "Custom",
    "version": "1",
    "provider": "MyJenkinsProviderName"
},
"settings": {
    "entityUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}",
    "executionUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}/
{ExternalExecutionId}"
}
}
]
}

```

- Untuk detail API, lihat [ListActionTypes](#) di Referensi AWS CLI Perintah.

## list-pipeline-executions

Contoh kode berikut menunjukkan cara menggunakan `list-pipeline-executions`.

### AWS CLI

Untuk melihat riwayat eksekusi pipeline

`list-pipeline-executions` Contoh berikut menunjukkan riwayat eksekusi pipeline untuk pipeline di AWS akun Anda.

```
aws codepipeline list-pipeline-executions \
  --pipeline-name MyPipeline
```

Output:

```
{
  "pipelineExecutionSummaries": [
    {
      "lastUpdateTime": 1496380678.648,
      "pipelineExecutionId": "7cf7f7cb-3137-539g-j458-d7eu3EXAMPLE",
      "startTime": 1496380258.243,
      "status": "Succeeded"
    },
    {
      "lastUpdateTime": 1496591045.634,
      "pipelineExecutionId": "3137f7cb-8d494hj4-039j-d84l-d7eu3EXAMPLE",
      "startTime": 1496590401.222,
      "status": "Succeeded"
    },
    {
      "lastUpdateTime": 1496946071.6456,
      "pipelineExecutionId": "4992f7jf-7cf7-913k-k334-d7eu3EXAMPLE",
      "startTime": 1496945471.5645,
      "status": "Succeeded"
    }
  ]
}
```

Untuk informasi selengkapnya, [lihat Melihat riwayat eksekusi](#) di Panduan AWS CodePipeline Pengguna.

- Untuk detail API, lihat [ListPipelineExecutions](#) di Referensi AWS CLI Perintah.

## list-pipelines

Contoh kode berikut menunjukkan cara menggunakan `list-pipelines`.

### AWS CLI

Untuk melihat daftar saluran pipa

Contoh ini mencantumkan semua AWS CodePipeline pipeline yang terkait dengan AWS akun pengguna.

Perintah:

```
aws codepipeline list-pipelines
```

**Output:**

```
{
  "pipelines": [
    {
      "updated": 1439504274.641,
      "version": 1,
      "name": "MyFirstPipeline",
      "created": 1439504274.641
    },
    {
      "updated": 1436461837.992,
      "version": 2,
      "name": "MySecondPipeline",
      "created": 1436460801.381
    }
  ]
}
```

- Untuk detail API, lihat [ListPipelines](#) di Referensi AWS CLI Perintah.

**list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

**AWS CLI**

Untuk daftar tag

`list-tags-for-resource` Contoh berikut mengambil daftar semua tag yang dilampirkan ke sumber pipa yang ditentukan.

```
aws codepipeline list-tags-for-resource \
  --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline
```

**Output:**

```
{
  "tags": {
    "Project": "ProjectA",
    "IscontainerBased": "true"
  }
}
```



```
}
```

Untuk informasi selengkapnya, [lihat Melihat tag untuk pipeline \(CLI\)](#) di AWS CodePipeline Panduan Pengguna.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## list-webhooks

Contoh kode berikut menunjukkan cara menggunakan `list-webhooks`.

### AWS CLI

Untuk daftar webhooks

`list-webhooks` Contoh berikut mengambil daftar semua tag yang dilampirkan ke sumber pipa yang ditentukan.

```
aws codepipeline list-webhooks \
  --endpoint-url "https://codepipeline.eu-central-1.amazonaws.com" \
  --region "eu-central-1"
```

Output:

```
{
  "webhooks": [
    {
      "url": "https://webhooks.domain.com/
trigger1111111111EXAMPLE1111111111111111111": {
        "authenticationConfiguration": {
          "SecretToken": "Secret"
        },
        "name": "my-webhook",
        "authentication": "GITHUB_HMAC",
        "targetPipeline": "my-Pipeline",
        "targetAction": "Source",
        "filters": [
          {
            "jsonPath": "$.ref",
            "matchEquals": "refs/heads/{Branch}"
          }
        ]
      }
    ]
  }
```

```
    },
    "arn": "arn:aws:codepipeline:eu-central-1:123456789012:webhook:my-
webhook"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Daftar webhook di akun Anda](#) di AWS CodePipeline Panduan Pengguna.

- Untuk detail API, lihat [ListWebhooks](#) di Referensi AWS CLI Perintah.

## poll-for-jobs

Contoh kode berikut menunjukkan cara menggunakan `poll-for-jobs`.

### AWS CLI

Untuk melihat pekerjaan yang tersedia

Contoh ini mengembalikan informasi tentang pekerjaan apa pun untuk ditindaklanjuti oleh pekerja kerja. Contoh ini menggunakan file JSON yang telah ditentukan sebelumnya (`MyActionTypeInfo.json`) untuk memberikan informasi tentang jenis tindakan yang pekerja pekerjaan memproses pekerjaan. Perintah ini hanya digunakan untuk tindakan kustom. Saat perintah ini dipanggil, AWS CodePipeline mengembalikan kredensi sementara untuk bucket Amazon S3 yang digunakan untuk menyimpan artefak untuk pipeline. Perintah ini juga akan mengembalikan nilai rahasia yang ditentukan untuk tindakan, jika ada yang ditentukan.

Perintah:

```
aws codepipeline poll-for-jobs --cli-input-json file://MyActionTypeInfo.json
```

Isi sampel file JSON:

```
{
  "actionTypeId": {
    "category": "Test",
    "owner": "Custom",
    "provider": "MyJenkinsProviderName",
    "version": "1"
  },
}
```

```

    "maxBatchSize": 5,
    "queryParam": {
      "ProjectName": "MyJenkinsTestProject"
    }
  }
}

```

### Output:

```

{
  "jobs": [
    {
      "accountId": "111111111111",
      "data": {
        "actionConfiguration": {
          "__type": "ActionConfiguration",
          "configuration": {
            "ProjectName": "MyJenkinsExampleTestProject"
          }
        },
        "actionTypeId": {
          "__type": "ActionTypeId",
          "category": "Test",
          "owner": "Custom",
          "provider": "MyJenkinsProviderName",
          "version": "1"
        },
        "artifactCredentials": {
          "__type": "AWSSessionCredentials",
          "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
          "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
          "sessionToken":
            "fICCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMaKGA1UEBhMCVVMxCzAJBgNVBAGTA1dBMRaWdgYDQVQQHEwd
            +a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/
            f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/
            MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpEibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQ
            +auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs
        },
        "inputArtifacts": [
          {
            "__type": "Artifact",
            "location": {
              "s3Location": {
                "bucketName": "codepipeline-us-east-1-11EXAMPLE11",

```

```

        "objectKey": "MySecondPipeline/MyAppBuild/EXAMPLE"
      },
      "type": "S3"
    },
    "name": "MyAppBuild"
  }
],
"outputArtifacts": [],
"pipelineContext": {
  "__type": "PipelineContext",
  "action": {
    "name": "MyJenkinsTest-Action"
  },
  "pipelineName": "MySecondPipeline",
  "stage": {
    "name": "Testing"
  }
}
},
"id": "ef66c259-64f9-EXAMPLE",
"nonce": "3"
}
]
}

```

- Untuk detail API, lihat [PollForJobs](#) di Referensi AWS CLI Perintah.

## put-webhook

Contoh kode berikut menunjukkan cara menggunakan `put-webhook`.

### AWS CLI

Untuk membuat webhook

`put-webhook` Contoh berikut membuat webhook untuk aksi sumber GitHub versi 1. Setelah Anda membuat webhook, Anda harus menggunakan perintah `register-webhook-with-third-party` untuk mendaftarkannya.

```

aws codepipeline put-webhook \
  --cli-input-json file://webhook_json.json \
  --region "eu-central-1"

```

Isi dari `webhook_json.json`:

```
{
  "webhook": {
    "name": "my-webhook",
    "targetPipeline": "pipeline_name",
    "targetAction": "source_action_name",
    "filters": [
      {
        "jsonPath": "$.ref",
        "matchEquals": "refs/heads/{Branch}"
      }
    ],
    "authentication": "GITHUB_HMAC",
    "authenticationConfiguration": {
      "SecretToken": "secret"
    }
  }
}
```

Output:

```
{
  "webhook": {
    "url": "https://webhooks.domain.com/trigger1111111111EXAMPLE111111111111111111111",
    "definition": {
      "authenticationConfiguration": {
        "SecretToken": "secret"
      },
      "name": "my-webhook",
      "authentication": "GITHUB_HMAC",
      "targetPipeline": "pipeline_name",
      "targetAction": "Source",
      "filters": [
        {
          "jsonPath": "$.ref",
          "matchEquals": "refs/heads/{Branch}"
        }
      ]
    },
    "arn": "arn:aws:codepipeline:eu-central-1:123456789012:webhook:my-webhook"
  },
}
```

```
    "tags": [  
      {  
        "key": "Project",  
        "value": "ProjectA"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat webhook untuk GitHub sumber](#) di Panduan AWS CodePipeline Pengguna.

- Untuk detail API, lihat [PutWebhook](#) di Referensi AWS CLI Perintah.

## retry-stage-execution

Contoh kode berikut menunjukkan cara menggunakan `retry-stage-execution`.

### AWS CLI

Untuk mencoba lagi tindakan yang gagal

`retry-stage-execution` Contoh berikut mencoba ulang tahap yang memiliki tindakan gagal.

```
aws codepipeline retry-stage-execution \  
  --pipeline-name MyPipeline \  
  --stage-name Deploy \  
  --pipeline-execution-id b59babff-5f34-EXAMPLE \  
  --retry-mode FAILED_ACTIONS
```

Output:

```
{  
  "pipelineExecutionId": "b59babff-5f34-EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Coba lagi tindakan yang gagal \(CLI\)](#) di Panduan Pengguna AWS CodePipeline .

- Untuk detail API, lihat [RetryStageExecution](#) di Referensi AWS CLI Perintah.

## start-pipeline-execution

Contoh kode berikut menunjukkan cara menggunakan `start-pipeline-execution`.

### AWS CLI

Untuk menjalankan revisi terbaru melalui pipeline

Contoh ini menjalankan revisi terbaru yang ada di tahap sumber pipa melalui pipa bernama "MyFirstPipeline".

Perintah:

```
aws codepipeline start-pipeline-execution --name MyFirstPipeline
```

Output:

```
{
  "pipelineExecutionId": "3137f7cb-7cf7-EXAMPLE"
}
```

- Untuk detail API, lihat [StartPipelineExecution](#) di Referensi AWS CLI Perintah.

## stop-pipeline-execution

Contoh kode berikut menunjukkan cara menggunakan `stop-pipeline-execution`.

### AWS CLI

Untuk menghentikan eksekusi pipeline

`stop-pipeline-execution` Contoh berikut secara default menunggu hingga tindakan yang sedang berlangsung selesai, dan kemudian menghentikan pipeline. Anda tidak dapat memilih untuk berhenti dan menunggu jika eksekusi sudah dalam status Berhenti. Anda dapat memilih untuk berhenti dan meninggalkan eksekusi yang sudah dalam status Berhenti.

```
aws codepipeline stop-pipeline-execution \
  --pipeline-name MyFirstPipeline \
  --pipeline-execution-id d-EXAMPLE \
  --reason "Stopping pipeline after the build action is done"
```

Perintah ini tidak mengembalikan output.

Untuk informasi selengkapnya, lihat [Menghentikan eksekusi pipeline \(CLI\)](#) di AWS CodePipeline Panduan Pengguna.

- Untuk detail API, lihat [StopPipelineExecution](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut mengaitkan satu set tag yang disediakan dengan pipeline. Gunakan perintah ini untuk menambah atau mengedit tag.

```
aws codepipeline tag-resource \  
  --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline \  
  --tags key=Project,value=ProjectA key=IscontainerBased,value=true
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan tag ke pipeline \(CLI\)](#) di AWS CodePipeline Panduan Pengguna.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus AWS tag dari sumber daya koneksi

`untag-resource` Contoh berikut menghapus tag dari sumber daya yang ditentukan.

```
aws codepipeline untag-resource \  
  --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline \  
  --tag-keys Project IscontainerBased
```

Perintah ini tidak menghasilkan output.



Untuk informasi selengkapnya, lihat [Menghapus tag dari pipeline \(CLI\)](#) di AWS CodePipeline Panduan Pengguna.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-pipeline

Contoh kode berikut menunjukkan cara menggunakan `update-pipeline`.

### AWS CLI

Untuk memperbarui struktur pipa

Contoh ini menggunakan perintah `update-pipeline` dengan argumen `--cli-input-json`. Contoh ini menggunakan file JSON yang telah ditentukan sebelumnya (`MyFirstPipeline.json`) untuk memperbarui struktur pipeline. AWS CodePipeline mengenali nama pipeline yang terkandung dalam file JSON, dan kemudian menerapkan perubahan apa pun dari bidang yang dimodifikasi dalam struktur pipa untuk memperbarui pipeline.

Gunakan panduan berikut saat membuat file JSON yang telah ditentukan sebelumnya:

Jika Anda bekerja dengan struktur pipa yang diambil menggunakan perintah `get-pipeline`, Anda harus menghapus bagian metadata dari struktur pipa di file JSON (baris `"metadata": {}` dan bidang `"dibuat," "PipelineEarn,"` dan `"diperbarui"` di dalamnya). Nama pipeline tidak dapat diubah.

Perintah:

```
aws codepipeline update-pipeline --cli-input-json file://MyFirstPipeline.json
```

Contoh isi file JSON:

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source",
            "actionTypeId": {
```

```
        "category": "Source",
        "owner": "AWS",
        "version": "1",
        "provider": "S3"
    },
    "outputArtifacts": [
        {
            "name": "MyApp"
        }
    ],
    "configuration": {
        "S3Bucket": "awscodepipeline-demo-bucket2",
        "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
    },
    "runOrder": 1
}
]
},
{
    "name": "Beta",
    "actions": [
        {
            "inputArtifacts": [
                {
                    "name": "MyApp"
                }
            ],
            "name": "CodePipelineDemoFleet",
            "actionTypeId": {
                "category": "Deploy",
                "owner": "AWS",
                "version": "1",
                "provider": "CodeDeploy"
            },
            "outputArtifacts": [],
            "configuration": {
                "ApplicationName": "CodePipelineDemoApplication",
                "DeploymentGroupName": "CodePipelineDemoFleet"
            },
            "runOrder": 1
        }
    ]
}
],
```

```
"artifactStore": {
  "type": "S3",
  "location": "codepipeline-us-east-1-11EXAMPLE11"
},
"name": "MyFirstPipeline",
"version": 1
}
}
```

## Output:

```
{
  "pipeline": {
    "artifactStore": {
      "location": "codepipeline-us-east-1-11EXAMPLE11",
      "type": "S3"
    },
    "name": "MyFirstPipeline",
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "actions": [
          {
            "actionTypeId": {
              "__type": "ActionTypeId",
              "category": "Source",
              "owner": "AWS",
              "provider": "S3",
              "version": "1"
            },
            "configuration": {
              "S3Bucket": "awscodepipeline-demo-bucket2",
              "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
            },
            "inputArtifacts": [],
            "name": "Source",
            "outputArtifacts": [
              {
                "name": "MyApp"
              }
            ],
            "runOrder": 1
          }
        ]
      }
    ]
  }
}
```

```
    ],
    "name": "Source"
  },
  {
    "actions": [
      {
        "actionTypeId": {
          "__type": "ActionTypeId",
          "category": "Deploy",
          "owner": "AWS",
          "provider": "CodeDeploy",
          "version": "1"
        },
        "configuration": {
          "ApplicationName": "CodePipelineDemoApplication",
          "DeploymentGroupName": "CodePipelineDemoFleet"
        },
        "inputArtifacts": [
          {
            "name": "MyApp"
          }
        ],
        "name": "CodePipelineDemoFleet",
        "outputArtifacts": [],
        "runOrder": 1
      }
    ],
    "name": "Beta"
  }
],
"version": 3
}
```

- Untuk detail API, lihat [UpdatePipeline](#) di Referensi AWS CLI Perintah.

## AWS CodeStar contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS CodeStar.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-team-member**

Contoh kode berikut menunjukkan cara menggunakan `associate-team-member`.

AWS CLI

Untuk menambahkan anggota tim ke proyek

`associate-team-member` Contoh berikut membuat intern pengguna penampil pada proyek dengan ID yang ditentukan.

```
aws codestar associate-team-member \  
  --project-id my-project \  
  --user-arn arn:aws:iam::123456789012:user/intern \  
  --project-role Viewer
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AssociateTeamMember](#) di Referensi AWS CLI Perintah.

### **create-project**

Contoh kode berikut menunjukkan cara menggunakan `create-project`.

AWS CLI

Untuk membuat proyek

`create-project` Contoh berikut menggunakan file input JSON untuk membuat CodeStar proyek.

```
aws codestar create-project \  
  --cli-input-json file://create-project.json
```

Isi dari `create-project.json`:

```
{  
  "name": "Custom Project",  
  "id": "custom-project",  
  "sourceCode": [  
    {  
      "source": {  
        "s3": {  
          "bucketName": "codestar-artifacts",  
          "bucketKey": "nodejs-function.zip"  
        }  
      },  
      "destination": {  
        "codeCommit": {  
          "name": "codestar-custom-project"  
        }  
      }  
    }  
  ],  
  "toolchain": {  
    "source": {  
      "s3": {  
        "bucketName": "codestar-artifacts",  
        "bucketKey": "toolchain.yml"  
      }  
    },  
    "roleArn": "arn:aws:iam::123456789012:role/service-role/aws-codestar-  
service-role",  
    "stackParameters": {  
      "ProjectId": "custom-project"  
    }  
  }  
}
```

Output:

```
{
  "id": "my-project",
  "arn": "arn:aws:codestar:us-east-2:123456789012:project/custom-project"
}
```

Untuk tutorial yang menyertakan kode sampel dan templat untuk proyek kustom, lihat [Membuat Proyek AWS CodeStar dengan AWS CLI](https://docs.aws.amazon.com/codestar/latest/userguide/cli-tutorial.html) < <https://docs.aws.amazon.com/codestar/latest/userguide/cli-tutorial.html> > di AWS CodeStar Panduan Pengguna.

- Untuk detail API, lihat [CreateProject](#) di Referensi AWS CLI Perintah.

## create-user-profile

Contoh kode berikut menunjukkan cara menggunakan `create-user-profile`.

### AWS CLI

Untuk membuat profil pengguna

`create-user-profile` Contoh berikut membuat profil pengguna untuk pengguna IAM dengan ARN yang ditentukan.

```
aws codestar create-user-profile \
  --user-arn arn:aws:iam::123456789012:user/intern \
  --display-name Intern \
  --email-address intern@example.com
```

Output:

```
{
  "userArn": "arn:aws:iam::123456789012:user/intern",
  "displayName": "Intern",
  "emailAddress": "intern@example.com",
  "sshPublicKey": "",
  "createdTimestamp": 1572552308.607,
  "lastModifiedTimestamp": 1572552308.607
}
```

- Untuk detail API, lihat [CreateUserProfile](#) di Referensi AWS CLI Perintah.

## delete-project

Contoh kode berikut menunjukkan cara menggunakan `delete-project`.

### AWS CLI

Untuk menghapus proyek

`delete-project` Contoh berikut menghapus proyek yang ditentukan.

```
aws codestar delete-project \  
  --project-id my-project
```

Output:

```
{  
  "projectArn": "arn:aws:codestar:us-east-2:123456789012:project/my-project"  
}
```

- Untuk detail API, lihat [DeleteProject](#) di Referensi AWS CLI Perintah.

## delete-user-profile

Contoh kode berikut menunjukkan cara menggunakan `delete-user-profile`.

### AWS CLI

Untuk menghapus profil pengguna

`delete-user-profile` Contoh berikut menghapus profil pengguna untuk pengguna dengan ARN yang ditentukan.

```
aws codestar delete-user-profile \  
  --user-arn arn:aws:iam::123456789012:user/intern
```

Output:

```
{  
  "userArn": "arn:aws:iam::123456789012:user/intern"  
}
```

- Untuk detail API, lihat [DeleteUserProfile](#) di Referensi AWS CLI Perintah.



## describe-project

Contoh kode berikut menunjukkan cara menggunakan `describe-project`.

### AWS CLI

Untuk melihat proyek

`describe-project` Contoh berikut mengambil rincian tentang proyek yang ditentukan.

```
aws codestar describe-project \  
  --id my-project
```

Output:

```
{  
  "name": "my project",  
  "id": "my-project",  
  "arn": "arn:aws:codestar:us-west-2:123456789012:project/my-project",  
  "description": "My first CodeStar project.",  
  "createdTimeStamp": 1572547510.128,  
  "status": {  
    "state": "CreateComplete"  
  }  
}
```

- Untuk detail API, lihat [DescribeProject](#) di Referensi AWS CLI Perintah.

## describe-user-profile

Contoh kode berikut menunjukkan cara menggunakan `describe-user-profile`.

### AWS CLI

Untuk melihat profil pengguna

`describe-user-profile` Contoh berikut mengambil rincian tentang profil pengguna untuk pengguna dengan ARN yang ditentukan.

```
aws codestar describe-user-profile \  
  --user-arn arn:aws:iam::123456789012:user/intern
```

**Output:**

```
{
  "userArn": "arn:aws:iam::123456789012:user/intern",
  "displayName": "Intern",
  "emailAddress": "intern@example.com",
  "sshPublicKey": "intern",
  "createdTimestamp": 1572552308.607,
  "lastModifiedTimestamp": 1572553495.47
}
```

- Untuk detail API, lihat [DescribeUserProfile](#) di Referensi AWS CLI Perintah.

**disassociate-team-member**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-team-member`.

**AWS CLI**

Untuk menghapus anggota tim

`disassociate-team-member` Contoh berikut menghapus pengguna dengan ARN yang ditentukan dari proyek. `my-project`

```
aws codestar disassociate-team-member \
  --project-id my-project \
  --user-arn arn:aws:iam::123456789012:user/intern
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DisassociateTeamMember](#) di Referensi AWS CLI Perintah.

**list-projects**

Contoh kode berikut menunjukkan cara menggunakan `list-projects`.

**AWS CLI**

Untuk melihat proyek

`list-projects` Contoh berikut mengambil daftar proyek di Wilayah saat ini.

```
aws codestar list-projects
```

Output:

```
{
  "projects": [
    {
      "projectId": "intern-projects",
      "projectArn": "arn:aws:codestar:us-west-2:123456789012:project/intern-projects"
    },
    {
      "projectId": "my-project",
      "projectArn": "arn:aws:codestar:us-west-2:123456789012:project/my-project"
    }
  ]
}
```

- Untuk detail API, lihat [ListProjects](#) di Referensi AWS CLI Perintah.

## list-resources

Contoh kode berikut menunjukkan cara menggunakan `list-resources`.

AWS CLI

Untuk melihat sumber daya

`list-resources` Contoh berikut mengambil daftar sumber daya untuk proyek yang ditentukan.

```
aws codestar list-resources \
  --id my-project
```

Output:

```
{
  "resources": [
    {
      "id": "arn:aws:execute-api:us-east-2:123456789012:r3wxmplbv8"
    },
    {
```

```
    "id": "arn:aws:codedeploy:us-
east-2:123456789012:application:awscodestar-my-project-lambda-
ServerlessDeploymentApplication-PF0LXMPL1KA0"
  },
  {
    "id": "arn:aws:s3::aws-codestar-us-east-2-123456789012-my-project-pipe"
  },
  {
    "id": "arn:aws:lambda:us-east-2:123456789012:function:awscodestar-my-
project-lambda-GetHelloWorld-16W3LVXMPLNNS"
  },
  {
    "id": "arn:aws:cloudformation:us-east-2:123456789012:stack/awscodestar-
my-project-lambda/b4904ea0-fc20-xmpl-bec6-029123b1cc42"
  },
  {
    "id": "arn:aws:cloudformation:us-east-2:123456789012:stack/awscodestar-
my-project/1b133f30-fc20-xmpl-a93a-0688c4290cb8"
  },
  {
    "id": "arn:aws:iam::123456789012:role/CodeStarWorker-my-project-
ToolChain"
  },
  {
    "id": "arn:aws:iam::123456789012:policy/CodeStar_my-
project_PermissionsBoundary"
  },
  {
    "id": "arn:aws:s3::aws-codestar-us-east-2-123456789012-my-project-app"
  },
  {
    "id": "arn:aws:codepipeline:us-east-2:123456789012:my-project-Pipeline"
  },
  {
    "id": "arn:aws:codedeploy:us-east-2:123456789012:deploymentgroup:my-
project/awscodestar-my-project-lambda-GetHelloWorldDeploymentGroup-P7YWXMPLT0QB"
  },
  {
    "id": "arn:aws:iam::123456789012:role/CodeStar-my-project-Execution"
  },
  {
    "id": "arn:aws:iam::123456789012:role/CodeStarWorker-my-project-
CodeDeploy"
  },
},
```

```
{
  "id": "arn:aws:codebuild:us-east-2:123456789012:project/my-project"
},
{
  "id": "arn:aws:iam::123456789012:role/CodeStarWorker-my-project-
CloudFormation"
},
{
  "id": "arn:aws:codecommit:us-east-2:123456789012:Go-project"
}
]
}
```

- Untuk detail API, lihat [ListResources](#) di Referensi AWS CLI Perintah.

## list-tags-for-project

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-project`.

### AWS CLI

Untuk melihat tag untuk proyek

`list-tags-for-project` Contoh berikut mengambil tag yang dilampirkan ke proyek yang ditentukan.

```
aws codestar list-tags-for-project \
  --id my-project
```

Output:

```
{
  "tags": {
    "Department": "Marketing",
    "Team": "Website"
  }
}
```

- Untuk detail API, lihat [ListTagsForProject](#) di Referensi AWS CLI Perintah.

## list-team-members

Contoh kode berikut menunjukkan cara menggunakan `list-team-members`.

### AWS CLI

Untuk melihat daftar anggota tim

`list-team-members` Contoh berikut mengambil daftar pengguna yang terkait dengan proyek yang ditentukan.

```
aws codestar list-team-members \
  --project-id my-project
```

Output:

```
{
  "teamMembers": [
    {
      "userArn": "arn:aws:iam::123456789012:user/admin",
      "projectRole": "Owner",
      "remoteAccessAllowed": false
    },
    {
      "userArn": "arn:aws:iam::123456789012:user/intern",
      "projectRole": "Contributor",
      "remoteAccessAllowed": false
    }
  ]
}
```

- Untuk detail API, lihat [ListTeamMembers](#) di Referensi AWS CLI Perintah.

## list-user-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-user-profiles`.

### AWS CLI

Untuk melihat daftar profil pengguna

`list-user-profiles` Contoh berikut mengambil daftar semua profil pengguna di Wilayah saat ini.

```
aws codestar list-user-profiles
```

Output:

```
{
  "userProfiles": [
    {
      "userArn": "arn:aws:iam::123456789012:user/admin",
      "displayName": "me",
      "emailAddress": "me@example.com",
      "sshPublicKey": ""
    },
    {
      "userArn": "arn:aws:iam::123456789012:user/intern",
      "displayName": "Intern",
      "emailAddress": "intern@example.com",
      "sshPublicKey": "intern"
    }
  ]
}
```

- Untuk detail API, lihat [ListUserProfiles](#) di Referensi AWS CLI Perintah.

## tag-project

Contoh kode berikut menunjukkan cara menggunakan `tag-project`.

### AWS CLI

Untuk melampirkan tag ke proyek

`tag-project` Contoh berikut menambahkan tag bernama `Department` dan nilai `Marketing` untuk proyek tertentu.

```
aws codestar tag-project \
  --id my-project \
  --tags Department=Marketing
```

Output:

```
{
```

```
"tags": {  
  "Department": "Marketing"  
}  
}
```

- Untuk detail API, lihat [TagProject](#) di Referensi AWS CLI Perintah.

## untag-project

Contoh kode berikut menunjukkan cara menggunakan `untag-project`.

### AWS CLI

Untuk menghapus tag dari proyek

`untag-project` Contoh berikut menghapus tag apapun dengan nama kunci Team dari dari proyek specific.

```
aws codestar untag-project \  
  --id my-project \  
  --tags Team
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagProject](#) di Referensi AWS CLI Perintah.

## update-project

Contoh kode berikut menunjukkan cara menggunakan `update-project`.

### AWS CLI

Untuk memperbarui proyek

`update-project` Contoh berikut menambahkan deskripsi untuk proyek yang ditentukan.

```
aws codestar update-project \  
  --id my-project \  
  --description "My first CodeStar project"
```

Perintah ini tidak menghasilkan output.



- Untuk detail API, lihat [UpdateProject](#) di Referensi AWS CLI Perintah.

## update-team-member

Contoh kode berikut menunjukkan cara menggunakan `update-team-member`.

### AWS CLI

Untuk memodifikasi anggota tim

`update-team-member` Contoh berikut membuat pengguna tertentu kontributor pada proyek dan memberi mereka akses jarak jauh ke sumber daya proyek.

```
aws codestar update-team-member \
  --project-id my-project \
  --user-arn arn:aws:iam::123456789012:user/intern \
  --project-role Contributor -\
  --remote-access-allowed
```

Output:

```
{
  "userArn": "arn:aws:iam::123456789012:user/intern",
  "projectRole": "Contributor",
  "remoteAccessAllowed": true
}
```

- Untuk detail API, lihat [UpdateTeamMember](#) di Referensi AWS CLI Perintah.

## update-user-profile

Contoh kode berikut menunjukkan cara menggunakan `update-user-profile`.

### AWS CLI

Untuk memodifikasi profil pengguna

`update-user-profile` Contoh berikut menambahkan kunci SSH yang ditentukan untuk pengguna tertentu.

```
aws codestar update-user-profile \
```

```
--ssh-public-key intern \  
--user-arn arn:aws:iam::123456789012:user/intern
```

#### Output:

```
{  
  "userArn": "arn:aws:iam::123456789012:user/intern",  
  "displayName": "Intern",  
  "emailAddress": "intern@example.com",  
  "sshPublicKey": "intern",  
  "createdTimestamp": 1572552308.607,  
  "lastModifiedTimestamp": 1572553495.47  
}
```

- Untuk detail API, lihat [UpdateUserProfile](#) di Referensi AWS CLI Perintah.

## AWS CodeStar Contoh pemberitahuan menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With AWS CodeStar Notifications.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-notification-rule**

Contoh kode berikut menunjukkan cara menggunakan `create-notification-rule`.

## AWS CLI

Untuk membuat aturan notifikasi

`create-notification-rule` Contoh berikut menggunakan file JSON bernama `rule.json` untuk membuat aturan notifikasi bernama `MyNotificationRule` untuk repositori bernama `MyDemoRepo` dalam akun tertentu. AWS Pemberitahuan dengan jenis `FULL` detail dikirim ke topik Amazon SNS target yang ditentukan saat cabang dan tag dibuat.

```
aws codestar-notifications create-notification-rule \  
  --cli-input-json file://rule.json
```

Isi dari `rule.json`:

```
{  
  "Name": "MyNotificationRule",  
  "EventTypeId": [  
    "codecommit-repository-branches-and-tags-created"  
  ],  
  "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",  
  "Targets": [  
    {  
      "TargetType": "SNS",  
      "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"  
    }  
  ],  
  "Status": "ENABLED",  
  "DetailType": "FULL"  
}
```

Output:

```
{  
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE"  
}
```

Untuk selengkapnya, lihat [Membuat aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [CreateNotificationRule](#) di Referensi AWS CLI Perintah.

## delete-notification-rule

Contoh kode berikut menunjukkan cara menggunakan `delete-notification-rule`.

### AWS CLI

Untuk menghapus aturan notifikasi

`delete-notification-rule` Contoh berikut menghapus aturan pemberitahuan yang ditentukan.

```
aws codestar-notifications delete-notification-rule \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE
```

Output:

```
{  
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [DeleteNotificationRule](#) di Referensi AWS CLI Perintah.

## delete-target

Contoh kode berikut menunjukkan cara menggunakan `delete-target`.

### AWS CLI

Untuk menghapus target aturan notifikasi

`delete-target` Contoh berikut menghapus target yang ditentukan dari semua aturan notifikasi yang dikonfigurasi untuk menggunakannya sebagai target, dan kemudian menghapus target.

```
aws codestar-notifications delete-target \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE
```

```
--target-address arn:aws:sns:us-east-1:123456789012:MyNotificationTopic \  
--force-unsubscribe-all
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Target Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [DeleteTarget](#) di Referensi AWS CLI Perintah.

## describe-notification-rule

Contoh kode berikut menunjukkan cara menggunakan `describe-notification-rule`.

### AWS CLI

Untuk mengambil rincian aturan notifikasi

`describe-notification-rule` Contoh berikut mengambil rincian aturan pemberitahuan yang ditentukan.

```
aws codestar-notifications describe-notification-rule \  
  --arn arn:aws:codestar-notifications:us-west-2:123456789012:notificationrule/  
dc82df7a-EXAMPLE
```

Output:

```
{  
  "LastModifiedTimestamp": 1569199844.857,  
  "EventTypes": [  
    {  
      "ServiceName": "CodeCommit",  
      "EventTypeName": "Branches and tags: Created",  
      "ResourceType": "Repository",  
      "EventTypeId": "codecommit-repository-branches-and-tags-created"  
    }  
  ],  
  "Status": "ENABLED",  
  "DetailType": "FULL",  
  "Resource": "arn:aws:codecommit:us-west-2:123456789012:MyDemoRepo",  
  "Arn": "arn:aws:codestar-notifications:us-west-w:123456789012:notificationrule/  
dc82df7a-EXAMPLE",
```

```
"Targets": [  
  {  
    "TargetStatus": "ACTIVE",  
    "TargetAddress": "arn:aws:sns:us-  
west-2:123456789012:MyNotificationTopic",  
    "TargetType": "SNS"  
  }  
],  
"Name": "MyNotificationRule",  
"CreatedTimestamp": 1569199844.857,  
"CreatedBy": "arn:aws:iam::123456789012:user/Mary_Major"  
}
```

Untuk selengkapnya, lihat [Melihat Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [DescribeNotificationRule](#) di Referensi AWS CLI Perintah.

## list-event-types

Contoh kode berikut menunjukkan cara menggunakan `list-event-types`.

### AWS CLI

Untuk mendapatkan daftar jenis acara untuk aturan notifikasi

`list-event-types` Contoh berikut mengambil daftar yang difilter dari semua jenis acara pemberitahuan yang tersedia untuk CodeDeploy aplikasi. Jika sebaliknya Anda tidak menggunakan filter, perintah mengembalikan semua jenis acara notifikasi untuk semua jenis sumber daya.

```
aws codestar-notifications list-event-types \  
--filters Name=SERVICE_NAME,Value=CodeDeploy
```

Output:

```
{  
  "EventTypes": [  
    {  
      "EventTypeId": "codedeploy-application-deployment-succeeded",  
      "ServiceName": "CodeDeploy",
```

```

    "EventTypeName": "Deployment: Succeeded",
    "ResourceType": "Application"
  },
  {
    "EventTypeId": "codedeploy-application-deployment-failed",
    "ServiceName": "CodeDeploy",
    "EventTypeName": "Deployment: Failed",
    "ResourceType": "Application"
  },
  {
    "EventTypeId": "codedeploy-application-deployment-started",
    "ServiceName": "CodeDeploy",
    "EventTypeName": "Deployment: Started",
    "ResourceType": "Application"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [ListEventTypes](#) di Referensi AWS CLI Perintah.

## list-notification-rules

Contoh kode berikut menunjukkan cara menggunakan `list-notification-rules`.

### AWS CLI

Untuk mengambil daftar aturan notifikasi

`list-notification-rules` Contoh berikut mengambil daftar semua aturan notifikasi di AWS Wilayah tertentu.

```
aws codestar-notifications list-notification-rules --region us-east-1
```

Output:

```

{
  "NotificationRules": [
    {
      "Id": "dc82df7a-EXAMPLE",

```

```

    "Arn": "arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/dc82df7a-EXAMPLE"
  },
  {
    "Id": "8d1f0983-EXAMPLE",
    "Arn": "arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/8d1f0983-EXAMPLE"
  }
]
}

```

Untuk selengkapnya, lihat [Melihat Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [ListNotificationRules](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mendapatkan daftar tag yang dilampirkan pada aturan notifikasi

`list-tags-for-resource` Contoh berikut mengambil daftar semua tag yang dilampirkan pada aturan pemberitahuan yang ditentukan. Dalam contoh ini, aturan notifikasi saat ini tidak memiliki tag yang terkait dengannya.

```

aws codestar-notifications list-tags-for-resource \
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
fe1efd35-EXAMPLE

```

Output:

```

{
  "Tags": {}
}

```

Untuk informasi selengkapnya, lihat [Membuat Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.



## list-targets

Contoh kode berikut menunjukkan cara menggunakan `list-targets`.

### AWS CLI

Untuk mengambil daftar target aturan notifikasi

`list-targets` Contoh berikut mengambil daftar semua target aturan notifikasi di AWS Wilayah tertentu.

```
aws codestar-notifications list-targets \  
  --region us-east-1
```

Output:

```
{  
  "Targets": [  
    {  
      "TargetAddress": "arn:aws:sns:us-  
east-1:123456789012:MySNSTopicForNotificationRules",  
      "TargetType": "SNS",  
      "TargetStatus": "ACTIVE"  
    },  
    {  
      "TargetAddress": "arn:aws:sns:us-  
east-1:123456789012:MySNSTopicForNotificationsAboutMyDemoRepo",  
      "TargetType": "SNS",  
      "TargetStatus": "ACTIVE"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat Target Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [ListTargets](#) di Referensi AWS CLI Perintah.

## subscribe

Contoh kode berikut menunjukkan cara menggunakan `subscribe`.

## AWS CLI

Untuk menambahkan target ke aturan notifikasi

`subscribe` Contoh berikut menambahkan topik Amazon SNS sebagai target untuk aturan notifikasi yang ditentukan.

```
aws codestar-notifications subscribe \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE \  
  --target TargetType=SNS,TargetAddress=arn:aws:sns:us-  
east-1:123456789012:MyNotificationTopic
```

Output:

```
{  
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE"  
}
```

Untuk selengkapnya, lihat [Menambahkan atau Menghapus Topik Amazon SNS sebagai Target untuk Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [Berlangganan](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menambahkan tag ke aturan notifikasi

`tag-resource` Contoh berikut menambahkan tag dengan nama kunci `Team` dan nilai `Li_Juan` untuk aturan pemberitahuan yang ditentukan.

```
aws codestar-notifications tag-resource \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
fe1efd35-EXAMPLE \  
  --tags Team=Li_Juan
```

**Output:**

```
{
  "Tags": {
    "Team": "Li_Juan"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

**unsubscribe**

Contoh kode berikut menunjukkan cara menggunakan `unsubscribe`.

**AWS CLI**

Untuk menghapus target dari aturan notifikasi

`unsubscribe` Contoh berikut menghapus topik Amazon SNS sebagai target dari aturan notifikasi yang ditentukan.

```
aws codestar-notifications unsubscribe \
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE \
  --target TargetType=SNS,TargetAddress=arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic
```

**Output:**

```
{
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
  "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"
}
```

Untuk selengkapnya, lihat [Menambahkan atau Menghapus Topik Amazon SNS sebagai Target untuk Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [Berhenti berlangganan](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari aturan notifikasi

`untag-resource` Contoh berikut menghapus tag dengan nama kunci Team dari aturan pemberitahuan yang ditentukan.

```
aws codestar-notifications untag-resource \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
fe1efd35-EXAMPLE \  
  --tag-keys Team
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengedit Aturan Pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-notification-rule

Contoh kode berikut menunjukkan cara menggunakan `update-notification-rule`.

### AWS CLI

Untuk memperbarui aturan notifikasi

`update-notification-rule` Contoh berikut memperbarui aturan notifikasi bernama MyNotificationRule di AWS akun 123456789012 menggunakan file JSON bernama `update.json`.

```
aws codestar-notifications update-notification-rule \  
  --cli-input-json file://update.json
```

Isi dari `update.json`:

```
{  
  "Name": "MyUpdatedNotificationRule",
```

```
"EventTypeId": [
  "codecommit-repository-branches-and-tags-created"
],
"Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
"Targets": [
  {
    "TargetType": "SNS",
    "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic"
  }
],
>Status": "ENABLED",
"DetailType": "FULL"
}
```

Output:

```
{
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Mengedit aturan pemberitahuan](#) di Panduan Pengguna Konsol Alat AWS Pengembang.

- Untuk detail API, lihat [UpdateNotificationRule](#) di Referensi AWS CLI Perintah.

## CodeConnections contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with CodeConnections.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **create-connection**

Contoh kode berikut menunjukkan cara menggunakan `create-connection`.

## AWS CLI

Untuk membuat koneksi

`create-connection` Contoh berikut menunjukkan cara membuat koneksi ke repositori pihak ketiga. Contoh ini membuat koneksi di mana penyedia pihak ketiga adalah Bitbucket.

Koneksi yang dibuat melalui AWS CLI atau AWS CloudFormation dalam status Tertunda secara default. Setelah Anda membuat koneksi dengan CLI atau AWS CloudFormation, gunakan konsol untuk mengedit koneksi agar statusnya Tersedia.

```
aws codestar-connections create-connection \  
  --provider-type Bitbucket \  
  --connection-name MyConnection
```

Output:

```
{  
  "ConnectionArn": "arn:aws:codestar-connections:us-  
east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f"  
}
```

Untuk informasi selengkapnya, lihat [Membuat sambungan](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [CreateConnection](#) di Referensi AWS CLI Perintah.

### **create-host**

Contoh kode berikut menunjukkan cara menggunakan `create-host`.

## AWS CLI

Untuk membuat host

`create-host` Contoh berikut menunjukkan cara membuat host untuk mewakili titik akhir infrastruktur tempat penyedia pihak ketiga Anda diinstal. Contoh ini membuat host di mana penyedia yang diinstal pihak ketiga adalah GitHub Enterprise Server.

Host yang dibuat melalui AWS CLI berada dalam status Tertunda secara default. Setelah Anda membuat host dengan CLI, gunakan konsol atau CLI untuk mengatur host agar statusnya Tersedia.

```
aws codestar-connections create-host \  
  --name MyHost \  
  --provider-type GitHubEnterpriseServer \  
  --provider-endpoint "https://my-instance.dev"
```

Output:

```
{  
  "HostArn": "arn:aws:codestar-connections:us-east-1:123456789012:host/My-  
Host-28aef605"  
}
```

Untuk informasi selengkapnya, lihat [Membuat host \(CLI\)](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [CreateHost](#) di Referensi AWS CLI Perintah.

## **delete-connection**

Contoh kode berikut menunjukkan cara menggunakan `delete-connection`.

## AWS CLI

Untuk menghapus koneksi

`delete-connection` Contoh berikut menunjukkan cara menghapus koneksi.

```
aws codestar-connections delete-connection \  
  --connection-id MyConnectionId
```

```
--connection-arn arn:aws:codestar-connections:us-west-2:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus sambungan \(CLI\)](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [DeleteConnection](#) di Referensi AWS CLI Perintah.

## delete-host

Contoh kode berikut menunjukkan cara menggunakan `delete-host`.

### AWS CLI

Untuk menghapus host

`delete-host` Contoh berikut menunjukkan cara menghapus host. Sebelum Anda dapat menghapus host, Anda harus menghapus semua koneksi yang terkait dengan host.

```
aws codestar-connections delete-host \  
  --host-arn "arn:aws:codestar-connections:us-east-1 :123456789012:host/My-  
Host-28aef605"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus host \(CLI\)](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [DeleteHost](#) di Referensi AWS CLI Perintah.

## get-connection

Contoh kode berikut menunjukkan cara menggunakan `get-connection`.

### AWS CLI

Untuk mendapatkan informasi tentang koneksi

`get-connection` Contoh berikut menunjukkan rincian tentang koneksi.



```
aws codestar-connections get-connection \  
  --connection-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Output:

```
{  
  "Connection": {  
    "ConnectionName": "MyConnection",  
    "ConnectionArn": "arn:aws:codestar-connections:us-  
east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",  
    "ProviderType": "Bitbucket",  
    "OwnerAccountId": "123456789012",  
    "ConnectionStatus": "AVAILABLE"  
  }  
}
```

Untuk informasi selengkapnya, [lihat Melihat detail koneksi](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [GetConnection](#) di Referensi AWS CLI Perintah.

## get-host

Contoh kode berikut menunjukkan cara menggunakan `get-host`.

### AWS CLI

Untuk mendapatkan informasi tentang host

`get-host` Contoh berikut menunjukkan rincian tentang host:

```
aws codestar-connections get-host \  
  --host-arn arn:aws:codestar-connections:us-east-1:123456789012:host/  
MyHost-28aef605
```

Output:

```
{  
  "Name": "MyHost",
```

```
"Status": "AVAILABLE",
"ProviderType": "GitHubEnterpriseServer",
"ProviderEndpoint": "https://test-instance-1.dev/"
}
```

Untuk informasi selengkapnya, [lihat Melihat detail host \(CLI\)](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [GetHost](#) di Referensi AWS CLI Perintah.

## list-connections

Contoh kode berikut menunjukkan cara menggunakan `list-connections`.

### AWS CLI

Untuk membuat daftar koneksi

`list-connections` Contoh berikut mengambil daftar semua koneksi di akun Anda untuk jenis penyedia Bitbucket. :

```
aws codestar-connections list-connections \
--provider-type Bitbucket \
--max-results 5 \
--next-token: next-token
```

Output:

```
{
  "Connections": [
    {
      "ConnectionName": "my-connection",
      "ProviderType": "Bitbucket",
      "Status": "PENDING",
      "ARN": "arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
      "OwnerAccountId": "123456789012"
    },
    {
      "ConnectionName": "my-other-connection",
      "ProviderType": "Bitbucket",
      "Status": "AVAILABLE",
```

```
    "ARN": "arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
    "OwnerId": "123456789012"
  },
],
"NextToken": "next-token"
}
```

Untuk informasi selengkapnya, lihat [Daftar koneksi \(CLI\)](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [ListConnections](#) di Referensi AWS CLI Perintah.

## list-hosts

Contoh kode berikut menunjukkan cara menggunakan `list-hosts`.

### AWS CLI

Untuk daftar host

`list-hosts` Contoh berikut mengambil daftar semua host di akun Anda.

```
aws codestar-connections list-hosts
```

Output:

```
{
  "Hosts": [
    {
      "Name": "My-Host",
      "HostArn": "arn:aws:codestar-connections:us-east-1:123456789012:host/My-
Host-28aef605",
      "ProviderType": "GitHubEnterpriseServer",
      "ProviderEndpoint": "https://my-instance.test.dev",
      "Status": "AVAILABLE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Daftar host \(CLI\)](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [ListHosts](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag

`list-tags-for-resource` Contoh berikut mengambil daftar semua tag yang dilampirkan ke sumber daya koneksi tertentu.

```
aws codestar-connections list-tags-for-resource \
  --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Output:

```
{
  "Tags": [
    {
      "Key": "Project",
      "Value": "ProjectA"
    },
    {
      "Key": "ReadOnly",
      "Value": "true"
    }
  ]
}
```

Untuk informasi selengkapnya, [lihat Melihat tag untuk sumber daya koneksi](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut mengaitkan satu set tag yang disediakan dengan koneksi. Gunakan perintah ini untuk menambah atau mengedit tag.

```
aws codestar-connections tag-resource \  
  --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f \  
  --tags Key=Project,Value=ProjectA Key=IscontainerBased,Value=true
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan tag ke sumber daya koneksi](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus AWS tag dari sumber daya koneksi

Berikut ini `untag-resource` menghapus tag dari sumber daya yang ditentukan.

```
aws codestar-connections untag-resource \  
  --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f \  
  --tag-keys Project ReadOnly
```

Output:

```
{  
  "Tags": []  
}
```

Untuk informasi selengkapnya, lihat [Menghapus tag dari sumber daya koneksi](#) di Panduan Pengguna konsol Alat Pengembang.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Contoh Identitas Amazon Cognito menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Identitas Cognito AWS Command Line Interface With Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-identity-pool**

Contoh kode berikut menunjukkan cara menggunakan `create-identity-pool`.

AWS CLI

Untuk membuat kumpulan identitas dengan penyedia kumpulan identitas Cognito

Contoh ini membuat kumpulan identitas bernama `MyIdentityPool`. Ini memiliki penyedia kumpulan identitas Cognito. Identitas yang tidak diautentikasi tidak diperbolehkan.

Perintah:

```
aws cognito-identity create-identity-pool --identity-pool-name
MyIdentityPool --no-allow-unauthenticated-identities --cognito-
identity-providers ProviderName="cognito-idp.us-west-2.amazonaws.com/us-
west-2_aaaaaaaa",ClientId="3n4b5urk1ft4f13mg5e62d9ado",ServerSideTokenCheck=false
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "IdentityPoolName": "MyIdentityPool",
  "AllowUnauthenticatedIdentities": false,
  "CognitoIdentityProviders": [
    {
      "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_1111111111",
      "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",
      "ServerSideTokenCheck": false
    }
  ]
}
```

- Untuk detail API, lihat [CreatelIdentityPool](#) di Referensi AWS CLI Perintah.

## **delete-identities**

Contoh kode berikut menunjukkan cara menggunakan `delete-identities`.

### AWS CLI

Untuk menghapus kumpulan identitas

Contoh ini menghapus kumpulan identitas.

Perintah:

```
aws cognito-identity delete-identity-pool --identity-ids-to-delete "us-west-2:11111111-1111-1111-1111-111111111111"
```

Output:

```
{
  "UnprocessedIdentityIds": []
}
```

- Untuk detail API, lihat [DeletelIdentities](#) di Referensi AWS CLI Perintah.

## **delete-identity-pool**

Contoh kode berikut menunjukkan cara menggunakan `delete-identity-pool`.

## AWS CLI

Untuk menghapus kumpulan identitas

`delete-identity-pool` Contoh berikut menghapus kumpulan identitas yang ditentukan.

Perintah:

```
aws cognito-identity delete-identity-pool \  
  --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteIdentityPool](#) di Referensi AWS CLI Perintah.

## **describe-identity-pool**

Contoh kode berikut menunjukkan cara menggunakan `describe-identity-pool`.

## AWS CLI

Untuk menggambarkan kumpulan identitas

Contoh ini menjelaskan kumpulan identitas.

Perintah:

```
aws cognito-identity describe-identity-pool --identity-pool-id "us-  
west-2:11111111-1111-1111-1111-111111111111"
```

Output:

```
{  
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",  
  "IdentityPoolName": "MyIdentityPool",  
  "AllowUnauthenticatedIdentities": false,  
  "CognitoIdentityProviders": [  
    {  
      "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_1111111111",  
      "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",  
      "ServerSideTokenCheck": false  
    }  
  ]  
}
```



```
}
```

- Untuk detail API, lihat [DescribeIdentityPool](#) di Referensi AWS CLI Perintah.

## get-identity-pool-roles

Contoh kode berikut menunjukkan cara menggunakan `get-identity-pool-roles`.

### AWS CLI

Untuk mendapatkan peran kumpulan identitas

Contoh ini mendapatkan peran kumpulan identitas.

Perintah:

```
aws cognito-identity get-identity-pool-roles --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111"
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "Roles": {
    "authenticated": "arn:aws:iam::111111111111:role/Cognito_MyIdentityPoolAuth_Role",
    "unauthenticated": "arn:aws:iam::111111111111:role/Cognito_MyIdentityPoolUnauth_Role"
  }
}
```

- Untuk detail API, lihat [GetIdentityPoolRoles](#) di Referensi AWS CLI Perintah.

## list-identity-pools

Contoh kode berikut menunjukkan cara menggunakan `list-identity-pools`.

### AWS CLI

Untuk membuat daftar kumpulan identitas

Contoh ini mencantumkan kumpulan identitas. Ada maksimal 20 identitas yang terdaftar.

**Perintah:**

```
aws cognito-identity list-identity-pools --max-results 20
```

**Output:**

```
{
  "IdentityPools": [
    {
      "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
      "IdentityPoolName": "MyIdentityPool"
    },
    {
      "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
      "IdentityPoolName": "AnotherIdentityPool"
    },
    {
      "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
      "IdentityPoolName": "IdentityPoolRegionA"
    }
  ]
}
```

- Untuk detail API, lihat [ListIdentityPools](#) di Referensi AWS CLI Perintah.

**set-identity-pool-roles**

Contoh kode berikut menunjukkan cara menggunakan `set-identity-pool-roles`.

**AWS CLI**

Untuk mengatur peran kumpulan identitas

`set-identity-pool-roles` Contoh berikut menetapkan peran kumpulan identitas.

```
aws cognito-identity set-identity-pool-roles \
  --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111" \
  --roles authenticated="arn:aws:iam::111111111111:role/
Cognito_MyIdentityPoolAuth_Role"
```

- Untuk detail API, lihat [SetIdentityPoolRoles](#) di Referensi AWS CLI Perintah.

## update-identity-pool

Contoh kode berikut menunjukkan cara menggunakan `update-identity-pool`.

### AWS CLI

Untuk memperbarui kumpulan identitas

Contoh ini memperbarui kumpulan identitas. Ini menetapkan nama untuk `MyIdentityPool`. Ia menambahkan Cognito sebagai penyedia identitas. Ini melarang identitas yang tidak diautentikasi.

Perintah:

```
aws cognito-identity update-identity-pool --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111" --identity-pool-name "MyIdentityPool" --no-allow-unauthenticated-identities --cognito-identity-providers ProviderName="cognito-idp.us-west-2.amazonaws.com/us-west-2_11111111",ClientId="3n4b5urk1ft4f13mg5e62d9ado",ServerSideTokenCheck=false
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "IdentityPoolName": "MyIdentityPool",
  "AllowUnauthenticatedIdentities": false,
  "CognitoIdentityProviders": [
    {
      "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_11111111",
      "ClientId": "3n4b5urk1ft4f13mg5e62d9ado",
      "ServerSideTokenCheck": false
    }
  ]
}
```

- Untuk detail API, lihat [UpdateIdentityPool](#) di Referensi AWS CLI Perintah.

## Contoh Penyedia Identitas Amazon Cognito menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Penyedia Identitas Amazon Cognito AWS Command Line Interface dengan.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-custom-attributes**

Contoh kode berikut menunjukkan cara menggunakan `add-custom-attributes`.

AWS CLI

Untuk menambahkan atribut kustom

Contoh ini menambahkan atribut kustom `CustomAttr 1` ke kumpulan pengguna. Ini adalah tipe `String`, dan membutuhkan minimal 1 karakter dan maksimal 15. Itu tidak diperlukan.

Perintah:

```
aws cognito-idp add-custom-attributes --user-pool-id us-west-2_aaaaaaaaa --custom-attributes Name="CustomAttr1",AttributeDataType="String",DeveloperOnlyAttribute=false,Required=false,S
```

- Untuk detail API, lihat [AddCustomAttributes](#) di Referensi AWS CLI Perintah.

### **admim-disable-user**

Contoh kode berikut menunjukkan cara menggunakan `admim-disable-user`.

AWS CLI

Untuk menonaktifkan pengguna

Contoh ini menonaktifkan pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp admin-disable-user --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com
```

- Untuk detail API, lihat [AdminDisableUser](#) di Referensi AWS CLI Perintah.

## **admim-enable-user**

Contoh kode berikut menunjukkan cara menggunakan `admim-enable-user`.

AWS CLI

Untuk mengaktifkan pengguna

Contoh ini memungkinkan nama pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp admin-enable-user --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com
```

- Untuk detail API, lihat [AdminEnableUser](#) di Referensi AWS CLI Perintah.

## **admin-add-user-to-group**

Contoh kode berikut menunjukkan cara menggunakan `admin-add-user-to-group`.

AWS CLI

Untuk menambahkan pengguna ke grup

Contoh ini menambahkan pengguna Jane ke grup MyGroup.

Perintah:

```
aws cognito-idp admin-add-user-to-group --user-pool-id us-west-2_aaaaaaaaa --username Jane --group-name MyGroup
```

- Untuk detail API, lihat [AdminAddUserToGroup](#) di Referensi AWS CLI Perintah.

## admin-confirm-sign-up

Contoh kode berikut menunjukkan cara menggunakan `admin-confirm-sign-up`.

### AWS CLI

Untuk mengonfirmasi pendaftaran pengguna

Contoh ini mengkonfirmasi pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp admin-confirm-sign-up --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com
```

- Untuk detail API, lihat [AdminConfirmSignUp](#) di Referensi AWS CLI Perintah.

## admin-create-user

Contoh kode berikut menunjukkan cara menggunakan `admin-create-user`.

### AWS CLI

Untuk membuat pengguna

`admin-create-user` Contoh berikut membuat pengguna dengan pengaturan alamat email dan nomor telepon yang ditentukan.

```
aws cognito-idp admin-create-user \
  --user-pool-id us-west-2_aaaaaaaaa \
  --username diego \
  --user-attributes Name=email,Value=diego@example.com
Name=phone_number,Value="+15555551212" \
  --message-action SUPPRESS
```

Output:

```
{
  "User": {
    "Username": "diego",
    "Attributes": [
```

```
    {
      "Name": "sub",
      "Value": "7325c1de-b05b-4f84-b321-9adc6e61f4a2"
    },
    {
      "Name": "phone_number",
      "Value": "+15555551212"
    },
    {
      "Name": "email",
      "Value": "diego@example.com"
    }
  ],
  "UserCreateDate": 1548099495.428,
  "UserLastModifiedDate": 1548099495.428,
  "Enabled": true,
  "UserStatus": "FORCE_CHANGE_PASSWORD"
}
```

- Untuk detail API, lihat [AdminCreateUser](#) di Referensi AWS CLI Perintah.

## admin-delete-user-attributes

Contoh kode berikut menunjukkan cara menggunakan `admin-delete-user-attributes`.

### AWS CLI

Untuk menghapus atribut pengguna

Contoh ini menghapus atribut kustom `CustomAttr 1` untuk pengguna `diego@example.com`.

Perintah:

```
aws cognito-idp admin-delete-user-attributes --user-pool-id us-west-2_aaaaaaaaaa --
username diego@example.com --user-attribute-names "custom:CustomAttr1"
```

- Untuk detail API, lihat [AdminDeleteUserAttributes](#) di Referensi AWS CLI Perintah.

## admin-delete-user

Contoh kode berikut menunjukkan cara menggunakan `admin-delete-user`.

## AWS CLI

Untuk menghapus pengguna

Contoh ini menghapus pengguna.

Perintah:

```
aws cognito-idp admin-delete-user --user-pool-id us-west-2_aaaaaaaaa --username
diego@example.com
```

- Untuk detail API, lihat [AdminDeleteUser](#) di Referensi AWS CLI Perintah.

## **admin-forget-device**

Contoh kode berikut menunjukkan cara menggunakan `admin-forget-device`.

## AWS CLI

Untuk melupakan perangkat

Contoh ini lupa perangkat untuk nama pengguna `jane@example.com`

Perintah:

```
aws cognito-idp admin-forget-device --user-pool-id us-west-2_aaaaaaaaa --username
jane@example.com --device-key us-west-2_abcd_1234-5678
```

- Untuk detail API, lihat [AdminForgetDevice](#) di Referensi AWS CLI Perintah.

## **admin-get-device**

Contoh kode berikut menunjukkan cara menggunakan `admin-get-device`.

## AWS CLI

Untuk mendapatkan perangkat

Contoh ini mendapatkan perangkat untuk nama pengguna `jane@example.com`

Perintah:



```
aws cognito-idp admin-get-device --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com --device-key us-west-2_abcd_1234-5678
```

- Untuk detail API, lihat [AdminGetDevice](#) di Referensi AWS CLI Perintah.

## admin-get-user

Contoh kode berikut menunjukkan cara menggunakan `admin-get-user`.

### AWS CLI

Untuk mendapatkan pengguna

Contoh ini mendapatkan informasi tentang nama pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp admin-get-user --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com
```

Output:

```
{
  "Username": "4320de44-2322-4620-999b-5e2e1c8df013",
  "Enabled": true,
  "UserStatus": "FORCE_CHANGE_PASSWORD",
  "UserCreateDate": 1548108509.537,
  "UserAttributes": [
    {
      "Name": "sub",
      "Value": "4320de44-2322-4620-999b-5e2e1c8df013"
    },
    {
      "Name": "email_verified",
      "Value": "true"
    },
    {
      "Name": "phone_number_verified",
      "Value": "true"
    },
    {
```

```

        "Name": "phone_number",
        "Value": "+01115551212"
    },
    {
        "Name": "email",
        "Value": "jane@example.com"
    }
],
"UserLastModifiedDate": 1548108509.537
}

```

- Untuk detail API, lihat [AdminGetUser](#) di Referensi AWS CLI Perintah.

## admin-initiate-auth

Contoh kode berikut menunjukkan cara menggunakan `admin-initiate-auth`.

### AWS CLI

Untuk memulai otorisasi

Contoh ini memulai otorisasi menggunakan alur `ADMIN_NO_SRP_AUTH` untuk nama pengguna `jane@example.com`

Klien harus mengaktifkan API login untuk otentikasi berbasis server (`ADMIN_NO_SRP_AUTH`).

Gunakan informasi sesi dalam nilai kembali untuk memanggil `admin-respond-to-auth -challenge`.

Perintah:

```

aws cognito-idp admin-initiate-auth --user-pool-id us-west-2_aaaaaaaaa --client-id 3n4b5urk1ft4f13mg5e62d9ado --auth-flow ADMIN_NO_SRP_AUTH --auth-parameters USERNAME=jane@example.com,PASSWORD=password

```

Output:

```

{
  "ChallengeName": "NEW_PASSWORD_REQUIRED",
  "Session": "SESSION",
  "ChallengeParameters": {
    "USER_ID_FOR_SRP": "84514837-dcbc-4af1-abff-f3c109334894",
    "requiredAttributes": "[]"
  }
}

```

```
"userAttributes": "{\\"email_verified\\":\\"true\\",\\"phone_number_verified\\":\\"true\\",\\"phone_number\\":\\"+01xxx5550100\\",\\"email\\":\\"jane@example.com\\"}"
}
```

- Untuk detail API, lihat [AdminInitiateAuth](#) di Referensi AWS CLI Perintah.

## admin-list-devices

Contoh kode berikut menunjukkan cara menggunakan `admin-list-devices`.

### AWS CLI

Untuk membuat daftar perangkat untuk pengguna

Contoh ini mencantumkan perangkat untuk nama pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp admin-list-devices --user-pool-id us-west-2_aaaaaaaaaa --username jane@example.com
```

- Untuk detail API, lihat [AdminListDevices](#) di Referensi AWS CLI Perintah.

## admin-list-groups-for-user

Contoh kode berikut menunjukkan cara menggunakan `admin-list-groups-for-user`.

### AWS CLI

Untuk daftar grup untuk pengguna

Contoh ini mencantumkan grup untuk nama pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp admin-list-groups-for-user --user-pool-id us-west-2_aaaaaaaaaa --username diego@example.com
```

Output:

```
{
  "Groups": [
    {
      "Description": "Sample group",
      "Precedence": 1,
      "LastModifiedDate": 1548097827.125,
      "RoleArn": "arn:aws:iam::111111111111:role/SampleRole",
      "GroupName": "SampleGroup",
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "CreationDate": 1548097827.125
    }
  ]
}
```

- Untuk detail API, lihat [AdminListGroupForUser](#) di Referensi AWS CLI Perintah.

### **admin-list-user-auth-events**

Contoh kode berikut menunjukkan cara menggunakan `admin-list-user-auth-events`.

#### AWS CLI

Untuk mencantumkan acara otorisasi untuk pengguna

Contoh ini mencantumkan acara otorisasi untuk nama pengguna `diego@example.com`.

Perintah:

```
aws cognito-idp admin-list-user-auth-events --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com
```

- Untuk detail API, lihat [AdminListUserAuthEvents](#) di Referensi AWS CLI Perintah.

### **admin-remove-user-from-group**

Contoh kode berikut menunjukkan cara menggunakan `admin-remove-user-from-group`.

#### AWS CLI

Untuk menghapus pengguna dari grup

Contoh ini menghapus `jane@example.com` dari `SampleGroup`.

Perintah:

```
aws cognito-idp admin-remove-user-from-group --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com --group-name SampleGroup
```

- Untuk detail API, lihat [AdminRemoveUserFromGroup](#) di Referensi AWS CLI Perintah.

## **admin-reset-user-password**

Contoh kode berikut menunjukkan cara menggunakan `admin-reset-user-password`.

AWS CLI

Untuk mengatur ulang kata sandi pengguna

Contoh ini mengatur ulang kata sandi untuk `diego@example.com`.

Perintah:

```
aws cognito-idp admin-reset-user-password --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com
```

- Untuk detail API, lihat [AdminResetUserPassword](#) di Referensi AWS CLI Perintah.

## **admin-set-user-mfa-preference**

Contoh kode berikut menunjukkan cara menggunakan `admin-set-user-mfa-preference`.

AWS CLI

Untuk mengatur preferensi MFA pengguna

Contoh ini menetapkan preferensi SMS MFA untuk nama pengguna `diego@example.com`.

Perintah:

```
aws cognito-idp admin-set-user-mfa-preference --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com --sms-mfa-settings Enabled=false,PreferredMfa=false
```

- Untuk detail API, lihat [AdminSetUserMfaPreference](#) di Referensi AWS CLI Perintah.

## admin-set-user-settings

Contoh kode berikut menunjukkan cara menggunakan `admin-set-user-settings`.

### AWS CLI

Untuk mengatur pengaturan pengguna

Contoh ini menetapkan preferensi pengiriman MFA untuk nama pengguna `diego@example.com` ke EMAIL.

Perintah:

```
aws cognito-idp admin-set-user-settings --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com --mfa-options DeliveryMedium=EMAIL
```

- Untuk detail API, lihat [AdminSetUserSettings](#) di Referensi AWS CLI Perintah.

## admin-update-auth-event-feedback

Contoh kode berikut menunjukkan cara menggunakan `admin-update-auth-event-feedback`.

### AWS CLI

Untuk memberikan umpan balik untuk acara otorisasi

Contoh ini menetapkan nilai umpan balik untuk peristiwa otorisasi yang diidentifikasi oleh `event-id` ke Valid.

Perintah:

```
aws cognito-idp admin-update-auth-event-feedback --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com --event-id c2c2cf89-c0d3-482d-aba6-99d78a5b0bfe --feedback-value Valid
```

- Untuk detail API, lihat [AdminUpdateAuthEventFeedback](#) di Referensi AWS CLI Perintah.

## admin-update-device-status

Contoh kode berikut menunjukkan cara menggunakan `admin-update-device-status`.

## AWS CLI

Untuk memperbarui status perangkat

Contoh ini menetapkan status perangkat yang diingat untuk perangkat yang diidentifikasi oleh device-key ke not\_remembered.

Perintah:

```
aws cognito-idp admin-update-device-status --user-pool-id us-west-2_aaaaaaaaaa
--username diego@example.com --device-key xxxx --device-remembered-status
not_remembered
```

- Untuk detail API, lihat [AdminUpdateDeviceStatus](#) di Referensi AWS CLI Perintah.

## admin-update-user-attributes

Contoh kode berikut menunjukkan cara menggunakan admin-update-user-attributes.

## AWS CLI

Untuk memperbarui atribut pengguna

Contoh ini memperbarui atribut pengguna kustom CustomAttr 1 untuk pengguna diego@example.com.

Perintah:

```
aws cognito-idp admin-update-user-attributes --user-pool-id us-
west-2_aaaaaaaaaa --username diego@example.com --user-attributes
Name="custom:CustomAttr1",Value="Purple"
```

- Untuk detail API, lihat [AdminUpdateUserAttributes](#) di Referensi AWS CLI Perintah.

## change-password

Contoh kode berikut menunjukkan cara menggunakan change-password.

## AWS CLI

Untuk mengubah kata sandi

Contoh ini mengubah kata sandi.

Perintah:

```
aws cognito-idp change-password --previous-password OldPassword --proposed-password  
NewPassword --access-token ACCESS_TOKEN
```

- Untuk detail API, lihat [ChangePassword](#) di Referensi AWS CLI Perintah.

## **confirm-forgot-password**

Contoh kode berikut menunjukkan cara menggunakan `confirm-forgot-password`.

AWS CLI

Untuk mengonfirmasi kata sandi yang terlupakan

Contoh ini mengonfirmasi kata sandi yang terlupakan untuk nama pengguna `diego@example.com`.

Perintah:

```
aws cognito-idp confirm-forgot-password --client-id 3n4b5urk1ft4f13mg5e62d9ado --  
username=diego@example.com --password PASSWORD --confirmation-code CONF_CODE
```

- Untuk detail API, lihat [ConfirmForgotPassword](#) di Referensi AWS CLI Perintah.

## **confirm-sign-up**

Contoh kode berikut menunjukkan cara menggunakan `confirm-sign-up`.

AWS CLI

Untuk mengonfirmasi pendaftaran

Contoh ini mengonfirmasi pendaftaran untuk nama pengguna `diego@example.com`.

Perintah:

```
aws cognito-idp confirm-sign-up --client-id 3n4b5urk1ft4f13mg5e62d9ado --  
username=diego@example.com --confirmation-code CONF_CODE
```



- Untuk detail API, lihat [ConfirmSignUp](#) di Referensi AWS CLI Perintah.

## create-group

Contoh kode berikut menunjukkan cara menggunakan `create-group`.

### AWS CLI

Untuk membuat grup

Contoh ini membuat grup dengan deskripsi.

Perintah:

```
aws cognito-idp create-group --user-pool-id us-west-2_aaaaaaaa --group-name
MyNewGroup --description "New group."
```

Output:

```
{
  "Group": {
    "GroupName": "MyNewGroup",
    "UserPoolId": "us-west-2_aaaaaaaa",
    "Description": "New group.",
    "LastModifiedDate": 1548270073.795,
    "CreationDate": 1548270073.795
  }
}
```

Untuk membuat grup dengan peran dan prioritas

Contoh ini membuat grup dengan deskripsi. Ini juga mencakup peran dan prioritas.

Perintah:

```
aws cognito-idp create-group --user-pool-id us-west-2_aaaaaaaa --group-
name MyNewGroupWithRole --description "New group with a role." --role-arn
arn:aws:iam::111111111111:role/MyNewGroupRole --precedence 2
```

Output:

```
{
```

```
"Group": {
  "GroupName": "MyNewGroupWithRole",
  "UserPoolId": "us-west-2_aaaaaaaaaa",
  "Description": "New group with a role.",
  "RoleArn": "arn:aws:iam::111111111111:role/MyNewGroupRole",
  "Precedence": 2,
  "LastModifiedDate": 1548270211.761,
  "CreationDate": 1548270211.761
}
```

- Untuk detail API, lihat [CreateGroup](#) di Referensi AWS CLI Perintah.

## create-user-import-job

Contoh kode berikut menunjukkan cara menggunakan `create-user-import-job`.

### AWS CLI

Untuk membuat pekerjaan impor pengguna

Contoh ini membuat pekerjaan impor pengguna bernama `MyImportJob`.

Untuk informasi selengkapnya tentang mengimpor pengguna, lihat [Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV](#).

Perintah:

```
aws cognito-idp create-user-import-job --user-pool-id us-west-2_aaaaaaaaaa --
job-name MyImportJob --cloud-watch-logs-role-arn arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole
```

Output:

```
{
  "UserImportJob": {
    "JobName": "MyImportJob",
    "JobId": "import-qQ0DCt2fRh",
    "UserPoolId": "us-west-2_aaaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548271795.471,
    "Status": "Created",
```

```

    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0
  }
}

```

Unggah file.csv dengan curl menggunakan URL yang telah ditandatangani sebelumnya:

Perintah:

```

curl -v -T "PATH_TO_CSV_FILE" -H "x-amz-server-side-encryption:aws:kms"
"PRE_SIGNED_URL"

```

- Untuk detail API, lihat [CreateUserImportJob](#) di Referensi AWS CLI Perintah.

## create-user-pool-client

Contoh kode berikut menunjukkan cara menggunakan `create-user-pool-client`.

AWS CLI

Untuk membuat klien kumpulan pengguna

Contoh ini membuat klien kumpulan pengguna baru dengan dua alur otorisasi eksplisit: `USER_PASSWORD_AUTH` dan `ADMIN_NO_SRP_AUTH`.

Perintah:

```

aws cognito-idp create-user-pool-client --user-pool-id us-west-2_aaaaaaaaa
--client-name MyNewClient --no-generate-secret --explicit-auth-flows
"USER_PASSWORD_AUTH" "ADMIN_NO_SRP_AUTH"

```

Output:

```

{
  "UserPoolClient": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientName": "MyNewClient",
    "ClientId": "6p3bs000no6a4ue1idruvd05ad",
    "LastModifiedDate": 1548697449.497,

```

```
"CreationDate": 1548697449.497,  
"RefreshTokenValidity": 30,  
"ExplicitAuthFlows": [  
    "USER_PASSWORD_AUTH",  
    "ADMIN_NO_SRP_AUTH"  
],  
"AllowedOAuthFlowsUserPoolClient": false  
}  
}
```

- Untuk detail API, lihat [CreateUserPoolClient](#) di Referensi AWS CLI Perintah.

## **create-user-pool-domain**

Contoh kode berikut menunjukkan cara menggunakan `create-user-pool-domain`.

### AWS CLI

Untuk membuat domain pool pengguna

Contoh ini membuat domain kumpulan pengguna baru. dengan dua alur otorisasi eksplisit: `USER_PASSWORD_AUTH` dan `ADMIN_NO_SRP_AUTH`.

Perintah:

```
aws cognito-idp create-user-pool-domain --user-pool-id us-west-2_aaaaaaaaa --domain  
my-new-domain
```

- Untuk detail API, lihat [CreateUserPoolDomain](#) di Referensi AWS CLI Perintah.

## **create-user-pool**

Contoh kode berikut menunjukkan cara menggunakan `create-user-pool`.

### AWS CLI

Untuk membuat kumpulan pengguna yang dikonfigurasi minimal

Contoh ini membuat kumpulan pengguna bernama `MyUserPool` menggunakan nilai default. Tidak ada atribut yang diperlukan dan tidak ada klien aplikasi. MFA dan keamanan tingkat lanjut dinonaktifkan.

**Perintah:**

```
aws cognito-idp create-user-pool --pool-name MyUserPool
```

**Output:**

```
{
  "UserPool": {
    "SchemaAttributes": [
      {
        "Name": "sub",
        "StringAttributeConstraints": {
          "MinLength": "1",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
        "Required": true,
        "AttributeDataType": "String",
        "Mutable": false
      },
      {
        "Name": "name",
        "StringAttributeConstraints": {
          "MinLength": "0",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
        "Required": false,
        "AttributeDataType": "String",
        "Mutable": true
      },
      {
        "Name": "given_name",
        "StringAttributeConstraints": {
          "MinLength": "0",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
        "Required": false,
        "AttributeDataType": "String",
        "Mutable": true
      },
      {
```

```
    "Name": "family_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "middle_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "nickname",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "preferred_username",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
```

```
    "Name": "profile",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "picture",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "website",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "email",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
```

```
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "email_verified",
    "Mutable": true
  },
  {
    "Name": "gender",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "birthdate",
    "StringAttributeConstraints": {
      "MinLength": "10",
      "MaxLength": "10"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "zoneinfo",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "locale",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    }
  }
}
```



```
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "phone_number",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "phone_number_verified",
    "Mutable": true
  },
  {
    "Name": "address",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "updated_at",
    "NumberAttributeConstraints": {
      "MinValue": "0"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "Number",
    "Mutable": true
  }
}
```

```

    }
  ],
  "MfaConfiguration": "OFF",
  "Name": "MyUserPool",
  "LastModifiedDate": 1547833345.777,
  "AdminCreateUserConfig": {
    "UnusedAccountValidityDays": 7,
    "AllowAdminCreateUserOnly": false
  },
  "EmailConfiguration": {},
  "Policies": {
    "PasswordPolicy": {
      "RequireLowercase": true,
      "RequireSymbols": true,
      "RequireNumbers": true,
      "MinimumLength": 8,
      "RequireUppercase": true
    }
  },
  "CreationDate": 1547833345.777,
  "EstimatedNumberOfUsers": 0,
  "Id": "us-west-2_aaaaaaaaa",
  "LambdaConfig": {}
}
}

```

Untuk membuat kumpulan pengguna dengan dua atribut yang diperlukan

Contoh ini membuat kumpulan pengguna MyUserPool. Pool dikonfigurasi untuk menerima email sebagai atribut nama pengguna. Ini juga menetapkan alamat sumber email ke alamat yang divalidasi menggunakan Amazon Simple Email Service.

Perintah:

```

aws cognito-idp create-user-pool --pool-name MyUserPool --username-
attributes "email" --email-configuration=SourceArn="arn:aws:ses:us-
east-1:111111111111:identity/
jane@example.com",ReplyToEmailAddress="jane@example.com"

```

Output:

```

{
  "UserPool": {

```

```
"SchemaAttributes": [  
  {  
    "Name": "sub",  
    "StringAttributeConstraints": {  
      "MinLength": "1",  
      "MaxLength": "2048"  
    },  
    "DeveloperOnlyAttribute": false,  
    "Required": true,  
    "AttributeDataType": "String",  
    "Mutable": false  
  },  
  {  
    "Name": "name",  
    "StringAttributeConstraints": {  
      "MinLength": "0",  
      "MaxLength": "2048"  
    },  
    "DeveloperOnlyAttribute": false,  
    "Required": false,  
    "AttributeDataType": "String",  
    "Mutable": true  
  },  
  {  
    "Name": "given_name",  
    "StringAttributeConstraints": {  
      "MinLength": "0",  
      "MaxLength": "2048"  
    },  
    "DeveloperOnlyAttribute": false,  
    "Required": false,  
    "AttributeDataType": "String",  
    "Mutable": true  
  },  
  {  
    "Name": "family_name",  
    "StringAttributeConstraints": {  
      "MinLength": "0",  
      "MaxLength": "2048"  
    },  
    "DeveloperOnlyAttribute": false,  
    "Required": false,  
    "AttributeDataType": "String",  
    "Mutable": true  
  }  
]
```

```
  },
  {
    "Name": "middle_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "nickname",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "preferred_username",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "profile",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  }
}
```

```
  },
  {
    "Name": "picture",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "website",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "email",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "email_verified",
    "Mutable": true
  },
  {
    "Name": "gender",
    "StringAttributeConstraints": {
```

```
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "birthdate",
    "StringAttributeConstraints": {
        "MinLength": "10",
        "MaxLength": "10"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "zoneinfo",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "locale",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "phone_number",
    "StringAttributeConstraints": {
```

```
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "phone_number_verified",
    "Mutable": true
},
{
    "Name": "address",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "updated_at",
    "NumberAttributeConstraints": {
        "MinValue": "0"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "Number",
    "Mutable": true
}
},
"MfaConfiguration": "OFF",
"Name": "MyUserPool",
"LastModifiedDate": 1547837788.189,
"AdminCreateUserConfig": {
    "UnusedAccountValidityDays": 7,
    "AllowAdminCreateUserOnly": false
},
}
```

```
"EmailConfiguration": {
  "ReplyToEmailAddress": "jane@example.com",
  "SourceArn": "arn:aws:ses:us-east-1:111111111111:identity/
jane@example.com"
},
"Policies": {
  "PasswordPolicy": {
    "RequireLowercase": true,
    "RequireSymbols": true,
    "RequireNumbers": true,
    "MinimumLength": 8,
    "RequireUppercase": true
  }
},
"UsernameAttributes": [
  "email"
],
"CreationDate": 1547837788.189,
"EstimatedNumberOfUsers": 0,
"Id": "us-west-2_aaaaaaaaa",
"LambdaConfig": {}
}
}
```

- Untuk detail API, lihat [CreateUserPool](#) di Referensi AWS CLI Perintah.

## delete-group

Contoh kode berikut menunjukkan cara menggunakan `delete-group`.

### AWS CLI

Untuk menghapus grup

Contoh ini menghapus grup.

Perintah:

```
aws cognito-idp delete-group --user-pool-id us-west-2_aaaaaaaaa --group-name
MyGroupName
```

- Untuk detail API, lihat [DeleteGroup](#) di Referensi AWS CLI Perintah.



## **delete-identity-provider**

Contoh kode berikut menunjukkan cara menggunakan `delete-identity-provider`.

### AWS CLI

Untuk menghapus penyedia identitas

Contoh ini menghapus penyedia identitas.

Perintah:

```
aws cognito-idp delete-identity-provider --user-pool-id us-west-2_aaaaaaaaa --
provider-name Facebook
```

- Untuk detail API, lihat [DeleteIdentityProvider](#) di Referensi AWS CLI Perintah.

## **delete-resource-server**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-server`.

### AWS CLI

Untuk menghapus server sumber daya

Contoh ini menghapus server sumber daya bernama `weather.example.com`.

Perintah:

```
aws cognito-idp delete-resource-server --user-pool-id us-west-2_aaaaaaaaa --
identifier weather.example.com
```

- Untuk detail API, lihat [DeleteResourceServer](#) di Referensi AWS CLI Perintah.

## **delete-user-attributes**

Contoh kode berikut menunjukkan cara menggunakan `delete-user-attributes`.

### AWS CLI

Untuk menghapus atribut pengguna

Contoh ini menghapus atribut pengguna "FAVORITE\_ANIMAL".

Perintah:

```
aws cognito-idp delete-user-attributes --access-token ACCESS_TOKEN --user-attribute-names "FAVORITE_ANIMAL"
```

- Untuk detail API, lihat [DeleteUserAttributes](#) di Referensi AWS CLI Perintah.

## **delete-user-pool-client**

Contoh kode berikut menunjukkan cara menggunakan `delete-user-pool-client`.

AWS CLI

Untuk menghapus klien kumpulan pengguna

Contoh ini menghapus klien kumpulan pengguna.

Perintah:

```
aws cognito-idp delete-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --client-id 38fjsnc484p94kpqsnet7mpld0
```

- Untuk detail API, lihat [DeleteUserPoolClient](#) di Referensi AWS CLI Perintah.

## **delete-user-pool-domain**

Contoh kode berikut menunjukkan cara menggunakan `delete-user-pool-domain`.

AWS CLI

Untuk menghapus domain kumpulan pengguna

`delete-user-pool-domain` Contoh berikut menghapus domain pool pengguna bernama `my-domain`

```
aws cognito-idp delete-user-pool-domain \
  --user-pool-id us-west-2_aaaaaaaaa \
  --domain my-domain
```

- Untuk detail API, lihat [DeleteUserPoolDomain](#) di Referensi AWS CLI Perintah.

## **delete-user-pool**

Contoh kode berikut menunjukkan cara menggunakan `delete-user-pool`.

### AWS CLI

Untuk menghapus kumpulan pengguna

Contoh ini menghapus kumpulan pengguna menggunakan id kumpulan pengguna, `us-west-2_aaaaaaaaaa`.

Perintah:

```
aws cognito-idp delete-user-pool --user-pool-id us-west-2_aaaaaaaaaa
```

- Untuk detail API, lihat [DeleteUserPool](#) di Referensi AWS CLI Perintah.

## **delete-user**

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

### AWS CLI

Untuk menghapus pengguna

Contoh ini menghapus pengguna.

Perintah:

```
aws cognito-idp delete-user --access-token ACCESS_TOKEN
```

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## **describe-identity-provider**

Contoh kode berikut menunjukkan cara menggunakan `describe-identity-provider`.

### AWS CLI

Untuk menggambarkan penyedia identitas

Contoh ini menjelaskan penyedia identitas bernama Facebook.

**Perintah:**

```
aws cognito-idp describe-identity-provider --user-pool-id us-west-2_aaaaaaaaa --
provider-name Facebook
```

**Output:**

```
{
  "IdentityProvider": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ProviderName": "Facebook",
    "ProviderType": "Facebook",
    "ProviderDetails": {
      "attributes_url": "https://graph.facebook.com/me?fields=",
      "attributes_url_add_attributes": "true",
      "authorize_scopes": "myscope",
      "authorize_url": "https://www.facebook.com/v2.9/dialog/oauth",
      "client_id": "11111",
      "client_secret": "11111",
      "token_request_method": "GET",
      "token_url": "https://graph.facebook.com/v2.9/oauth/access_token"
    },
    "AttributeMapping": {
      "username": "id"
    },
    "IdpIdentifiers": [],
    "LastModifiedDate": 1548105901.736,
    "CreationDate": 1548105901.736
  }
}
```

- Untuk detail API, lihat [DescribeIdentityProvider](#) di Referensi AWS CLI Perintah.

**describe-resource-server**

Contoh kode berikut menunjukkan cara menggunakan `describe-resource-server`.

**AWS CLI**

Untuk mendeskripsikan server sumber daya

Contoh ini menjelaskan server sumber daya `weather.example.com`.

**Perintah:**

```
aws cognito-idp describe-resource-server --user-pool-id us-west-2_aaaaaaaaa --
identifier weather.example.com
```

**Output:**

```
{
  "ResourceServer": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "Identifier": "weather.example.com",
    "Name": "Weather",
    "Scopes": [
      {
        "ScopeName": "weather.update",
        "ScopeDescription": "Update weather forecast"
      },
      {
        "ScopeName": "weather.read",
        "ScopeDescription": "Read weather forecasts"
      },
      {
        "ScopeName": "weather.delete",
        "ScopeDescription": "Delete a weather forecast"
      }
    ]
  }
}
```

- Untuk detail API, lihat [DescribeResourceServer](#) di Referensi AWS CLI Perintah.

**describe-risk-configuration**

Contoh kode berikut menunjukkan cara menggunakan `describe-risk-configuration`.

**AWS CLI**

Untuk menggambarkan konfigurasi risiko

Contoh ini menjelaskan konfigurasi risiko yang terkait dengan pool `us-west-2_aaaaaaaaa`.

**Perintah:**

```
aws cognito-idp describe-risk-configuration --user-pool-id us-west-2_aaaaaaaaaa
```

### Output:

```
{
  "RiskConfiguration": {
    "UserPoolId": "us-west-2_aaaaaaaaaa",
    "CompromisedCredentialsRiskConfiguration": {
      "EventFilter": [
        "SIGN_IN",
        "SIGN_UP",
        "PASSWORD_CHANGE"
      ],
      "Actions": {
        "EventAction": "BLOCK"
      }
    },
    "AccountTakeoverRiskConfiguration": {
      "NotifyConfiguration": {
        "From": "diego@example.com",
        "ReplyTo": "diego@example.com",
        "SourceArn": "arn:aws:ses:us-east-1:111111111111:identity/diego@example.com",
        "BlockEmail": {
          "Subject": "Blocked sign-in attempt",
          "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We blocked an unrecognized sign-in to your account with this information:\n<ul>\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city}, {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your password and notify us by clicking on <a href={one-click-link-invalid}>this link</a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this link</a> to let us know</pre>\n</body>\n</html>",
          "TextBody": "We blocked an unrecognized sign-in to your account with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation: {city}, {country}\nIf this sign-in was not by you, you should change your password and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you, you can follow {one-click-link-valid} to let us know"
        },
        "NoActionEmail": {
          "Subject": "New sign-in attempt",
          "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
```

```

observed an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
{country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
link</a> to let us know</pre>\n</body>\n</html>",
    "TextBody": "We observed an unrecognized sign-in to your account
with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
{city}, {country}\nIf this sign-in was not by you, you should change your password
and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
you can follow {one-click-link-valid} to let us know"
  },
  "MfaEmail": {
    "Subject": "New sign-in attempt",
    "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML email
context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We required
you to use multi-factor authentication for the following sign-in attempt:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
{country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
link</a> to let us know</pre>\n</body>\n</html>",
    "TextBody": "We required you to use multi-factor authentication
for the following sign-in attempt:\nTime: {login-time}\nDevice: {device-
name}\nLocation: {city}, {country}\nIf this sign-in was not by you, you should
change your password and notify us by clicking on {one-click-link-invalid}\nIf this
sign-in was by you, you can follow {one-click-link-valid} to let us know"
  }
},
"Actions": {
  "LowAction": {
    "Notify": true,
    "EventAction": "NO_ACTION"
  },
  "MediumAction": {
    "Notify": true,
    "EventAction": "MFA_IF_CONFIGURED"
  },
  "HighAction": {
    "Notify": true,
    "EventAction": "MFA_IF_CONFIGURED"
  }
}
}
}

```

```
}  
}
```

- Untuk detail API, lihat [DescribeRiskConfiguration](#) di Referensi AWS CLI Perintah.

## describe-user-import-job

Contoh kode berikut menunjukkan cara menggunakan `describe-user-import-job`.

### AWS CLI

Untuk mendeskripsikan pekerjaan impor pengguna

Contoh ini menjelaskan pekerjaan input pengguna.

Untuk informasi selengkapnya tentang mengimpor pengguna, lihat [Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV](#).

Perintah:

```
aws cognito-idp describe-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-id  
import-TZqNQvDRnW
```

Output:

```
{  
  "UserImportJob": {  
    "JobName": "import-Test1",  
    "JobId": "import-TZqNQvDRnW",  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "PreSignedUrl": "PRE_SIGNED URL",  
    "CreationDate": 1548271708.512,  
    "Status": "Created",  
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/  
CognitoCloudWatchLogsRole",  
    "ImportedUsers": 0,  
    "SkippedUsers": 0,  
    "FailedUsers": 0  
  }  
}
```

- Untuk detail API, lihat [DescribeUserImportJob](#) di Referensi AWS CLI Perintah.



## describe-user-pool-client

Contoh kode berikut menunjukkan cara menggunakan `describe-user-pool-client`.

### AWS CLI

Untuk mendeskripsikan klien kumpulan pengguna

Contoh ini menjelaskan klien kumpulan pengguna.

Perintah:

```
aws cognito-idp describe-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --
client-id 38fjsnc484p94kpqsnet7mpld0
```

Output:

```
{
  "UserPoolClient": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientName": "MyApp",
    "ClientId": "38fjsnc484p94kpqsnet7mpld0",
    "ClientSecret": "CLIENT_SECRET",
    "LastModifiedDate": 1548108676.163,
    "CreationDate": 1548108676.163,
    "RefreshTokenValidity": 30,
    "ReadAttributes": [
      "address",
      "birthdate",
      "custom:CustomAttr1",
      "custom:CustomAttr2",
      "email",
      "email_verified",
      "family_name",
      "gender",
      "given_name",
      "locale",
      "middle_name",
      "name",
      "nickname",
      "phone_number",
      "phone_number_verified",
      "picture",
      "preferred_username",
```

```
        "profile",
        "updated_at",
        "website",
        "zoneinfo"
    ],
    "WriteAttributes": [
        "address",
        "birthdate",
        "custom:CustomAttr1",
        "custom:CustomAttr2",
        "email",
        "family_name",
        "gender",
        "given_name",
        "locale",
        "middle_name",
        "name",
        "nickname",
        "phone_number",
        "picture",
        "preferred_username",
        "profile",
        "updated_at",
        "website",
        "zoneinfo"
    ],
    "ExplicitAuthFlows": [
        "ADMIN_NO_SRP_AUTH",
        "USER_PASSWORD_AUTH"
    ],
    "AllowedOAuthFlowsUserPoolClient": false
}
}
```

- Untuk detail API, lihat [DescribeUserPoolClient](#) di Referensi AWS CLI Perintah.

## **describe-user-pool-domain**

Contoh kode berikut menunjukkan cara menggunakan `describe-user-pool-domain`.

### AWS CLI

Untuk mendeskripsikan klien kumpulan pengguna

Contoh ini menjelaskan domain pool pengguna bernama my-domain.

Perintah:

```
aws cognito-idp describe-user-pool-domain --domain my-domain
```

Output:

```
{
  "DomainDescription": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "AWSAccountId": "111111111111",
    "Domain": "my-domain",
    "S3Bucket": "aws-cognito-prod-pdx-assets",
    "CloudFrontDistribution": "aaaaaaaaaaaaa.cloudfront.net",
    "Version": "20190128175402",
    "Status": "ACTIVE",
    "CustomDomainConfig": {}
  }
}
```

- Untuk detail API, lihat [DescribeUserPoolDomain](#) di Referensi AWS CLI Perintah.

## describe-user-pool

Contoh kode berikut menunjukkan cara menggunakan `describe-user-pool`.

AWS CLI

Untuk menggambarkan kumpulan pengguna

Contoh ini menjelaskan kumpulan pengguna dengan id kumpulan pengguna `us-west-2_aaaaaaaaa`.

Perintah:

```
aws cognito-idp describe-user-pool --user-pool-id us-west-2_aaaaaaaaa
```

Output:

```
{
  "UserPool": {
```

```
"SmsVerificationMessage": "Your verification code is {####}. ",
"SchemaAttributes": [
  {
    "Name": "sub",
    "StringAttributeConstraints": {
      "MinLength": "1",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": true,
    "AttributeDataType": "String",
    "Mutable": false
  },
  {
    "Name": "name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "given_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "family_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
```

```
    "Mutable": true
  },
  {
    "Name": "middle_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "nickname",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "preferred_username",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "profile",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
```

```
    "Mutable": true
  },
  {
    "Name": "picture",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "website",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "email",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": true,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "email_verified",
    "Mutable": true
  },
  {
    "Name": "gender",
```

```
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "birthdate",
    "StringAttributeConstraints": {
      "MinLength": "10",
      "MaxLength": "10"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "zoneinfo",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "locale",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "phone_number",
```

```
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "phone_number_verified",
    "Mutable": true
  },
  {
    "Name": "address",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "updated_at",
    "NumberAttributeConstraints": {
      "MinValue": "0"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "Number",
    "Mutable": true
  }
],
"EmailVerificationSubject": "Your verification code",
"MfaConfiguration": "OFF",
"Name": "MyUserPool",
"EmailVerificationMessage": "Your verification code is {#####}. ",
"SmsAuthenticationMessage": "Your authentication code is {#####}. ",
"LastModifiedDate": 1547763720.822,
```



```

    "AdminCreateUserConfig": {
      "InviteMessageTemplate": {
        "EmailMessage": "Your username is {username} and temporary password is
#####. ",
        "EmailSubject": "Your temporary password",
        "SMSMessage": "Your username is {username} and temporary password is
#####. "
      },
      "UnusedAccountValidityDays": 7,
      "AllowAdminCreateUserOnly": false
    },
    "EmailConfiguration": {
      "ReplyToEmailAddress": "myemail@mydomain.com"
      "SourceArn": "arn:aws:ses:us-east-1:000000000000:identity/
myemail@mydomain.com"
    },
    "AutoVerifiedAttributes": [
      "email"
    ],
    "Policies": {
      "PasswordPolicy": {
        "RequireLowercase": true,
        "RequireSymbols": true,
        "RequireNumbers": true,
        "MinimumLength": 8,
        "RequireUppercase": true
      }
    },
    "UserPoolTags": {},
    "UsernameAttributes": [
      "email"
    ],
    "CreationDate": 1547763720.822,
    "EstimatedNumberOfUsers": 1,
    "Id": "us-west-2_aaaaaaaaaa",
    "LambdaConfig": {}
  }
}

```

- Untuk detail API, lihat [DescribeUserPool](#) di Referensi AWS CLI Perintah.

## forget-device

Contoh kode berikut menunjukkan cara menggunakan `forget-device`.

### AWS CLI

Untuk melupakan perangkat

Contoh ini melupakan perangkat perangkat.

Perintah:

```
aws cognito-idp forget-device --device-key us-west-2_abcd_1234-5678
```

- Untuk detail API, lihat [ForgetDevice](#) di Referensi AWS CLI Perintah.

## forgot-password

Contoh kode berikut menunjukkan cara menggunakan `forgot-password`.

### AWS CLI

Untuk memaksa perubahan kata sandi

`forgot-password` Contoh berikut mengirimkan pesan ke `jane@example.com` untuk mengubah kata sandi mereka.

```
aws cognito-idp forgot-password --client-id 38fjsnc484p94kpbsnet7mpld0 --username jane@example.com
```

Output:

```
{
  "CodeDeliveryDetails": {
    "Destination": "j***@e***.com",
    "DeliveryMedium": "EMAIL",
    "AttributeName": "email"
  }
}
```

- Untuk detail API, lihat [ForgotPassword](#) di Referensi AWS CLI Perintah.

## get-csv-header

Contoh kode berikut menunjukkan cara menggunakan `get-csv-header`.

### AWS CLI

Untuk membuat header csv

Contoh ini membuat header csv.

Untuk informasi selengkapnya tentang mengimpor pengguna, lihat [Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV](#).

Perintah:

```
aws cognito-idp get-csv-header --user-pool-id us-west-2_aaaaaaaaa
```

Output:

```
{
  "UserPoolId": "us-west-2_aaaaaaaaa",
  "CSVHeader": [
    "name",
    "given_name",
    "family_name",
    "middle_name",
    "nickname",
    "preferred_username",
    "profile",
    "picture",
    "website",
    "email",
    "email_verified",
    "gender",
    "birthdate",
    "zoneinfo",
    "locale",
    "phone_number",
    "phone_number_verified",
    "address",
    "updated_at",
    "cognito:mfa_enabled",
    "cognito:username"
  ]
}
```

```
}
```

... Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV: <https://docs.aws.amazon.com/cognito/latest/developerguide/-import-tool.html> `cognito-user-pools-using`

- Untuk detail API, lihat [GetCsvHeader](#) di Referensi AWS CLI Perintah.

## get-group

Contoh kode berikut menunjukkan cara menggunakan `get-group`.

### AWS CLI

Untuk mendapatkan informasi tentang grup

Contoh ini mendapatkan informasi tentang grup bernama `MyGroup`.

Perintah:

```
aws cognito-idp get-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyGroup
```

Output:

```
{
  "Group": {
    "GroupName": "MyGroup",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "Description": "A sample group.",
    "LastModifiedDate": 1548270073.795,
    "CreationDate": 1548270073.795
  }
}
```

- Untuk detail API, lihat [GetGroup](#) di Referensi AWS CLI Perintah.

## get-signing-certificate

Contoh kode berikut menunjukkan cara menggunakan `get-signing-certificate`.

### AWS CLI

Untuk mendapatkan sertifikat penandatanganan

Contoh ini mendapatkan sertifikat penandatanganan untuk kumpulan pengguna.

Perintah:

```
aws cognito-idp get-signing-certificate --user-pool-id us-west-2_aaaaaaaaa
```

Output:

```
{
  "Certificate": "CERTIFICATE_DATA"
}
```

- Untuk detail API, lihat [GetSigningCertificate](#) di Referensi AWS CLI Perintah.

## get-ui-customization

Contoh kode berikut menunjukkan cara menggunakan `get-ui-customization`.

AWS CLI

Untuk mendapatkan informasi kustomisasi UI

Contoh ini mendapatkan informasi kustomisasi UI untuk kumpulan pengguna.

Perintah:

```
aws cognito-idp get-ui-customization --user-pool-id us-west-2_aaaaaaaaa
```

Output:

```
{
  "UICustomization": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientId": "ALL",
    "ImageUrl": "https://aaaaaaaaaaaaa.cloudfront.net/us-west-2_aaaaaaaaa/ALL/20190128231240/assets/images/image.jpg",
    "CSS": ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;\n}\n\n.banner-customizable {\n\tpadding: 25px 0px 25px 10px;\n\tbackground-color: lightgray;\n}\n\n.label-customizable {\n\tfont-weight: 300;\n}\n\n.textDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;
```

```

\n\tfont-size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-customizable {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable {\n\tfont-size: 14px;\n\tfont-weight: bold;\n\tmargin: 20px 0px 10px 0px;\n\theight: 40px;\n\twidth: 100%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color: #286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor: #D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor: #555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n}\n.inputField-customizable:focus {\n\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-customizable {\n\theight: 40px;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-bottom: 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color: #46b8da;\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color: #31b0d5;\n}\n.socialButton-customizable {\n\theight: 40px;\n\ttext-align: left;\n\twidth: 100%;\n\tmargin-bottom: 15px;\n}\n.redirect-customizable {\n\ttext-align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-customizable {\n\tbackground-color: #faf;\n}\n",
    "CSSVersion": "20190128231240"
  }
}

```

- Untuk detail API, lihat [GetUiCustomization](#) di Referensi AWS CLI Perintah.

## list-user-import-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-user-import-jobs`.

### AWS CLI

Untuk daftar pekerjaan impor pengguna

Contoh ini mencantumkan pekerjaan impor pengguna.

Untuk informasi selengkapnya tentang mengimpor pengguna, lihat [Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV](#).

Perintah:

```
aws cognito-idp list-user-import-jobs --user-pool-id us-west-2_aaaaaaaaa --max-results 20
```

## Output:

```
{
  "UserImportJobs": [
    {
      "JobName": "Test2",
      "JobId": "import-d00nwGA3mV",
      "UserPoolId": "us-west-2_aaaaaaaaaa",
      "PreSignedUrl": "PRE_SIGNED_URL",
      "CreationDate": 1548272793.069,
      "Status": "Created",
      "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
      "ImportedUsers": 0,
      "SkippedUsers": 0,
      "FailedUsers": 0
    },
    {
      "JobName": "Test1",
      "JobId": "import-qQ0DCt2fRh",
      "UserPoolId": "us-west-2_aaaaaaaaaa",
      "PreSignedUrl": "PRE_SIGNED_URL",
      "CreationDate": 1548271795.471,
      "Status": "Created",
      "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
      "ImportedUsers": 0,
      "SkippedUsers": 0,
      "FailedUsers": 0
    },
    {
      "JobName": "import-Test1",
      "JobId": "import-TZqNQvDRnW",
      "UserPoolId": "us-west-2_aaaaaaaaaa",
      "PreSignedUrl": "PRE_SIGNED_URL",
      "CreationDate": 1548271708.512,
      "StartDate": 1548277247.962,
      "CompletionDate": 1548277248.912,
      "Status": "Failed",
      "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
      "ImportedUsers": 0,
      "SkippedUsers": 0,
      "FailedUsers": 1,
    }
  ]
}
```

```
    "CompletionMessage": "Too many users have failed or been skipped during
the import."
  }
]
}
```

- Untuk detail API, lihat [ListUserImportJobs](#) di Referensi AWS CLI Perintah.

## list-user-pools

Contoh kode berikut menunjukkan cara menggunakan `list-user-pools`.

### AWS CLI

Untuk daftar kumpulan pengguna

Contoh ini mencantumkan hingga 20 kumpulan pengguna.

Perintah:

```
aws cognito-idp list-user-pools --max-results 20
```

Output:

```
{
  "UserPools": [
    {
      "CreationDate": 1547763720.822,
      "LastModifiedDate": 1547763720.822,
      "LambdaConfig": {},
      "Id": "us-west-2_aaaaaaaaaa",
      "Name": "MyUserPool"
    }
  ]
}
```

- Untuk detail API, lihat [ListUserPools](#) di Referensi AWS CLI Perintah.

## list-users-in-group

Contoh kode berikut menunjukkan cara menggunakan `list-users-in-group`.



## AWS CLI

Untuk mencantumkan pengguna dalam grup

Contoh ini mencantumkan pengguna dalam grup MyGroup.

Perintah:

```
aws cognito-idp list-users-in-group --user-pool-id us-west-2_aaaaaaaaa --group-name
MyGroup
```

Output:

```
{
  "Users": [
    {
      "Username": "acf10624-80bb-401a-ac61-607bee2110ec",
      "Attributes": [
        {
          "Name": "sub",
          "Value": "acf10624-80bb-401a-ac61-607bee2110ec"
        },
        {
          "Name": "custom:CustomAttr1",
          "Value": "New Value!"
        },
        {
          "Name": "email",
          "Value": "jane@example.com"
        }
      ],
      "UserCreateDate": 1548102770.284,
      "UserLastModifiedDate": 1548103204.893,
      "Enabled": true,
      "UserStatus": "CONFIRMED"
    },
    {
      "Username": "22704aa3-fc10-479a-97eb-2af5806bd327",
      "Attributes": [
        {
          "Name": "sub",
          "Value": "22704aa3-fc10-479a-97eb-2af5806bd327"
        }
      ],
    }
  ]
}
```

```
    {
      "Name": "email_verified",
      "Value": "true"
    },
    {
      "Name": "email",
      "Value": "diego@example.com"
    }
  ],
  "UserCreateDate": 1548089817.683,
  "UserLastModifiedDate": 1548089817.683,
  "Enabled": true,
  "UserStatus": "FORCE_CHANGE_PASSWORD"
}
]
```

- Untuk detail API, lihat [ListUsersInGroup](#) di Referensi AWS CLI Perintah.

## list-users

Contoh kode berikut menunjukkan cara menggunakan `list-users`.

### AWS CLI

Untuk daftar pengguna

Contoh ini mencantumkan hingga 20 pengguna.

Perintah:

```
aws cognito-idp list-users --user-pool-id us-west-2_aaaaaaaaa --limit 20
```

Output:

```
{
  "Users": [
    {
      "Username": "22704aa3-fc10-479a-97eb-2af5806bd327",
      "Enabled": true,
      "UserStatus": "FORCE_CHANGE_PASSWORD",
      "UserCreateDate": 1548089817.683,
      "UserLastModifiedDate": 1548089817.683,
```

```

    "Attributes": [
      {
        "Name": "sub",
        "Value": "22704aa3-fc10-479a-97eb-2af5806bd327"
      },
      {
        "Name": "email_verified",
        "Value": "true"
      },
      {
        "Name": "email",
        "Value": "mary@example.com"
      }
    ]
  }
]
}

```

- Untuk detail API, lihat [ListUsers](#) di Referensi AWS CLI Perintah.

## resend-confirmation-code

Contoh kode berikut menunjukkan cara menggunakan `resend-confirmation-code`.

### AWS CLI

Untuk mengirim ulang kode konfirmasi

`resend-confirmation-code` Contoh berikut mengirimkan kode konfirmasi kepada `jane`.

```

aws cognito-idp resend-confirmation-code \
  --client-id 12a3b456c7de890f11g123hijk \
  --username jane

```

### Output:

```

{
  "CodeDeliveryDetails": {
    "Destination": "j***@e***.com",
    "DeliveryMedium": "EMAIL",
    "AttributeName": "email"
  }
}

```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Mendaftar dan mengonfirmasi akun pengguna](#) di Panduan Pengembang Amazon Cognito.

- Untuk detail API, lihat [ResendConfirmationCode](#) di Referensi AWS CLI Perintah.

## respond-to-auth-challenge

Contoh kode berikut menunjukkan cara menggunakan `respond-to-auth-challenge`.

### AWS CLI

Untuk menanggapi tantangan otorisasi

Contoh ini menanggapi tantangan otorisasi yang dimulai dengan `initiate-auth`. Ini adalah respons terhadap tantangan `NEW_PASSWORD_REQUIRED`. Ini menetapkan kata sandi untuk pengguna `jane@example.com`.

Perintah:

```
aws cognito-idp respond-to-auth-challenge --client-id 3n4b5urk1ft4f13mg5e62d9ado  
--challenge-name NEW_PASSWORD_REQUIRED --challenge-responses  
USERNAME=jane@example.com,NEW_PASSWORD="password" --session "SESSION_TOKEN"
```

Output:

```
{  
  "ChallengeParameters": {},  
  "AuthenticationResult": {  
    "AccessToken": "ACCESS_TOKEN",  
    "ExpiresIn": 3600,  
    "TokenType": "Bearer",  
    "RefreshToken": "REFRESH_TOKEN",  
    "IdToken": "ID_TOKEN",  
    "NewDeviceMetadata": {  
      "DeviceKey": "us-west-2_fec070d2-fa88-424a-8ec8-b26d7198eb23",  
      "DeviceGroupKey": "-wt2ha1Zd"  
    }  
  }  
}
```

- Untuk detail API, lihat [RespondToAuthChallenge](#) di Referensi AWS CLI Perintah.

## set-risk-configuration

Contoh kode berikut menunjukkan cara menggunakan `set-risk-configuration`.

### AWS CLI

Untuk mengatur konfigurasi risiko

Contoh ini menetapkan konfigurasi risiko untuk kumpulan pengguna. Ini menetapkan tindakan acara pendaftaran ke `NO_ACTION`.

Perintah:

```
aws cognito-idp set-risk-configuration --user-pool-id us-west-2_aaaaaaaaaa --compromised-credentials-risk-configuration EventFilter=SIGN_UP,Actions={EventAction=NO_ACTION}
```

Output:

```
{
  "RiskConfiguration": {
    "UserId": "us-west-2_aaaaaaaaaa",
    "CompromisedCredentialsRiskConfiguration": {
      "EventFilter": [
        "SIGN_UP"
      ],
      "Actions": {
        "EventAction": "NO_ACTION"
      }
    }
  }
}
```

- Untuk detail API, lihat [SetRiskConfiguration](#) di Referensi AWS CLI Perintah.

## set-ui-customization

Contoh kode berikut menunjukkan cara menggunakan `set-ui-customization`.

## AWS CLI

Untuk mengatur kustomisasi UI

Contoh ini menyesuaikan pengaturan CSS untuk kumpulan pengguna.

Perintah:

```
aws cognito-idp set-ui-customization --user-pool-id us-west-2_aaaaaaaaa --css
".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;\n}\n.banner-
customizable {\n\tpadding: 25px 0px 25px 10px;\n\tbackground-color: lightgray;
\n}\n.label-customizable {\n\tfont-weight: 300;\n}\n.textDescription-customizable
{\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-
size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-
bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-customizable
{\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable
{\n\tfont-size: 14px;\n\tfont-weight: bold;\n\tmargin: 20px 0px 10px 0px;\n
\theight: 40px;\n\twidth: 100%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;
\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;
\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor:
#D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor:
#555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n}\n.inputField-
customizable:focus {\n\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-
customizable {\n\theight: 40px;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-
bottom: 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color:
#46b8da;\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#31b0d5;\n}\n.socialButton-customizable {\n\theight: 40px;\n\ttext-align: left;
\n\twidth: 100%;\n\tmargin-bottom: 15px;\n}\n.redirect-customizable {\n\ttext-
align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;
\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-
customizable {\n\tbackground-color: #faf;\n}\n"
```

Output:

```
{
  "UICustomization": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientId": "ALL",
    "CSS": ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;
\n}\n.banner-customizable {\n\tpadding: 25px 0px 25px 10px;\n\tbackground-color:
lightgray;\n}\n.label-customizable {\n\tfont-weight: 300;\n}\n.textDescription-
customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;
```

```

\n\tfont-size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-customizable {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable {\n\tfont-size: 14px;\n\tfont-weight: bold;\n\tmargin: 20px 0px 10px 0px;\n\theight: 40px;\n\twidth: 100%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color: #286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor: #D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor: #555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n}\n.inputField-customizable:focus {\n\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-customizable {\n\theight: 40px;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-bottom: 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color: #46b8da;\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color: #31b0d5;\n}\n.socialButton-customizable {\n\theight: 40px;\n\ttext-align: left;\n\twidth: 100%;\n\tmargin-bottom: 15px;\n}\n.redirect-customizable {\n\ttext-align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-customizable {\n\tbackground-color: #faf;\n}\n",
    "CSSVersion": "20190129172214"
  }
}

```

- Untuk detail API, lihat [SetUiCustomization](#) di Referensi AWS CLI Perintah.

## set-user-mfa-preference

Contoh kode berikut menunjukkan cara menggunakan `set-user-mfa-preference`.

### AWS CLI

Untuk mengatur pengaturan MFA pengguna

`set-user-mfa-preference` Contoh berikut memodifikasi opsi pengiriman MFA. Ini mengubah media pengiriman MFA menjadi SMS.

```

aws cognito-idp set-user-mfa-preference \
  --access-token "eyJra12345EXAMPLE" \
  --software-token-mfa-settings Enabled=true,PreferredMfa=true \
  --sms-mfa-settings Enabled=false,PreferredMfa=false

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan MFA ke kumpulan pengguna](#) di Panduan Pengembang Amazon Cognito.

- Untuk detail API, lihat [SetUserMfaPreference](#) di Referensi AWS CLI Perintah.

## set-user-settings

Contoh kode berikut menunjukkan cara menggunakan `set-user-settings`.

### AWS CLI

Untuk mengatur pengaturan pengguna

Contoh ini menetapkan preferensi pengiriman MFA ke EMAIL.

Perintah:

```
aws cognito-idp set-user-settings --access-token ACCESS_TOKEN --mfa-options
DeliveryMedium=EMAIL
```

- Untuk detail API, lihat [SetUserSettings](#) di Referensi AWS CLI Perintah.

## sign-up

Contoh kode berikut menunjukkan cara menggunakan `sign-up`.

### AWS CLI

Untuk mendaftar pengguna

Contoh ini mendaftar `jane@example.com`.

Perintah:

```
aws cognito-idp sign-up --client-id 3n4b5urk1ft4f13mg5e62d9ado --
username jane@example.com --password PASSWORD --user-attributes
Name="email",Value="jane@example.com" Name="name",Value="Jane"
```

Output:

```
{
  "UserConfirmed": false,
```



```
"UserSub": "e04d60a6-45dc-441c-a40b-e25a787d4862"
}
```

- Untuk detail API, lihat [SignUp](#) di Referensi AWS CLI Perintah.

## start-user-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-user-import-job`.

### AWS CLI

Untuk memulai pekerjaan impor pengguna

Contoh ini memulai pekerjaan input pengguna.

Untuk informasi selengkapnya tentang mengimpor pengguna, lihat [Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV](#).

Perintah:

```
aws cognito-idp start-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-id
import-TZqNQvDRnW
```

Output:

```
{
  "UserImportJob": {
    "JobName": "import-Test10",
    "JobId": "import-lmpxS0uIzH",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548278378.928,
    "StartDate": 1548278397.334,
    "Status": "Pending",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0
  }
}
```

- Untuk detail API, lihat [StartUserImportJob](#) di Referensi AWS CLI Perintah.

## stop-user-import-job

Contoh kode berikut menunjukkan cara menggunakan `stop-user-import-job`.

### AWS CLI

Untuk menghentikan pekerjaan impor pengguna

Contoh ini menghentikan pekerjaan input pengguna.

Untuk informasi selengkapnya tentang mengimpor pengguna, lihat [Mengimpor Pengguna ke Kumpulan Pengguna Dari File CSV](#).

Perintah:

```
aws cognito-idp stop-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-id
import-TZqNQvDRnW
```

Output:

```
{
  "UserImportJob": {
    "JobName": "import-Test5",
    "JobId": "import-Fx0kARISFL",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548278576.259,
    "StartDate": 1548278623.366,
    "CompletionDate": 1548278626.741,
    "Status": "Stopped",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0,
    "CompletionMessage": "The Import Job was stopped by the developer."
  }
}
```

- Untuk detail API, lihat [StopUserImportJob](#) di Referensi AWS CLI Perintah.

## update-auth-event-feedback

Contoh kode berikut menunjukkan cara menggunakan `update-auth-event-feedback`.

### AWS CLI

Untuk memperbarui umpan balik acara autentikasi

Contoh ini memperbarui umpan balik acara otorisasi. Ini menandai acara "Valid".

Perintah:

```
aws cognito-idp update-auth-event-feedback --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com --event-id EVENT_ID --feedback-token FEEDBACK_TOKEN --feedback-value "Valid"
```

- Untuk detail API, lihat [UpdateAuthEventFeedback](#) di Referensi AWS CLI Perintah.

## update-device-status

Contoh kode berikut menunjukkan cara menggunakan `update-device-status`.

### AWS CLI

Untuk memperbarui status perangkat

Contoh ini memperbarui status perangkat ke "not\_remembered".

Perintah:

```
aws cognito-idp update-device-status --access-token ACCESS_TOKEN --device-key DEVICE_KEY --device-remembered-status "not_remembered"
```

- Untuk detail API, lihat [UpdateDeviceStatus](#) di Referensi AWS CLI Perintah.

## update-group

Contoh kode berikut menunjukkan cara menggunakan `update-group`.

### AWS CLI

Untuk memperbarui grup

Contoh ini memperbarui deskripsi dan prioritas untuk `MyGroup`

Perintah:

```
aws cognito-idp update-group --user-pool-id us-west-2_aaaaaaaa --group-name MyGroup
--description "New description" --precedence 2
```

Output:

```
{
  "Group": {
    "GroupName": "MyGroup",
    "UserPoolId": "us-west-2_aaaaaaaa",
    "Description": "New description",
    "RoleArn": "arn:aws:iam::111111111111:role/MyRole",
    "Precedence": 2,
    "LastModifiedDate": 1548800862.812,
    "CreationDate": 1548097827.125
  }
}
```

- Untuk detail API, lihat [UpdateGroup](#) di Referensi AWS CLI Perintah.

## update-resource-server

Contoh kode berikut menunjukkan cara menggunakan `update-resource-server`.

AWS CLI

Untuk memperbarui server sumber daya

Contoh ini memperbarui Weather server sumber daya. Ini menambahkan ruang lingkup baru.

Perintah:

```
aws cognito-idp update-resource-server --user-pool-id us-west-2_aaaaaaaa
--identifier weather.example.com --name Weather --scopes
ScopeName=NewScope,ScopeDescription="New scope description"
```

Output:

```
{
```

```
"ResourceServer": {
  "UserPoolId": "us-west-2_aaaaaaaaa",
  "Identifier": "weather.example.com",
  "Name": "Happy",
  "Scopes": [
    {
      "ScopeName": "NewScope",
      "ScopeDescription": "New scope description"
    }
  ]
}
```

- Untuk detail API, lihat [UpdateResourceServer](#) di Referensi AWS CLI Perintah.

## update-user-attributes

Contoh kode berikut menunjukkan cara menggunakan `update-user-attributes`.

### AWS CLI

Untuk memperbarui atribut pengguna

Contoh ini memperbarui atribut pengguna “nama panggilan”.

Perintah:

```
aws cognito-idp update-user-attributes --access-token ACCESS_TOKEN --user-attributes
Name="nickname",Value="Dan"
```

- Untuk detail API, lihat [UpdateUserAttributes](#) di Referensi AWS CLI Perintah.

## update-user-pool-client

Contoh kode berikut menunjukkan cara menggunakan `update-user-pool-client`.

### AWS CLI

Untuk memperbarui klien kumpulan pengguna

Contoh ini memperbarui nama klien kumpulan pengguna. Itu juga menambahkan atribut yang dapat ditulis “nama panggilan”.

**Perintah:**

```
aws cognito-idp update-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --client-id 3n4b5urk1ft4fl3mg5e62d9ado --client-name "NewClientName" --write-attributes "nickname"
```

**Output:**

```
{
  "UserPoolClient": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientName": "NewClientName",
    "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",
    "LastModifiedDate": 1548802761.334,
    "CreationDate": 1548178931.258,
    "RefreshTokenValidity": 30,
    "WriteAttributes": [
      "nickname"
    ],
    "AllowedOAuthFlowsUserPoolClient": false
  }
}
```

- Untuk detail API, lihat [UpdateUserPoolClient](#) di Referensi AWS CLI Perintah.

**update-user-pool**

Contoh kode berikut menunjukkan cara menggunakan `update-user-pool`.

**AWS CLI**

Untuk memperbarui kumpulan pengguna

Contoh ini menambahkan tag ke kumpulan pengguna.

**Perintah:**

```
aws cognito-idp update-user-pool --user-pool-id us-west-2_aaaaaaaaa --user-pool-tags Team=Blue,Area=West
```

- Untuk detail API, lihat [UpdateUserPool](#) di Referensi AWS CLI Perintah.

## Amazon Comprehend contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon AWS Command Line Interface Comprehend.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-detect-dominant-language**

Contoh kode berikut menunjukkan cara menggunakan `batch-detect-dominant-language`.

AWS CLI

Untuk mendeteksi bahasa dominan dari beberapa teks input

`batch-detect-dominant-language` Contoh berikut menganalisis beberapa teks masukan dan mengembalikan bahasa dominan masing-masing. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```
aws comprehend batch-detect-dominant-language \  
  --text-list "Physics is the natural science that involves the study of matter  
and its motion and behavior through space and time, along with related concepts  
such as energy and force."
```

Output:

```
{  
  "ResultList": [  
    {
```

```

        "Index": 0,
        "Languages": [
            {
                "LanguageCode": "en",
                "Score": 0.9986501932144165
            }
        ]
    },
    "ErrorList": []
}

```

Untuk informasi selengkapnya, lihat [Bahasa Dominan](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [BatchDetectDominantLanguage](#) di Referensi AWS CLI Perintah.

## batch-detect-entities

Contoh kode berikut menunjukkan cara menggunakan `batch-detect-entities`.

### AWS CLI

Untuk mendeteksi entitas dari beberapa teks masukan

`batch-detect-entities` Contoh berikut menganalisis beberapa teks masukan dan mengembalikan entitas bernama masing-masing. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```

aws comprehend batch-detect-entities \
  --language-code en \
  --text-list "Dear Jane, Your AnyCompany Financial Services LLC credit card
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
31st." "Please send customer feedback to Sunshine Spa, 123 Main St, Anywhere or to
Alice at AnySpa@example.com."

```

Output:

```

{
  "ResultList": [
    {
      "Index": 0,
      "Entities": [

```



```
    {
      "Score": 0.9985517859458923,
      "Type": "PERSON",
      "Text": "Jane",
      "BeginOffset": 5,
      "EndOffset": 9
    },
    {
      "Score": 0.9767839312553406,
      "Type": "ORGANIZATION",
      "Text": "AnyCompany Financial Services, LLC",
      "BeginOffset": 16,
      "EndOffset": 50
    },
    {
      "Score": 0.9856694936752319,
      "Type": "OTHER",
      "Text": "1111-XXXX-1111-XXXX",
      "BeginOffset": 71,
      "EndOffset": 90
    },
    {
      "Score": 0.9652159810066223,
      "Type": "QUANTITY",
      "Text": ".53",
      "BeginOffset": 116,
      "EndOffset": 119
    },
    {
      "Score": 0.9986667037010193,
      "Type": "DATE",
      "Text": "July 31st",
      "BeginOffset": 135,
      "EndOffset": 144
    }
  ]
},
{
  "Index": 1,
  "Entities": [
    {
      "Score": 0.720084547996521,
      "Type": "ORGANIZATION",
      "Text": "Sunshine Spa",
```

```
        "BeginOffset": 33,
        "EndOffset": 45
    },
    {
        "Score": 0.9865870475769043,
        "Type": "LOCATION",
        "Text": "123 Main St",
        "BeginOffset": 47,
        "EndOffset": 58
    },
    {
        "Score": 0.5895616412162781,
        "Type": "LOCATION",
        "Text": "Anywhere",
        "BeginOffset": 60,
        "EndOffset": 68
    },
    {
        "Score": 0.6809214353561401,
        "Type": "PERSON",
        "Text": "Alice",
        "BeginOffset": 75,
        "EndOffset": 80
    },
    {
        "Score": 0.9979087114334106,
        "Type": "OTHER",
        "Text": "AnySpa@example.com",
        "BeginOffset": 84,
        "EndOffset": 99
    }
    ]
}
],
"ErrorList": []
}
```

Untuk informasi selengkapnya, lihat [Entitas](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [BatchDetectEntities](#) di Referensi AWS CLI Perintah.

## batch-detect-key-phrases

Contoh kode berikut menunjukkan cara menggunakan `batch-detect-key-phrases`.

### AWS CLI

Untuk mendeteksi frase kunci dari beberapa input teks

`batch-detect-key-phrases` Contoh berikut menganalisis beberapa teks masukan dan mengembalikan frase kata benda kunci masing-masing. Skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi juga merupakan output.

```
aws comprehend batch-detect-key-phrases \  
  --language-code en \  
  --text-list "Hello Zhang Wei, I am John, writing to you about the trip for next  
Saturday." "Dear Jane, Your AnyCompany Financial Services LLC credit card account  
1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July 31st."  
"Please send customer feedback to Sunshine Spa, 123 Main St, Anywhere or to Alice  
at AnySpa@example.com."
```

Output:

```
{  
  "ResultList": [  
    {  
      "Index": 0,  
      "KeyPhrases": [  
        {  
          "Score": 0.99700927734375,  
          "Text": "Zhang Wei",  
          "BeginOffset": 6,  
          "EndOffset": 15  
        },  
        {  
          "Score": 0.9929308891296387,  
          "Text": "John",  
          "BeginOffset": 22,  
          "EndOffset": 26  
        },  
        {  
          "Score": 0.9997230172157288,  
          "Text": "the trip",  
          "BeginOffset": 49,  
          "EndOffset": 55  
        }  
      ]  
    }  
  ]  
}
```

```
        "EndOffset": 57
      },
      {
        "Score": 0.9999470114707947,
        "Text": "next Saturday",
        "BeginOffset": 62,
        "EndOffset": 75
      }
    ]
  },
  {
    "Index": 1,
    "KeyPhrases": [
      {
        "Score": 0.8358274102210999,
        "Text": "Dear Jane",
        "BeginOffset": 0,
        "EndOffset": 9
      },
      {
        "Score": 0.989359974861145,
        "Text": "Your AnyCompany Financial Services",
        "BeginOffset": 11,
        "EndOffset": 45
      },
      {
        "Score": 0.8812323808670044,
        "Text": "LLC credit card account 1111-XXXX-1111-XXXX",
        "BeginOffset": 47,
        "EndOffset": 90
      },
      {
        "Score": 0.9999381899833679,
        "Text": "a minimum payment",
        "BeginOffset": 95,
        "EndOffset": 112
      },
      {
        "Score": 0.9997439980506897,
        "Text": ".53",
        "BeginOffset": 116,
        "EndOffset": 119
      }
    ]
  }
}
```

```
        "Score": 0.996875524520874,
        "Text": "July 31st",
        "BeginOffset": 135,
        "EndOffset": 144
      }
    ]
  },
  {
    "Index": 2,
    "KeyPhrases": [
      {
        "Score": 0.9990295767784119,
        "Text": "customer feedback",
        "BeginOffset": 12,
        "EndOffset": 29
      },
      {
        "Score": 0.9994127750396729,
        "Text": "Sunshine Spa",
        "BeginOffset": 33,
        "EndOffset": 45
      },
      {
        "Score": 0.9892991185188293,
        "Text": "123 Main St",
        "BeginOffset": 47,
        "EndOffset": 58
      },
      {
        "Score": 0.9969810843467712,
        "Text": "Alice",
        "BeginOffset": 75,
        "EndOffset": 80
      },
      {
        "Score": 0.9703696370124817,
        "Text": "AnySpa@example.com",
        "BeginOffset": 84,
        "EndOffset": 99
      }
    ]
  }
],
"ErrorList": []
```

```
}
```

Untuk informasi selengkapnya, lihat [Frasa Kunci](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [BatchDetectKeyPhrases](#) di Referensi AWS CLI Perintah.

## batch-detect-sentiment

Contoh kode berikut menunjukkan cara menggunakan `batch-detect-sentiment`.

### AWS CLI

Untuk mendeteksi sentimen yang berlaku dari beberapa teks masukan

`batch-detect-sentiment` Contoh berikut menganalisis beberapa teks masukan dan mengembalikan sentimen yang berlaku (POSITIVE, NEUTRALMIXED, atau NEGATIVE, masing-masing).

```
aws comprehend batch-detect-sentiment \
  --text-list "That movie was very boring, I can't believe it was over four hours
long." "It is a beautiful day for hiking today." "My meal was okay, I'm excited to
try other restaurants." \
  --language-code en
```

### Output:

```
{
  "ResultList": [
    {
      "Index": 0,
      "Sentiment": "NEGATIVE",
      "SentimentScore": {
        "Positive": 0.00011316669406369328,
        "Negative": 0.9995445609092712,
        "Neutral": 0.00014722718333359808,
        "Mixed": 0.00019498742767609656
      }
    },
    {
      "Index": 1,
      "Sentiment": "POSITIVE",
      "SentimentScore": {
        "Positive": 0.9981263279914856,
```

```

        "Negative": 0.00015240783977787942,
        "Neutral": 0.0013876151060685515,
        "Mixed": 0.00033366199932061136
    }
},
{
    "Index": 2,
    "Sentiment": "MIXED",
    "SentimentScore": {
        "Positive": 0.15930435061454773,
        "Negative": 0.11471917480230331,
        "Neutral": 0.26897063851356506,
        "Mixed": 0.45700588822364807
    }
}
],
"ErrorList": []
}

```

Untuk informasi selengkapnya, lihat [Sentimen](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [BatchDetectSentiment](#) di Referensi AWS CLI Perintah.

## batch-detect-syntax

Contoh kode berikut menunjukkan cara menggunakan `batch-detect-syntax`.

### AWS CLI

Untuk memeriksa sintaks dan bagian ucapan kata-kata dalam beberapa teks masukan

`batch-detect-syntax` Contoh berikut menganalisis sintaks dari beberapa teks masukan dan mengembalikan bagian-bagian yang berbeda dari pidato. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```

aws comprehend batch-detect-syntax \
  --text-list "It is a beautiful day." "Can you please pass the salt?" "Please pay
the bill before the 31st." \
  --language-code en

```

Output:

```
{
```

```
"ResultList": [  
  {  
    "Index": 0,  
    "SyntaxTokens": [  
      {  
        "TokenId": 1,  
        "Text": "It",  
        "BeginOffset": 0,  
        "EndOffset": 2,  
        "PartOfSpeech": {  
          "Tag": "PRON",  
          "Score": 0.9999740719795227  
        }  
      },  
      {  
        "TokenId": 2,  
        "Text": "is",  
        "BeginOffset": 3,  
        "EndOffset": 5,  
        "PartOfSpeech": {  
          "Tag": "VERB",  
          "Score": 0.999937117099762  
        }  
      },  
      {  
        "TokenId": 3,  
        "Text": "a",  
        "BeginOffset": 6,  
        "EndOffset": 7,  
        "PartOfSpeech": {  
          "Tag": "DET",  
          "Score": 0.9999926686286926  
        }  
      },  
      {  
        "TokenId": 4,  
        "Text": "beautiful",  
        "BeginOffset": 8,  
        "EndOffset": 17,  
        "PartOfSpeech": {  
          "Tag": "ADJ",  
          "Score": 0.9987891912460327  
        }  
      }  
    ]  
  },  
  ]  
]
```



```
    {
      "TokenId": 5,
      "Text": "day",
      "BeginOffset": 18,
      "EndOffset": 21,
      "PartOfSpeech": {
        "Tag": "NOUN",
        "Score": 0.9999778866767883
      }
    },
    {
      "TokenId": 6,
      "Text": ".",
      "BeginOffset": 21,
      "EndOffset": 22,
      "PartOfSpeech": {
        "Tag": "PUNCT",
        "Score": 0.9999974966049194
      }
    }
  ]
},
{
  "Index": 1,
  "SyntaxTokens": [
    {
      "TokenId": 1,
      "Text": "Can",
      "BeginOffset": 0,
      "EndOffset": 3,
      "PartOfSpeech": {
        "Tag": "AUX",
        "Score": 0.9999770522117615
      }
    },
    {
      "TokenId": 2,
      "Text": "you",
      "BeginOffset": 4,
      "EndOffset": 7,
      "PartOfSpeech": {
        "Tag": "PRON",
        "Score": 0.9999986886978149
      }
    }
  ]
}
```

```
    },
    {
      "TokenId": 3,
      "Text": "please",
      "BeginOffset": 8,
      "EndOffset": 14,
      "PartOfSpeech": {
        "Tag": "INTJ",
        "Score": 0.9681622385978699
      }
    },
    {
      "TokenId": 4,
      "Text": "pass",
      "BeginOffset": 15,
      "EndOffset": 19,
      "PartOfSpeech": {
        "Tag": "VERB",
        "Score": 0.9999874830245972
      }
    },
    {
      "TokenId": 5,
      "Text": "the",
      "BeginOffset": 20,
      "EndOffset": 23,
      "PartOfSpeech": {
        "Tag": "DET",
        "Score": 0.9999827146530151
      }
    },
    {
      "TokenId": 6,
      "Text": "salt",
      "BeginOffset": 24,
      "EndOffset": 28,
      "PartOfSpeech": {
        "Tag": "NOUN",
        "Score": 0.9995040893554688
      }
    },
    {
      "TokenId": 7,
      "Text": "?",
```

```
        "BeginOffset": 28,  
        "EndOffset": 29,  
        "PartOfSpeech": {  
            "Tag": "PUNCT",  
            "Score": 0.999998152256012  
        }  
    }  
]  
},  
{  
    "Index": 2,  
    "SyntaxTokens": [  
        {  
            "TokenId": 1,  
            "Text": "Please",  
            "BeginOffset": 0,  
            "EndOffset": 6,  
            "PartOfSpeech": {  
                "Tag": "INTJ",  
                "Score": 0.9997857809066772  
            }  
        },  
        {  
            "TokenId": 2,  
            "Text": "pay",  
            "BeginOffset": 7,  
            "EndOffset": 10,  
            "PartOfSpeech": {  
                "Tag": "VERB",  
                "Score": 0.9999252557754517  
            }  
        },  
        {  
            "TokenId": 3,  
            "Text": "the",  
            "BeginOffset": 11,  
            "EndOffset": 14,  
            "PartOfSpeech": {  
                "Tag": "DET",  
                "Score": 0.9999842643737793  
            }  
        },  
        {  
            "TokenId": 4,
```

```
    "Text": "bill",
    "BeginOffset": 15,
    "EndOffset": 19,
    "PartOfSpeech": {
      "Tag": "NOUN",
      "Score": 0.9999588131904602
    }
  },
  {
    "TokenId": 5,
    "Text": "before",
    "BeginOffset": 20,
    "EndOffset": 26,
    "PartOfSpeech": {
      "Tag": "ADP",
      "Score": 0.9958304762840271
    }
  },
  {
    "TokenId": 6,
    "Text": "the",
    "BeginOffset": 27,
    "EndOffset": 30,
    "PartOfSpeech": {
      "Tag": "DET",
      "Score": 0.9999947547912598
    }
  },
  {
    "TokenId": 7,
    "Text": "31st",
    "BeginOffset": 31,
    "EndOffset": 35,
    "PartOfSpeech": {
      "Tag": "NOUN",
      "Score": 0.9924124479293823
    }
  },
  {
    "TokenId": 8,
    "Text": ".",
    "BeginOffset": 35,
    "EndOffset": 36,
    "PartOfSpeech": {
```

```

        "Tag": "PUNCT",
        "Score": 0.9999955892562866
      }
    ]
  },
  "ErrorList": []
}

```

Untuk informasi selengkapnya, lihat [Analisis Sintaks](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [BatchDetectSyntax](#) di Referensi AWS CLI Perintah.

## batch-detect-targeted-sentiment

Contoh kode berikut menunjukkan cara menggunakan `batch-detect-targeted-sentiment`.

### AWS CLI

Untuk mendeteksi sentimen dan setiap entitas bernama untuk beberapa teks masukan

`batch-detect-targeted-sentiment` Contoh berikut menganalisis beberapa teks masukan dan mengembalikan entitas bernama bersama dengan sentimen yang berlaku melekat pada setiap entitas. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```

aws comprehend batch-detect-targeted-sentiment \
  --language-code en \
  --text-list "That movie was really boring, the original was way more
entertaining" "The trail is extra beautiful today." "My meal was just okay."

```

Output:

```

{
  "ResultList": [
    {
      "Index": 0,
      "Entities": [
        {
          "DescriptiveMentionIndex": [

```

```
    0
  ],
  "Mentions": [
    {
      "Score": 0.9999009966850281,
      "GroupScore": 1.0,
      "Text": "movie",
      "Type": "MOVIE",
      "MentionSentiment": {
        "Sentiment": "NEGATIVE",
        "SentimentScore": {
          "Positive": 0.13887299597263336,
          "Negative": 0.8057460188865662,
          "Neutral": 0.05525200068950653,
          "Mixed": 0.00012799999967683107
        }
      }
    },
    {
      "BeginOffset": 5,
      "EndOffset": 10
    }
  ]
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "Score": 0.9921110272407532,
      "GroupScore": 1.0,
      "Text": "original",
      "Type": "MOVIE",
      "MentionSentiment": {
        "Sentiment": "POSITIVE",
        "SentimentScore": {
          "Positive": 0.9999989867210388,
          "Negative": 9.99999974752427e-07,
          "Neutral": 0.0,
          "Mixed": 0.0
        }
      }
    },
    {
      "BeginOffset": 34,
      "EndOffset": 42
    }
  ]
}
```

```
    ]
  }
]
},
{
  "Index": 1,
  "Entities": [
    {
      "DescriptiveMentionIndex": [
        0
      ],
      "Mentions": [
        {
          "Score": 0.7545599937438965,
          "GroupScore": 1.0,
          "Text": "trail",
          "Type": "OTHER",
          "MentionSentiment": {
            "Sentiment": "POSITIVE",
            "SentimentScore": {
              "Positive": 1.0,
              "Negative": 0.0,
              "Neutral": 0.0,
              "Mixed": 0.0
            }
          }
        },
        {
          "BeginOffset": 4,
          "EndOffset": 9
        }
      ]
    }
  ],
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "Score": 0.9999960064888,
      "GroupScore": 1.0,
      "Text": "today",
      "Type": "DATE",
      "MentionSentiment": {
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
```

```

        "Positive": 9.000000318337698e-06,
        "Negative": 1.999999949504854e-06,
        "Neutral": 0.9999859929084778,
        "Mixed": 3.99999989900971e-06
    }
},
"BeginOffset": 29,
"EndOffset": 34
}
]
}
]
},
{
    "Index": 2,
    "Entities": [
        {
            "DescriptiveMentionIndex": [
                0
            ],
            "Mentions": [
                {
                    "Score": 0.9999880194664001,
                    "GroupScore": 1.0,
                    "Text": "My",
                    "Type": "PERSON",
                    "MentionSentiment": {
                        "Sentiment": "NEUTRAL",
                        "SentimentScore": {
                            "Positive": 0.0,
                            "Negative": 0.0,
                            "Neutral": 1.0,
                            "Mixed": 0.0
                        }
                    }
                }
            ],
            "BeginOffset": 0,
            "EndOffset": 2
        }
    ],
    "DescriptiveMentionIndex": [
        0
    ],

```



```

        "Mentions": [
            {
                "Score": 0.9995260238647461,
                "GroupScore": 1.0,
                "Text": "meal",
                "Type": "OTHER",
                "MentionSentiment": {
                    "Sentiment": "NEUTRAL",
                    "SentimentScore": {
                        "Positive": 0.04695599898695946,
                        "Negative": 0.003226999891921878,
                        "Neutral": 0.6091709733009338,
                        "Mixed": 0.34064599871635437
                    }
                },
                "BeginOffset": 3,
                "EndOffset": 7
            }
        ]
    },
    "ErrorList": []
}

```

Untuk informasi selengkapnya, lihat [Sentimen Bertarget](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [BatchDetectTargetedSentiment](#) di Referensi AWS CLI Perintah.

## classify-document

Contoh kode berikut menunjukkan cara menggunakan `classify-document`.

### AWS CLI

Untuk mengklasifikasikan dokumen dengan titik akhir khusus model

`classify-document` Contoh berikut mengklasifikasikan dokumen dengan titik akhir model kustom. Model dalam contoh ini dilatih pada dataset yang berisi pesan sms berlabel spam atau non-spam, atau, “ham”.

```
aws comprehend classify-document \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint \  
  --text "CONGRATULATIONS! TXT 1235550100 to win $5000"
```

### Output:

```
{  
  "Classes": [  
    {  
      "Name": "spam",  
      "Score": 0.9998599290847778  
    },  
    {  
      "Name": "ham",  
      "Score": 0.00014001205272506922  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Klasifikasi Kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ClassifyDocument](#) di Referensi AWS CLI Perintah.

## contains-pii-entities

Contoh kode berikut menunjukkan cara menggunakan `contains-pii-entities`.

### AWS CLI

Untuk menganalisis teks masukan untuk keberadaan informasi PII

`contains-pii-entities` Contoh berikut menganalisis teks input untuk keberadaan informasi identitas pribadi (PII) dan mengembalikan label jenis entitas PII yang diidentifikasi seperti nama, alamat, nomor rekening bank, atau nomor telepon.

```
aws comprehend contains-pii-entities \  
  --language-code en \  
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC  
credit card"
```

```
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
July 31st. Based on your autopay settings,
we will withdraw your payment on the due date from your bank account number
XXXXXX1111 with the routing number XXXXX0000.
Customer feedback for Sunshine Spa, 100 Main St, Anywhere. Send comments to
Alice at AnySpa@example.com."
```

## Output:

```
{
  "Labels": [
    {
      "Name": "NAME",
      "Score": 1.0
    },
    {
      "Name": "EMAIL",
      "Score": 1.0
    },
    {
      "Name": "BANK_ACCOUNT_NUMBER",
      "Score": 0.9995794296264648
    },
    {
      "Name": "BANK_ROUTING",
      "Score": 0.9173126816749573
    },
    {
      "Name": "CREDIT_DEBIT_NUMBER",
      "Score": 1.0
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Informasi Identifikasi Pribadi \(PII\)](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ContainsPiiEntities](#) di Referensi AWS CLI Perintah.

## create-dataset

Contoh kode berikut menunjukkan cara menggunakan `create-dataset`.

## AWS CLI

Untuk membuat dataset flywheel

`create-dataset` Contoh berikut membuat dataset untuk flywheel. Dataset ini akan digunakan sebagai data pelatihan tambahan seperti yang ditentukan oleh `--dataset-type` tag.

```
aws comprehend create-dataset \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity \  
  --dataset-name example-dataset \  
  --dataset-type "TRAIN" \  
  --input-data-config file://inputConfig.json
```

Isi dari `file://inputConfig.json`:

```
{  
  "DataFormat": "COMPREHEND_CSV",  
  "DocumentClassifierInputDataConfig": {  
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/training-data.csv"  
  }  
}
```

Output:

```
{  
  "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset"  
}
```

Untuk informasi selengkapnya, lihat [Ikhtisar Roda Gila di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [CreateDataset](#) di Referensi AWS CLI Perintah.

## **create-document-classifier**

Contoh kode berikut menunjukkan cara menggunakan `create-document-classifier`.

## AWS CLI

Untuk membuat pengklasifikasi dokumen untuk mengkategorikan dokumen

`create-document-classifier` Contoh berikut memulai proses pelatihan untuk model pengklasifikasi dokumen. File data pelatihan `training.csv`, terletak di `--input-data-config` tag. `training.csv` adalah dokumen dua kolom di mana label, atau, klasifikasi disediakan di kolom pertama dan dokumen disediakan di kolom kedua.

```
aws comprehend create-document-classifier \  
  --document-classifier-name example-classifier \  
  --data-access-arn arn:aws:comprehend:us-west-2:111122223333:pii-entities-  
detection-job/123456abcdeb0e11022f22a11EXAMPLE \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \  
  --language-code en
```

Output:

```
{  
  "DocumentClassifierArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier"  
}
```

Untuk informasi selengkapnya, lihat [Klasifikasi Kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [CreateDocumentClassifier](#) di Referensi AWS CLI Perintah.

## **create-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `create-endpoint`.

### AWS CLI

Untuk membuat endpoint untuk model kustom

`create-endpoint` Contoh berikut membuat titik akhir untuk inferensi sinkron untuk model kustom yang dilatih sebelumnya.

```
aws comprehend create-endpoint \  
  --endpoint-name example-classifier-endpoint-1 \  
  --model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier \  
  --desired-inference-units 1
```

**Output:**

```
{
  "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint-1"
}
```

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [CreateEndpoint](#) di Referensi AWS CLI Perintah.

**create-entity-recognizer**

Contoh kode berikut menunjukkan cara menggunakan `create-entity-recognizer`.

**AWS CLI**

Untuk membuat pengenalan entitas kustom

`create-entity-recognizer` Contoh berikut memulai proses pelatihan untuk model pengenalan entitas kustom. Contoh ini menggunakan file CSV yang berisi dokumen pelatihan `raw_text.csv`, dan daftar entitas CSV, `entity_list.csv` untuk melatih model. `entity-list.csv` berisi kolom berikut: teks dan jenis.

```
aws comprehend create-entity-recognizer \
  --recognizer-name example-entity-recognizer
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
  --input-data-config "EntityTypes=[{Type=DEVICE}],Documents={S3Uri=s3://DOC-
EXAMPLE-BUCKET/trainingdata/raw_text.csv},EntityList={S3Uri=s3://DOC-EXAMPLE-BUCKET/
trainingdata/entity_list.csv}"
  --language-code en
```

**Output:**

```
{
  "EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:example-
entity-recognizer/entityrecognizer1"
}
```

Untuk informasi selengkapnya, lihat [Pengenalan entitas khusus](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [CreateEntityRecognizer](#) di Referensi AWS CLI Perintah.

## **create-flywheel**

Contoh kode berikut menunjukkan cara menggunakan `create-flywheel`.

### AWS CLI

Untuk membuat flywheel

`create-flywheel` Contoh berikut membuat flywheel untuk mengatur pelatihan berkelanjutan baik klasifikasi dokumen atau model pengenalan entitas. Flywheel dalam contoh ini dibuat untuk mengelola model terlatih yang sudah ada yang ditentukan oleh tag. `--active-model-arn` Saat flywheel dibuat, danau data dibuat di tag. `--input-data-lake`

```
aws comprehend create-flywheel \  
  --flywheel-name example-flywheel \  
  --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-model/version/1 \  
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role \  
  --data-lake-s3-uri "s3://DOC-EXAMPLE-BUCKET"
```

Output:

```
{  
  "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel"  
}
```

Untuk informasi selengkapnya, lihat [Ikhtisar Roda Gila di Panduan Pengembang](#) Amazon Comprehend.

- Untuk detail API, lihat [CreateFlywheel](#) di Referensi AWS CLI Perintah.

## **delete-document-classifier**

Contoh kode berikut menunjukkan cara menggunakan `delete-document-classifier`.

## AWS CLI

Untuk menghapus pengklasifikasi dokumen kustom

`delete-document-classifier` Contoh berikut menghapus model pengklasifikasi dokumen kustom.

```
aws comprehend delete-document-classifier \  
  --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DeleteDocumentClassifier](#) di Referensi AWS CLI Perintah.

## **delete-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `delete-endpoint`.

## AWS CLI

Untuk menghapus titik akhir untuk model kustom

`delete-endpoint` Contoh berikut menghapus titik akhir khusus model. Semua titik akhir harus dihapus agar model dihapus.

```
aws comprehend delete-endpoint \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DeleteEndpoint](#) di Referensi AWS CLI Perintah.

## **delete-entity-recognizer**

Contoh kode berikut menunjukkan cara menggunakan `delete-entity-recognizer`.



## AWS CLI

Untuk menghapus model pengenalan entitas kustom

`delete-entity-recognizer` Contoh berikut menghapus model pengenalan entitas kustom.

```
aws comprehend delete-entity-recognizer \  
  --entity-recognizer-arn arn:aws:comprehend:us-west-2:111122223333:entity-  
  recognizer/example-entity-recognizer-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DeleteEntityRecognizer](#) di Referensi AWS CLI Perintah.

## `delete-flywheel`

Contoh kode berikut menunjukkan cara menggunakan `delete-flywheel`.

## AWS CLI

Untuk menghapus flywheel

`delete-flywheel` Contoh berikut menghapus flywheel. Data lake atau model yang terkait dengan flywheel tidak dihapus.

```
aws comprehend delete-flywheel \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
  flywheel-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ikhtisar Flywheel di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DeleteFlywheel](#) di Referensi AWS CLI Perintah.

## `delete-resource-policy`

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-policy`.

## AWS CLI

Untuk menghapus kebijakan berbasis sumber daya

`delete-resource-policy` Contoh berikut menghapus kebijakan berbasis sumber daya dari sumber daya Amazon Comprehend.

```
aws comprehend delete-resource-policy \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier-1/version/1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyalin model kustom antar AWS akun di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DeleteResourcePolicy](#) di Referensi AWS CLI Perintah.

## **describe-dataset**

Contoh kode berikut menunjukkan cara menggunakan `describe-dataset`.

### AWS CLI

Untuk menggambarkan kumpulan data flywheel

`describe-dataset` Contoh berikut mendapatkan properti dari dataset flywheel.

```
aws comprehend describe-dataset \  
  --dataset-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset
```

Output:

```
{  
  "DatasetProperties": {  
    "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset",  
    "DatasetName": "example-dataset",  
    "DatasetType": "TRAIN",
```

```
"DatasetS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/
schemaVersion=1/12345678A123456Z/datasets/example-dataset/20230616T203710Z/",
  "Status": "CREATING",
  "CreationTime": "2023-06-16T20:37:10.400000+00:00"
}
}
```

Untuk informasi selengkapnya, lihat [Ikhtisar Roda Gila di Panduan Pengembang](#) Amazon Comprehend.

- Untuk detail API, lihat [DescribeDataset](#) di Referensi AWS CLI Perintah.

## describe-document-classification-job

Contoh kode berikut menunjukkan cara menggunakan `describe-document-classification-job`.

### AWS CLI

Untuk menggambarkan pekerjaan klasifikasi dokumen

`describe-document-classification-job` Contoh berikut mendapatkan properti pekerjaan klasifikasi dokumen asinkron.

```
aws comprehend describe-document-classification-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
  "DocumentClassificationJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classification-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "exampleclassificationjob",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-14T17:09:51.788000+00:00",
    "EndTime": "2023-06-14T17:15:58.582000+00:00",
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/mymodel/version/1",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/jobdata/",
```

```

        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-
CLN-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
    }
}

```

Untuk informasi selengkapnya, lihat [Klasifikasi Kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DescribeDocumentClassificationJob](#) di Referensi AWS CLI Perintah.

## describe-document-classifier

Contoh kode berikut menunjukkan cara menggunakan `describe-document-classifier`.

### AWS CLI

Untuk menggambarkan pengklasifikasi dokumen

`describe-document-classifier` Contoh berikut mendapatkan properti model pengklasifikasi dokumen kustom.

```

aws comprehend describe-document-classifier \
  --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier-1

```

Output:

```

{
  "DocumentClassifierProperties": {
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/example-classifier-1",
    "LanguageCode": "en",
    "Status": "TRAINED",
    "SubmitTime": "2023-06-13T19:04:15.735000+00:00",
    "EndTime": "2023-06-13T19:42:31.752000+00:00",
    "TrainingStartTime": "2023-06-13T19:08:20.114000+00:00",

```

```

    "TrainingEndTime": "2023-06-13T19:41:35.080000+00:00",
    "InputDataConfig": {
      "DataFormat": "COMPREHEND_CSV",
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata"
    },
    "OutputDataConfig": {},
    "ClassifierMetadata": {
      "NumberOfLabels": 3,
      "NumberOfTrainedDocuments": 5016,
      "NumberOfTestDocuments": 557,
      "EvaluationMetrics": {
        "Accuracy": 0.9856,
        "Precision": 0.9919,
        "Recall": 0.9459,
        "F1Score": 0.9673,
        "MicroPrecision": 0.9856,
        "MicroRecall": 0.9856,
        "MicroF1Score": 0.9856,
        "HammingLoss": 0.0144
      }
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role",
    "Mode": "MULTI_CLASS"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat dan mengelola model kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DescribeDocumentClassifier](#) di Referensi AWS CLI Perintah.

## describe-dominant-language-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-dominant-language-detection-job`.

### AWS CLI

Untuk menggambarkan pekerjaan deteksi bahasa yang dominan.

`describe-dominant-language-detection-job` Contoh berikut mendapatkan properti dari pekerjaan deteksi bahasa dominan asinkron.

```
aws comprehend describe-dominant-language-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "DominantLanguageDetectionJobProperties": {  
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-  
detection-job/123456abcdeb0e11022f22a11EXAMPLE",  
    "JobName": "languageanalysis1",  
    "JobStatus": "IN_PROGRESS",  
    "SubmitTime": "2023-06-09T18:10:38.037000+00:00",  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-  
LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"  
    },  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeDominantLanguageDetectionJob](#) di Referensi AWS CLI Perintah.

## describe-endpoint

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoint`.

### AWS CLI

Untuk menggambarkan titik akhir tertentu

`describe-endpoint` Contoh berikut mendapatkan properti dari endpoint model-spesifik.

```
aws comprehend describe-endpoint \  

```

```
--endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint
```

Output:

```
{  
  "EndpointProperties": {  
    "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier-endpoint/example-classifier-endpoint,  
    "Status": "IN_SERVICE",  
    "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
exampleclassifier1",  
    "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/exampleclassifier1",  
    "DesiredInferenceUnits": 1,  
    "CurrentInferenceUnits": 1,  
    "CreationTime": "2023-06-13T20:32:54.526000+00:00",  
    "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeEndpoint](#) di Referensi AWS CLI Perintah.

## describe-entities-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-entities-detection-job`.

AWS CLI

Untuk menggambarkan pekerjaan deteksi entitas

`describe-entities-detection-job` Contoh berikut mendapatkan properti dari pekerjaan deteksi entitas asinkron.

```
aws comprehend describe-entities-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
  "EntitiesDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "example-entity-detector",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
    "EndTime": "2023-06-08T21:40:23.509000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/thefolder/111122223333-
NER-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::12345678012:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeEntitiesDetectionJob](#) di Referensi AWS CLI Perintah.

## describe-entity-recognizer

Contoh kode berikut menunjukkan cara menggunakan `describe-entity-recognizer`.

### AWS CLI

Untuk menggambarkan pengenalan entitas

`describe-entity-recognizer` Contoh berikut mendapatkan properti model pengenalan entitas kustom.

```
aws comprehend describe-entity-recognizer \
  entity-recognizer-arn arn:aws:comprehend:us-west-2:111122223333:entity-
recognizer/business-recongizer-1/version/1
```



## Output:

```
{
  "EntityRecognizerProperties": {
    "EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/business-recongizer-1/version/1",
    "LanguageCode": "en",
    "Status": "TRAINED",
    "SubmitTime": "2023-06-14T20:44:59.631000+00:00",
    "EndTime": "2023-06-14T20:59:19.532000+00:00",
    "TrainingStartTime": "2023-06-14T20:48:52.811000+00:00",
    "TrainingEndTime": "2023-06-14T20:58:11.473000+00:00",
    "InputDataConfig": {
      "DataFormat": "COMPREHEND_CSV",
      "EntityTypes": [
        {
          "Type": "BUSINESS"
        }
      ],
      "Documents": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/dataset/",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "EntityList": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/entity.csv"
      }
    },
    "RecognizerMetadata": {
      "NumberOfTrainedDocuments": 1814,
      "NumberOfTestDocuments": 486,
      "EvaluationMetrics": {
        "Precision": 100.0,
        "Recall": 100.0,
        "F1Score": 100.0
      }
    },
    "EntityTypes": [
      {
        "Type": "BUSINESS",
        "EvaluationMetrics": {
          "Precision": 100.0,
          "Recall": 100.0,
          "F1Score": 100.0
        }
      }
    ],
    "NumberOfTrainMentions": 1520
  }
}
```

```

    }
  ],
  },
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
  "VersionName": "1"
}
}

```

Untuk informasi selengkapnya, lihat [Pengenalan entitas khusus](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DescribeEntityRecognizer](#) di Referensi AWS CLI Perintah.

## describe-events-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-events-detection-job`.

### AWS CLI

Untuk menggambarkan pekerjaan deteksi peristiwa.

`describe-events-detection-job` Contoh berikut mendapatkan properti pekerjaan deteksi peristiwa asinkron.

```
aws comprehend describe-events-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

### Output:

```

{
  "EventsDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "events_job_1",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-12T18:45:56.054000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/EventsData",
      "InputFormat": "ONE_DOC_PER_LINE"
    }
  },

```

```

    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-
EVENTS-123456abcdeb0e11022f22a11EXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "TargetEventTypes": [
      "BANKRUPTCY",
      "EMPLOYMENT",
      "CORPORATE_ACQUISITION",
      "CORPORATE_MERGER",
      "INVESTMENT_GENERAL"
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeEventsDetectionJob](#) di Referensi AWS CLI Perintah.

## describe-flywheel-iteration

Contoh kode berikut menunjukkan cara menggunakan `describe-flywheel-iteration`.

### AWS CLI

Untuk menggambarkan iterasi flywheel

`describe-flywheel-iteration` Contoh berikut mendapatkan properti dari iterasi flywheel.

```

aws comprehend describe-flywheel-iteration \
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-
flywheel \
  --flywheel-iteration-id 20232222AEXAMPLE

```

### Output:

```

{
  "FlywheelIterationProperties": {
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity",

```

```

    "FlywheelIterationId": "20232222AEXAMPLE",
    "CreationTime": "2023-06-16T21:10:26.385000+00:00",
    "EndTime": "2023-06-16T23:33:16.827000+00:00",
    "Status": "COMPLETED",
    "Message": "FULL_ITERATION: Flywheel iteration performed all functions
successfully.",
    "EvaluatedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier/version/1",
    "EvaluatedModelMetrics": {
      "AverageF1Score": 0.7742663922375772,
      "AveragePrecision": 0.8287636394041166,
      "AverageRecall": 0.7427084833645399,
      "AverageAccuracy": 0.8795394154118689
    },
    "TrainedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier/version/Comprehend-Generated-v1-bb52d585",
    "TrainedModelMetrics": {
      "AverageF1Score": 0.9767700253081214,
      "AveragePrecision": 0.9767700253081214,
      "AverageRecall": 0.9767700253081214,
      "AverageAccuracy": 0.9858281665190434
    },
    "EvaluationManifestS3Prefix": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/flywheel-
entity/schemaVersion=1/20230616T200543Z/evaluation/20230616T211026Z/"
  }
}

```

Untuk informasi selengkapnya, lihat [ikhtisar Flywheel di Panduan](#) Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DescribeFlywheelIteration](#) di Referensi AWS CLI Perintah.

## describe-flywheel

Contoh kode berikut menunjukkan cara menggunakan `describe-flywheel`.

### AWS CLI

Untuk menggambarkan flywheel

`describe-flywheel` Contoh berikut mendapatkan properti flywheel. Dalam contoh ini, model yang terkait dengan flywheel adalah model pengklasifikasi khusus yang dilatih untuk mengklasifikasikan dokumen sebagai spam atau nonspam, atau, “ham”.

```
aws comprehend describe-flywheel \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel
```

Output:

```
{  
  "FlywheelProperties": {  
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel",  
    "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-model/version/1",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role",  
    "TaskConfig": {  
      "LanguageCode": "en",  
      "DocumentClassificationConfig": {  
        "Mode": "MULTI_CLASS",  
        "Labels": [  
          "ham",  
          "spam"  
        ]  
      }  
    },  
    "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/example-flywheel/  
schemaVersion=1/20230616T200543Z/",  
    "DataSecurityConfig": {},  
    "Status": "ACTIVE",  
    "ModelType": "DOCUMENT_CLASSIFIER",  
    "CreationTime": "2023-06-16T20:05:43.242000+00:00",  
    "LastModifiedTime": "2023-06-16T20:21:43.567000+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Ikhtisar Roda Gila di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeFlywheel](#) di Referensi AWS CLI Perintah.

## describe-key-phrases-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-key-phrases-detection-job`.

## AWS CLI

Untuk menggambarkan pekerjaan deteksi frasa kunci

`describe-key-phrases-detection-job` Contoh berikut mendapatkan properti dari pekerjaan deteksi frase kunci asinkron.

```
aws comprehend describe-key-phrases-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "KeyPhrasesDetectionJobProperties": {  
    "JobId": "69aa080c00fc68934a6a98f10EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-  
job/69aa080c00fc68934a6a98f10EXAMPLE",  
    "JobName": "example-key-phrases-detection-job",  
    "JobStatus": "COMPLETED",  
    "SubmitTime": 1686606439.177,  
    "EndTime": 1686606806.157,  
    "InputDataConfig": {  
      "S3Uri": "s3://dereksbucket1001/EventsData/",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://dereksbucket1002/testfolder/111122223333-  
KP-69aa080c00fc68934a6a98f10EXAMPLE/output/output.tar.gz"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-testrole"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeKeyPhrasesDetectionJob](#) di Referensi AWS CLI Perintah.

## describe-pii-entities-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-pii-entities-detection-job`.

### AWS CLI

Untuk menggambarkan pekerjaan deteksi entitas PII

`describe-pii-entities-detection-job` Contoh berikut mendapatkan properti pekerjaan deteksi entitas pii asinkron.

```
aws comprehend describe-pii-entities-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

### Output:

```
{
  "PiiEntitiesDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "example-pii-entities-job",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
    "EndTime": "2023-06-08T21:40:23.509000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/thefolder/111122223333-
NER-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::12345678012:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribePiiEntitiesDetectionJob](#) di Referensi AWS CLI Perintah.

## describe-resource-policy

Contoh kode berikut menunjukkan cara menggunakan `describe-resource-policy`.

### AWS CLI

Untuk menggambarkan kebijakan sumber daya yang dilampirkan pada model

`describe-resource-policy` Contoh berikut mendapatkan properti kebijakan berbasis sumber daya yang dilampirkan ke model.

```
aws comprehend describe-resource-policy \
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1
```

### Output:

```
{
  "ResourcePolicy": "{ \"Version\": \"2012-10-17\", \"Statement\": [{ \"Effect\":
  \\\"Allow\\\", \"Principal\": { \"AWS\": \"arn:aws:iam::444455556666:root\" }, \"Action\":
  \\\"comprehend:ImportModel\\\", \"Resource\": \"*\" } ] }\",
  \"CreationTime\": \"2023-06-19T18:44:26.028000+00:00\",
  \"LastModifiedTime\": \"2023-06-19T18:53:02.002000+00:00\",
  \"PolicyRevisionId\": \"baa675d069d07afaa2aa3106ae280f61\"
}
```

Untuk informasi selengkapnya, lihat [Menyalin model kustom antar AWS akun di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeResourcePolicy](#) di Referensi AWS CLI Perintah.

## describe-sentiment-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-sentiment-detection-job`.

### AWS CLI

Untuk menggambarkan pekerjaan deteksi sentimen



`describe-sentiment-detection-job` Contoh berikut mendapatkan properti pekerjaan deteksi sentimen asinkron.

```
aws comprehend describe-sentiment-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
  "SentimentDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "movie_review_analysis",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam:111122223333:role/service-role/AmazonComprehendServiceRole-servicerole"
  }
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeSentimentDetectionJob](#) di Referensi AWS CLI Perintah.

## **describe-targeted-sentiment-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `describe-targeted-sentiment-detection-job`.

## AWS CLI

Untuk menggambarkan pekerjaan deteksi sentimen yang ditargetkan

`describe-targeted-sentiment-detection-job` Contoh berikut mendapatkan properti pekerjaan deteksi sentimen bertarget asinkron.

```
aws comprehend describe-targeted-sentiment-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "TargetedSentimentDetectionJobProperties": {  
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-  
detection-job/123456abcdeb0e11022f22a11EXAMPLE",  
    "JobName": "movie_review_analysis",  
    "JobStatus": "IN_PROGRESS",  
    "SubmitTime": "2023-06-09T23:16:15.956000+00:00",  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-  
TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-servicerole"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeTargetedSentimentDetectionJob](#) di Referensi AWS CLI Perintah.

## `describe-topics-detection-job`

Contoh kode berikut menunjukkan cara menggunakan `describe-topics-detection-job`.

## AWS CLI

Untuk menggambarkan pekerjaan deteksi topik

`describe-topics-detection-job` Contoh berikut mendapatkan properti pekerjaan deteksi topik asinkron.

```
aws comprehend describe-topics-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
  "TopicsDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "example_topics_detection",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T18:44:43.414000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-TOPICS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "NumberOfTopics": 10,
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-examplerole"
  }
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [DescribeTopicsDetectionJob](#) di Referensi AWS CLI Perintah.

## **detect-dominant-language**

Contoh kode berikut menunjukkan cara menggunakan `detect-dominant-language`.

## AWS CLI

Untuk mendeteksi bahasa dominan teks input

Berikut ini `detect-dominant-language` menganalisis teks input dan mengidentifikasi bahasa dominan. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output.

```
aws comprehend detect-dominant-language \  
  --text "It is a beautiful day in Seattle."
```

Output:

```
{  
  "Languages": [  
    {  
      "LanguageCode": "en",  
      "Score": 0.9877256155014038  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bahasa Dominan](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DetectDominantLanguage](#) di Referensi AWS CLI Perintah.

## **detect-entities**

Contoh kode berikut menunjukkan cara menggunakan `detect-entities`.

### AWS CLI

Untuk mendeteksi entitas bernama dalam teks input

`detect-entities` Contoh berikut menganalisis teks input dan mengembalikan entitas bernama. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```
aws comprehend detect-entities \  
  --language-code en \  
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC  
  credit card \  
"
```

```
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
31st. Based on your autopay settings, \
we will withdraw your payment on the due date from your bank account number
XXXXXX1111 with the routing number XXXXX0000. \
Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to
Alice at AnySpa@example.com."
```

## Output:

```
{
  "Entities": [
    {
      "Score": 0.9994556307792664,
      "Type": "PERSON",
      "Text": "Zhang Wei",
      "BeginOffset": 6,
      "EndOffset": 15
    },
    {
      "Score": 0.9981022477149963,
      "Type": "PERSON",
      "Text": "John",
      "BeginOffset": 22,
      "EndOffset": 26
    },
    {
      "Score": 0.9986887574195862,
      "Type": "ORGANIZATION",
      "Text": "AnyCompany Financial Services, LLC",
      "BeginOffset": 33,
      "EndOffset": 67
    },
    {
      "Score": 0.9959119558334351,
      "Type": "OTHER",
      "Text": "1111-XXXX-1111-XXXX",
      "BeginOffset": 88,
      "EndOffset": 107
    },
    {
      "Score": 0.9708039164543152,
      "Type": "QUANTITY",
      "Text": ".53",

```

```
    "BeginOffset": 133,  
    "EndOffset": 136  
  },  
  {  
    "Score": 0.9987268447875977,  
    "Type": "DATE",  
    "Text": "July 31st",  
    "BeginOffset": 152,  
    "EndOffset": 161  
  },  
  {  
    "Score": 0.9858865737915039,  
    "Type": "OTHER",  
    "Text": "XXXXXX1111",  
    "BeginOffset": 271,  
    "EndOffset": 281  
  },  
  {  
    "Score": 0.9700471758842468,  
    "Type": "OTHER",  
    "Text": "XXXXX0000",  
    "BeginOffset": 306,  
    "EndOffset": 315  
  },  
  {  
    "Score": 0.9591118693351746,  
    "Type": "ORGANIZATION",  
    "Text": "Sunshine Spa",  
    "BeginOffset": 340,  
    "EndOffset": 352  
  },  
  {  
    "Score": 0.9797496795654297,  
    "Type": "LOCATION",  
    "Text": "123 Main St",  
    "BeginOffset": 354,  
    "EndOffset": 365  
  },  
  {  
    "Score": 0.994929313659668,  
    "Type": "PERSON",  
    "Text": "Alice",  
    "BeginOffset": 394,  
    "EndOffset": 399  
  }
```

```

    },
    {
      "Score": 0.9949769377708435,
      "Type": "OTHER",
      "Text": "AnySpa@example.com",
      "BeginOffset": 403,
      "EndOffset": 418
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Entitas](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DetectEntities](#) di Referensi AWS CLI Perintah.

## detect-key-phrases

Contoh kode berikut menunjukkan cara menggunakan `detect-key-phrases`.

### AWS CLI

Untuk mendeteksi frasa kunci dalam teks input

`detect-key-phrases` Contoh berikut menganalisis teks input dan mengidentifikasi frasa kata benda kunci. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```

aws comprehend detect-key-phrases \
  --language-code en \
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
credit card \
  account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
July 31st. Based on your autopay settings, \
  we will withdraw your payment on the due date from your bank account number
XXXXXX1111 with the routing number XXXXX0000. \
  Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to
Alice at AnySpa@example.com."

```

Output:

```

{
  "KeyPhrases": [
    {

```

```
    "Score": 0.8996376395225525,  
    "Text": "Zhang Wei",  
    "BeginOffset": 6,  
    "EndOffset": 15  
  },  
  {  
    "Score": 0.9992469549179077,  
    "Text": "John",  
    "BeginOffset": 22,  
    "EndOffset": 26  
  },  
  {  
    "Score": 0.988385021686554,  
    "Text": "Your AnyCompany Financial Services",  
    "BeginOffset": 28,  
    "EndOffset": 62  
  },  
  {  
    "Score": 0.8740853071212769,  
    "Text": "LLC credit card account 1111-XXXX-1111-XXXX",  
    "BeginOffset": 64,  
    "EndOffset": 107  
  },  
  {  
    "Score": 0.9999437928199768,  
    "Text": "a minimum payment",  
    "BeginOffset": 112,  
    "EndOffset": 129  
  },  
  {  
    "Score": 0.9998900890350342,  
    "Text": ".53",  
    "BeginOffset": 133,  
    "EndOffset": 136  
  },  
  {  
    "Score": 0.9979453086853027,  
    "Text": "July 31st",  
    "BeginOffset": 152,  
    "EndOffset": 161  
  },  
  {  
    "Score": 0.9983011484146118,  
    "Text": "your autopay settings",
```



```
    "BeginOffset": 172,  
    "EndOffset": 193  
  },  
  {  
    "Score": 0.9996572136878967,  
    "Text": "your payment",  
    "BeginOffset": 211,  
    "EndOffset": 223  
  },  
  {  
    "Score": 0.9995037317276001,  
    "Text": "the due date",  
    "BeginOffset": 227,  
    "EndOffset": 239  
  },  
  {  
    "Score": 0.9702621698379517,  
    "Text": "your bank account number XXXXXX1111",  
    "BeginOffset": 245,  
    "EndOffset": 280  
  },  
  {  
    "Score": 0.9179925918579102,  
    "Text": "the routing number XXXXX0000.Customer feedback",  
    "BeginOffset": 286,  
    "EndOffset": 332  
  },  
  {  
    "Score": 0.9978160858154297,  
    "Text": "Sunshine Spa",  
    "BeginOffset": 337,  
    "EndOffset": 349  
  },  
  {  
    "Score": 0.9706913232803345,  
    "Text": "123 Main St",  
    "BeginOffset": 351,  
    "EndOffset": 362  
  },  
  {  
    "Score": 0.9941995143890381,  
    "Text": "comments",  
    "BeginOffset": 379,  
    "EndOffset": 387  
  }
```

```

    },
    {
      "Score": 0.9759287238121033,
      "Text": "Alice",
      "BeginOffset": 391,
      "EndOffset": 396
    },
    {
      "Score": 0.8376792669296265,
      "Text": "AnySpa@example.com",
      "BeginOffset": 400,
      "EndOffset": 415
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Frasa Kunci](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DetectKeyPhrases](#) di Referensi AWS CLI Perintah.

## detect-pii-entities

Contoh kode berikut menunjukkan cara menggunakan `detect-pii-entities`.

### AWS CLI

Untuk mendeteksi entitas pii dalam teks input

`detect-pii-entities` Contoh berikut menganalisis teks input dan mengidentifikasi entitas yang berisi informasi identitas pribadi (PII). Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```

aws comprehend detect-pii-entities \
  --language-code en \
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
credit card \
  account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
July 31st. Based on your autopay settings, \
  we will withdraw your payment on the due date from your bank account number
XXXXXX1111 with the routing number XXXXX0000. \
  Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to
Alice at AnySpa@example.com."

```

## Output:

```
{
  "Entities": [
    {
      "Score": 0.9998322129249573,
      "Type": "NAME",
      "BeginOffset": 6,
      "EndOffset": 15
    },
    {
      "Score": 0.9998878240585327,
      "Type": "NAME",
      "BeginOffset": 22,
      "EndOffset": 26
    },
    {
      "Score": 0.9994089603424072,
      "Type": "CREDIT_DEBIT_NUMBER",
      "BeginOffset": 88,
      "EndOffset": 107
    },
    {
      "Score": 0.9999760985374451,
      "Type": "DATE_TIME",
      "BeginOffset": 152,
      "EndOffset": 161
    },
    {
      "Score": 0.9999449253082275,
      "Type": "BANK_ACCOUNT_NUMBER",
      "BeginOffset": 271,
      "EndOffset": 281
    },
    {
      "Score": 0.9999847412109375,
      "Type": "BANK_ROUTING",
      "BeginOffset": 306,
      "EndOffset": 315
    },
    {
      "Score": 0.999925434589386,
      "Type": "ADDRESS",
      "BeginOffset": 354,
```

```

        "EndOffset": 365
      },
      {
        "Score": 0.9989161491394043,
        "Type": "NAME",
        "BeginOffset": 394,
        "EndOffset": 399
      },
      {
        "Score": 0.9994171857833862,
        "Type": "EMAIL",
        "BeginOffset": 403,
        "EndOffset": 418
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Informasi Identifikasi Pribadi \(PII\)](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DetectPiiEntities](#) di Referensi AWS CLI Perintah.

## detect-sentiment

Contoh kode berikut menunjukkan cara menggunakan `detect-sentiment`.

### AWS CLI

Untuk mendeteksi sentimen teks input

`detect-sentiment` Contoh berikut menganalisis teks masukan dan mengembalikan inferensi sentimen yang berlaku (POSITIVE, NEUTRAL, MIXED atau) NEGATIVE

```

aws comprehend detect-sentiment \
  --language-code en \
  --text "It is a beautiful day in Seattle"

```

Output:

```

{
  "Sentiment": "POSITIVE",
  "SentimentScore": {
    "Positive": 0.9976957440376282,

```

```
    "Negative": 9.653854067437351e-05,  
    "Neutral": 0.002169104292988777,  
    "Mixed": 3.857641786453314e-05  
  }  
}
```

Untuk informasi selengkapnya, lihat [Sentimen](#) di Panduan Pengembang Amazon Comprehend

- Untuk detail API, lihat [DetectSentiment](#) di Referensi AWS CLI Perintah.

## detect-syntax

Contoh kode berikut menunjukkan cara menggunakan `detect-syntax`.

### AWS CLI

Untuk mendeteksi bagian-bagian ucapan dalam teks input

`detect-syntax` Contoh berikut menganalisis sintaks teks masukan dan mengembalikan bagian-bagian yang berbeda dari pidato. Skor kepercayaan model yang telah dilatih sebelumnya juga merupakan output untuk setiap prediksi.

```
aws comprehend detect-syntax \  
  --language-code en \  
  --text "It is a beautiful day in Seattle."
```

Output:

```
{  
  "SyntaxTokens": [  
    {  
      "TokenId": 1,  
      "Text": "It",  
      "BeginOffset": 0,  
      "EndOffset": 2,  
      "PartOfSpeech": {  
        "Tag": "PRON",  
        "Score": 0.9999740719795227  
      }  
    },  
    {  
      "TokenId": 2,  
      "Text": "is",
```

```
    "BeginOffset": 3,  
    "EndOffset": 5,  
    "PartOfSpeech": {  
      "Tag": "VERB",  
      "Score": 0.999901294708252  
    }  
  },  
  {  
    "TokenId": 3,  
    "Text": "a",  
    "BeginOffset": 6,  
    "EndOffset": 7,  
    "PartOfSpeech": {  
      "Tag": "DET",  
      "Score": 0.9999938607215881  
    }  
  },  
  {  
    "TokenId": 4,  
    "Text": "beautiful",  
    "BeginOffset": 8,  
    "EndOffset": 17,  
    "PartOfSpeech": {  
      "Tag": "ADJ",  
      "Score": 0.9987351894378662  
    }  
  },  
  {  
    "TokenId": 5,  
    "Text": "day",  
    "BeginOffset": 18,  
    "EndOffset": 21,  
    "PartOfSpeech": {  
      "Tag": "NOUN",  
      "Score": 0.9999796748161316  
    }  
  },  
  {  
    "TokenId": 6,  
    "Text": "in",  
    "BeginOffset": 22,  
    "EndOffset": 24,  
    "PartOfSpeech": {  
      "Tag": "ADP",
```

```

        "Score": 0.9998047947883606
      }
    },
    {
      "TokenId": 7,
      "Text": "Seattle",
      "BeginOffset": 25,
      "EndOffset": 32,
      "PartOfSpeech": {
        "Tag": "PROPN",
        "Score": 0.9940530061721802
      }
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Analisis Sintaks](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DetectSyntax](#) di Referensi AWS CLI Perintah.

## detect-targeted-sentiment

Contoh kode berikut menunjukkan cara menggunakan `detect-targeted-sentiment`.

### AWS CLI

Untuk mendeteksi sentimen yang ditargetkan dari entitas bernama dalam teks input

`detect-targeted-sentiment` Contoh berikut menganalisis teks masukan dan mengembalikan entitas bernama selain sentimen yang ditargetkan terkait dengan masing-masing entitas. Skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi juga merupakan output.

```

aws comprehend detect-targeted-sentiment \
  --language-code en \
  --text "I do not enjoy January because it is too cold but August is the perfect
  temperature"

```

Output:

```

{
  "Entities": [
    {

```

```
    "DescriptiveMentionIndex": [
      0
    ],
    "Mentions": [
      {
        "Score": 0.9999979734420776,
        "GroupScore": 1.0,
        "Text": "I",
        "Type": "PERSON",
        "MentionSentiment": {
          "Sentiment": "NEUTRAL",
          "SentimentScore": {
            "Positive": 0.0,
            "Negative": 0.0,
            "Neutral": 1.0,
            "Mixed": 0.0
          }
        },
        "BeginOffset": 0,
        "EndOffset": 1
      }
    ]
  },
  {
    "DescriptiveMentionIndex": [
      0
    ],
    "Mentions": [
      {
        "Score": 0.9638869762420654,
        "GroupScore": 1.0,
        "Text": "January",
        "Type": "DATE",
        "MentionSentiment": {
          "Sentiment": "NEGATIVE",
          "SentimentScore": {
            "Positive": 0.0031610000878572464,
            "Negative": 0.9967250227928162,
            "Neutral": 0.00011100000119768083,
            "Mixed": 1.9999999949504854e-06
          }
        },
        "BeginOffset": 15,
        "EndOffset": 22
      }
    ]
  }
}
```



```
    }
  ]
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      {
        "Score": 0.9664419889450073,
        "GroupScore": 1.0,
        "Text": "August",
        "Type": "DATE",
        "MentionSentiment": {
          "Sentiment": "POSITIVE",
          "SentimentScore": {
            "Positive": 0.9999549984931946,
            "Negative": 3.999999989900971e-06,
            "Neutral": 4.099999932805076e-05,
            "Mixed": 0.0
          }
        }
      },
      "BeginOffset": 50,
      "EndOffset": 56
    }
  ]
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "Score": 0.9803199768066406,
      "GroupScore": 1.0,
      "Text": "temperature",
      "Type": "ATTRIBUTE",
      "MentionSentiment": {
        "Sentiment": "POSITIVE",
        "SentimentScore": {
          "Positive": 1.0,
          "Negative": 0.0,
          "Neutral": 0.0,

```

```

        "Mixed": 0.0
      }
    },
    "BeginOffset": 77,
    "EndOffset": 88
  }
]
}

```

Untuk informasi selengkapnya, lihat [Sentimen Bertarget](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [DetectTargetedSentiment](#) di Referensi AWS CLI Perintah.

## import-model

Contoh kode berikut menunjukkan cara menggunakan `import-model`.

### AWS CLI

Untuk mengimpor model

`import-model` Contoh berikut mengimpor model dari AWS akun yang berbeda. Model pengklasifikasi dokumen dalam akun 444455556666 memiliki kebijakan berbasis sumber daya yang memungkinkan akun 111122223333 untuk mengimpor model.

```

aws comprehend import-model \
  --source-model-arn arn:aws:comprehend:us-west-2:444455556666:document-
  classifier/example-classifier

```

Output:

```

{
  "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
  example-classifier"
}

```

Untuk informasi selengkapnya, lihat [Menyalin model kustom antar AWS akun di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ImportModel](#) di Referensi AWS CLI Perintah.

## list-datasets

Contoh kode berikut menunjukkan cara menggunakan `list-datasets`.

### AWS CLI

Untuk membuat daftar semua dataset flywheel

`list-datasets` Contoh berikut mencantumkan semua dataset yang terkait dengan flywheel.

```
aws comprehend list-datasets \
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
  entity
```

Output:

```
{
  "DatasetPropertiesList": [
    {
      "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
flywheel-entity/dataset/example-dataset-1",
      "DatasetName": "example-dataset-1",
      "DatasetType": "TRAIN",
      "DatasetS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/
schemaVersion=1/20230616T200543Z/datasets/example-dataset-1/20230616T203710Z/",
      "Status": "CREATING",
      "CreationTime": "2023-06-16T20:37:10.400000+00:00"
    },
    {
      "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
flywheel-entity/dataset/example-dataset-2",
      "DatasetName": "example-dataset-2",
      "DatasetType": "TRAIN",
      "DatasetS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/
schemaVersion=1/20230616T200543Z/datasets/example-dataset-2/20230616T200607Z/",
      "Description": "TRAIN Dataset created by Flywheel creation.",
      "Status": "COMPLETED",
      "NumberOfDocuments": 5572,
      "CreationTime": "2023-06-16T20:06:07.722000+00:00"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Ikhtisar Roda Gila di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListDatasets](#) di Referensi AWS CLI Perintah.

## list-document-classification-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-document-classification-jobs`.

### AWS CLI

Untuk daftar semua pekerjaan klasifikasi dokumen

`list-document-classification-jobs` Contoh berikut mencantumkan semua pekerjaan klasifikasi dokumen.

```
aws comprehend list-document-classification-jobs
```

### Output:

```
{
  "DocumentClassificationJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:1234567890101:document-
classification-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "exampleclassificationjob",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-14T17:09:51.788000+00:00",
      "EndTime": "2023-06-14T17:15:58.582000+00:00",
      "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:1234567890101:document-classifier/mymodel/version/12",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/jobdata/",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/1234567890101-CLN-e758dd56b824aa717ceab551f11749fb/output/output.tar.gz"
      }
    }
  ]
}
```

```

    },
    "DataAccessRoleArn": "arn:aws:iam::1234567890101:role/service-role/
AmazonComprehendServiceRole-example-role"
  },
  {
    "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
    "JobArn": "arn:aws:comprehend:us-west-2:1234567890101:document-
classification-job/123456abcdeb0e11022f22a1EXAMPLE2",
    "JobName": "exampleclassificationjob2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-14T17:22:39.829000+00:00",
    "EndTime": "2023-06-14T17:28:46.107000+00:00",
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:1234567890101:document-classifier/mymodel/version/12",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/jobdata/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/1234567890101-CLN-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::1234567890101:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Klasifikasi Kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListDocumentClassificationJobs](#) di Referensi AWS CLI Perintah.

## **list-document-classifier-summaries**

Contoh kode berikut menunjukkan cara menggunakan `list-document-classifier-summaries`.

### AWS CLI

Untuk membuat daftar ringkasan dari semua pengklasifikasi dokumen yang dibuat

`list-document-classifier-summaries` Contoh berikut mencantumkan semua ringkasan pengklasifikasi dokumen yang dibuat.

```
aws comprehend list-document-classifier-summaries
```

Output:

```
{
  "DocumentClassifierSummariesList": [
    {
      "DocumentClassifierName": "example-classifier-1",
      "NumberOfVersions": 1,
      "LatestVersionCreatedAt": "2023-06-13T22:07:59.825000+00:00",
      "LatestVersionName": "1",
      "LatestVersionStatus": "TRAINED"
    },
    {
      "DocumentClassifierName": "example-classifier-2",
      "NumberOfVersions": 2,
      "LatestVersionCreatedAt": "2023-06-13T21:54:59.589000+00:00",
      "LatestVersionName": "2",
      "LatestVersionStatus": "TRAINED"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat dan mengelola model kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListDocumentClassifierSummaries](#) di Referensi AWS CLI Perintah.

## list-document-classifiers

Contoh kode berikut menunjukkan cara menggunakan `list-document-classifiers`.

AWS CLI

Untuk daftar semua pengklasifikasi dokumen

`list-document-classifiers` Contoh berikut mencantumkan semua model pengklasifikasi dokumen terlatih dan dalam pelatihan.

```
aws comprehend list-document-classifiers
```

## Output:

```
{
  "DocumentClassifierPropertiesList": [
    {
      "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/exampleclassifier1",
      "LanguageCode": "en",
      "Status": "TRAINED",
      "SubmitTime": "2023-06-13T19:04:15.735000+00:00",
      "EndTime": "2023-06-13T19:42:31.752000+00:00",
      "TrainingStartTime": "2023-06-13T19:08:20.114000+00:00",
      "TrainingEndTime": "2023-06-13T19:41:35.080000+00:00",
      "InputDataConfig": {
        "DataFormat": "COMPREHEND_CSV",
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata"
      },
      "OutputDataConfig": {},
      "ClassifierMetadata": {
        "NumberOfLabels": 3,
        "NumberOfTrainedDocuments": 5016,
        "NumberOfTestDocuments": 557,
        "EvaluationMetrics": {
          "Accuracy": 0.9856,
          "Precision": 0.9919,
          "Recall": 0.9459,
          "F1Score": 0.9673,
          "MicroPrecision": 0.9856,
          "MicroRecall": 0.9856,
          "MicroF1Score": 0.9856,
          "HammingLoss": 0.0144
        }
      },
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testorle",
      "Mode": "MULTI_CLASS"
    },
    {
      "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/exampleclassifier2",
      "LanguageCode": "en",
      "Status": "TRAINING",
      "SubmitTime": "2023-06-13T21:20:28.690000+00:00",
      "InputDataConfig": {
```

```

        "DataFormat": "COMPREHEND_CSV",
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata"
    },
    "OutputDataConfig": {},
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testorle",
    "Mode": "MULTI_CLASS"
}
]
}

```

Untuk informasi selengkapnya, lihat [Membuat dan mengelola model kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListDocumentClassifiers](#) di Referensi AWS CLI Perintah.

## list-dominant-language-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-dominant-language-detection-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan deteksi bahasa yang dominan

`list-dominant-language-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi bahasa dominan asinkron yang sedang berlangsung dan diselesaikan.

```
aws comprehend list-dominant-language-detection-jobs
```

### Output:

```

{
  "DominantLanguageDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "languageanalysis1",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T18:10:38.037000+00:00",
      "EndTime": "2023-06-09T18:18:45.498000+00:00",
    }
  ]
}

```



```

    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/
output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  },
  {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "languageanalysis2",
    "JobStatus": "STOPPED",
    "SubmitTime": "2023-06-09T18:16:33.690000+00:00",
    "EndTime": "2023-06-09T18:24:40.608000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/
output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListDominantLanguageDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-endpoints

Contoh kode berikut menunjukkan cara menggunakan `list-endpoints`.

## AWS CLI

Untuk daftar semua titik akhir

`list-endpoints` Contoh berikut mencantumkan semua titik akhir khusus model aktif.

```
aws comprehend list-endpoints
```

Output:

```
{
  "EndpointPropertiesList": [
    {
      "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/ExampleClassifierEndpoint",
      "Status": "IN_SERVICE",
      "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
      "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
      "DesiredInferenceUnits": 1,
      "CurrentInferenceUnits": 1,
      "CreationTime": "2023-06-13T20:32:54.526000+00:00",
      "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
    },
    {
      "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/ExampleClassifierEndpoint2",
      "Status": "IN_SERVICE",
      "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2",
      "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2",
      "DesiredInferenceUnits": 1,
      "CurrentInferenceUnits": 1,
      "CreationTime": "2023-06-13T20:32:54.526000+00:00",
      "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListEndpoints](#) di Referensi AWS CLI Perintah.

## list-entities-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-entities-detection-jobs`.

### AWS CLI

Untuk mencantumkan semua pekerjaan deteksi entitas

`list-entities-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi entitas asinkron.

```
aws comprehend list-entities-detection-jobs
```

Output:

```
{
  "EntitiesDetectionJobPropertiesList": [
    {
      "JobId": "468af39c28ab45b83eb0c4ab9EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-job/468af39c28ab45b83eb0c4ab9EXAMPLE",
      "JobName": "example-entities-detection",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-08T20:57:46.476000+00:00",
      "EndTime": "2023-06-08T21:05:53.718000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/thefolder/111122223333-NER-468af39c28ab45b83eb0c4ab9EXAMPLE/output/output.tar.gz"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role"
    },
    {
      "JobId": "809691caeaab0e71406f80a28EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-job/809691caeaab0e71406f80a28EXAMPLE",

```

```

    "JobName": "example-entities-detection-2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
    "EndTime": "2023-06-08T21:40:23.509000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-NER-809691caeaab0e71406f80a28EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  },
  {
    "JobId": "e00597c36b448b91d70dea165EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/e00597c36b448b91d70dea165EXAMPLE",
    "JobName": "example-entities-detection-3",
    "JobStatus": "STOPPED",
    "SubmitTime": "2023-06-08T22:19:28.528000+00:00",
    "EndTime": "2023-06-08T22:27:33.991000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-NER-e00597c36b448b91d70dea165EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Entitas](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListEntitiesDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-entity-recognizer-summaries

Contoh kode berikut menunjukkan cara menggunakan `list-entity-recognizer-summaries`.

### AWS CLI

Untuk daftar ringkasan untuk semua pengenalan entitas yang dibuat

`list-entity-recognizer-summaries` Contoh berikut mencantumkan semua ringkasan pengenalan entitas.

```
aws comprehend list-entity-recognizer-summaries
```

Output:

```
{
  "EntityRecognizerSummariesList": [
    {
      "RecognizerName": "entity-recognizer-3",
      "NumberOfVersions": 2,
      "LatestVersionCreatedAt": "2023-06-15T23:15:07.621000+00:00",
      "LatestVersionName": "2",
      "LatestVersionStatus": "STOP_REQUESTED"
    },
    {
      "RecognizerName": "entity-recognizer-2",
      "NumberOfVersions": 1,
      "LatestVersionCreatedAt": "2023-06-14T22:55:27.805000+00:00",
      "LatestVersionName": "2",
      "LatestVersionStatus": "TRAINED"
    },
    {
      "RecognizerName": "entity-recognizer-1",
      "NumberOfVersions": 1,
      "LatestVersionCreatedAt": "2023-06-14T20:44:59.631000+00:00",
      "LatestVersionName": "1",
      "LatestVersionStatus": "TRAINED"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Pengenalan entitas khusus](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListEntityRecognizerSummaries](#) di Referensi AWS CLI Perintah.

## list-entity-recognizers

Contoh kode berikut menunjukkan cara menggunakan `list-entity-recognizers`.

### AWS CLI

Untuk daftar semua pengenalan entitas kustom

`list-entity-recognizers` Contoh berikut mencantumkan semua pengenalan entitas kustom yang dibuat.

```
aws comprehend list-entity-recognizers
```

Output:

```
{
  "EntityRecognizerPropertiesList": [
    {
      "EntityRecognizerArn": "arn:aws:comprehend:us-
west-2:111122223333:entity-recognizer/EntityRecognizer/version/1",
      "LanguageCode": "en",
      "Status": "TRAINED",
      "SubmitTime": "2023-06-14T20:44:59.631000+00:00",
      "EndTime": "2023-06-14T20:59:19.532000+00:00",
      "TrainingStartTime": "2023-06-14T20:48:52.811000+00:00",
      "TrainingEndTime": "2023-06-14T20:58:11.473000+00:00",
      "InputDataConfig": {
        "DataFormat": "COMPREHEND_CSV",
        "EntityTypes": [
          {
            "Type": "BUSINESS"
          }
        ],
        "Documents": {
          "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/dataset/",
          "InputFormat": "ONE_DOC_PER_LINE"
        },
        "EntityList": {
          "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/entity.csv"
        }
      },
    },
  ],
}
```

```
"RecognizerMetadata": {
  "NumberOfTrainedDocuments": 1814,
  "NumberOfTestDocuments": 486,
  "EvaluationMetrics": {
    "Precision": 100.0,
    "Recall": 100.0,
    "F1Score": 100.0
  },
  "EntityTypes": [
    {
      "Type": "BUSINESS",
      "EvaluationMetrics": {
        "Precision": 100.0,
        "Recall": 100.0,
        "F1Score": 100.0
      },
      "NumberOfTrainMentions": 1520
    }
  ]
},
"DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-servicerole",
"VersionName": "1"
},
{
  "EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/entityrecognizer3",
  "LanguageCode": "en",
  "Status": "TRAINED",
  "SubmitTime": "2023-06-14T22:57:51.056000+00:00",
  "EndTime": "2023-06-14T23:14:13.894000+00:00",
  "TrainingStartTime": "2023-06-14T23:01:33.984000+00:00",
  "TrainingEndTime": "2023-06-14T23:13:02.984000+00:00",
  "InputDataConfig": {
    "DataFormat": "COMPREHEND_CSV",
    "EntityTypes": [
      {
        "Type": "DEVICE"
      }
    ],
    "Documents": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/raw_txt.csv",
      "InputFormat": "ONE_DOC_PER_LINE"
    }
  },
}
```

```

    "EntityList": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/entity_list.csv"
    }
  },
  "RecognizerMetadata": {
    "NumberOfTrainedDocuments": 4616,
    "NumberOfTestDocuments": 3489,
    "EvaluationMetrics": {
      "Precision": 98.54227405247813,
      "Recall": 100.0,
      "F1Score": 99.26578560939794
    }
  },
  "EntityTypes": [
    {
      "Type": "DEVICE",
      "EvaluationMetrics": {
        "Precision": 98.54227405247813,
        "Recall": 100.0,
        "F1Score": 99.26578560939794
      }
    },
    {
      "NumberOfTrainMentions": 2764
    }
  ]
},
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-servicerole"
}
]
}

```

Untuk informasi selengkapnya, lihat [Pengenalan entitas khusus](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListEntityRecognizers](#) di Referensi AWS CLI Perintah.

## list-events-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-events-detection-jobs`.

### AWS CLI

Untuk mencantumkan semua pekerjaan deteksi peristiwa



`list-events-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi peristiwa asinkron.

```
aws comprehend list-events-detection-jobs
```

Output:

```
{
  "EventsDetectionJobPropertiesList": [
    {
      "JobId": "aa9593f9203e84f3ef032ce18EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:1111222233333:events-detection-job/aa9593f9203e84f3ef032ce18EXAMPLE",
      "JobName": "events_job_1",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-12T19:14:57.751000+00:00",
      "EndTime": "2023-06-12T19:21:04.962000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/EventsData/",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/1111222233333-EVENTS-aa9593f9203e84f3ef032ce18EXAMPLE/output/"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::1111222233333:role/service-role/AmazonComprehendServiceRole-example-role",
      "TargetEventTypes": [
        "BANKRUPTCY",
        "EMPLOYMENT",
        "CORPORATE_ACQUISITION",
        "CORPORATE_MERGER",
        "INVESTMENT_GENERAL"
      ]
    },
    {
      "JobId": "4a990a2f7e82adfca6e171135EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:1111222233333:events-detection-job/4a990a2f7e82adfca6e171135EXAMPLE",
      "JobName": "events_job_2",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-12T19:55:43.702000+00:00",
```

```

    "EndTime": "2023-06-12T20:03:49.893000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/EventsData/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/1111222233333-EVENTS-4a990a2f7e82adfca6e171135EXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::1111222233333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "TargetEventTypes": [
      "BANKRUPTCY",
      "EMPLOYMENT",
      "CORPORATE_ACQUISITION",
      "CORPORATE_MERGER",
      "INVESTMENT_GENERAL"
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembangan Amazon Comprehend](#).

- Untuk detail API, lihat [ListEventsDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-flywheel-iteration-history

Contoh kode berikut menunjukkan cara menggunakan `list-flywheel-iteration-history`.

### AWS CLI

Untuk mencantumkan semua riwayat iterasi flywheel

`list-flywheel-iteration-history` Contoh berikut mencantumkan semua iterasi flywheel.

```

aws comprehend list-flywheel-iteration-history
  --flywheel-arn arn:aws:comprehend:us-west-2:1111222233333:flywheel/example-
flywheel

```

Output:

```
{
  "FlywheelIterationPropertiesList": [
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel",
      "FlywheelIterationId": "20230619EXAMPLE",
      "CreationTime": "2023-06-19T04:00:32.594000+00:00",
      "EndTime": "2023-06-19T04:00:49.248000+00:00",
      "Status": "COMPLETED",
      "Message": "FULL_ITERATION: Flywheel iteration performed all functions
successfully.",
      "EvaluatedModelArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/example-classifier/version/1",
      "EvaluatedModelMetrics": {
        "AverageF1Score": 0.7742663922375772,
        "AverageF1Score": 0.9876464664646313,
        "AveragePrecision": 0.9800000253081214,
        "AverageRecall": 0.9445600253081214,
        "AverageAccuracy": 0.9997281665190434
      },
      "EvaluationManifestS3Prefix": "s3://DOC-EXAMPLE-BUCKET/example-flywheel/
schemaVersion=1/20230619EXAMPLE/evaluation/20230619EXAMPLE/"
    },
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel-2",
      "FlywheelIterationId": "20230616EXAMPLE",
      "CreationTime": "2023-06-16T21:10:26.385000+00:00",
      "EndTime": "2023-06-16T23:33:16.827000+00:00",
      "Status": "COMPLETED",
      "Message": "FULL_ITERATION: Flywheel iteration performed all functions
successfully.",
      "EvaluatedModelArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/spamvshamclassify/version/1",
      "EvaluatedModelMetrics": {
        "AverageF1Score": 0.7742663922375772,
        "AverageF1Score": 0.9767700253081214,
        "AveragePrecision": 0.9767700253081214,
        "AverageRecall": 0.9767700253081214,
        "AverageAccuracy": 0.9858281665190434
      },
      "EvaluationManifestS3Prefix": "s3://DOC-EXAMPLE-BUCKET/example-
flywheel-2/schemaVersion=1/20230616EXAMPLE/evaluation/20230616EXAMPLE/"
    }
  ]
}
```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [ikhtisar Flywheel di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListFlywheelIterationHistory](#) di Referensi AWS CLI Perintah.

## list-flywheels

Contoh kode berikut menunjukkan cara menggunakan `list-flywheels`.

### AWS CLI

Untuk daftar semua flywheels

`list-flywheels` Contoh berikut mencantumkan semua flywheels yang dibuat.

```
aws comprehend list-flywheels
```

Output:

```

{
  "FlywheelSummaryList": [
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-1",
      "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/exampleclassifier/version/1",
      "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/example-flywheel-1/schemaVersion=1/20230616T200543Z/",
      "Status": "ACTIVE",
      "ModelType": "DOCUMENT_CLASSIFIER",
      "CreationTime": "2023-06-16T20:05:43.242000+00:00",
      "LastModifiedTime": "2023-06-19T04:00:43.027000+00:00",
      "LatestFlywheelIteration": "20230619T040032Z"
    },
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-2",
      "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/exampleclassifier2/version/1",

```

```

        "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/example-flywheel-2/
schemaVersion=1/20220616T200543Z/",
        "Status": "ACTIVE",
        "ModelType": "DOCUMENT_CLASSIFIER",
        "CreationTime": "2022-06-16T20:05:43.242000+00:00",
        "LastModifiedTime": "2022-06-19T04:00:43.027000+00:00",
        "LatestFlywheelIteration": "20220619T040032Z"
    }
]
}

```

Untuk informasi selengkapnya, lihat [ikhtisar Flywheel di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListFlywheels](#) di Referensi AWS CLI Perintah.

## list-key-phrases-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-key-phrases-detection-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan deteksi frasa kunci

`list-key-phrases-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi frasa kunci asinkron yang sedang berlangsung dan diselesaikan.

```
aws comprehend list-key-phrases-detection-jobs
```

Output:

```

{
  "KeyPhrasesDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "keyphrasesanalysis1",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-08T22:31:43.767000+00:00",
      "EndTime": "2023-06-08T22:39:52.565000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/AsyncBatchJobs/"
      }
    }
  ]
}

```

```
        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-KP-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
},
{
    "JobId": "123456abcdeb0e11022f22a33EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a33EXAMPLE",
    "JobName": "keyphrasesanalysis2",
    "JobStatus": "STOPPED",
    "SubmitTime": "2023-06-08T22:57:52.154000+00:00",
    "EndTime": "2023-06-08T23:05:48.385000+00:00",
    "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-KP-123456abcdeb0e11022f22a33EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
},
{
    "JobId": "123456abcdeb0e11022f22a44EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a44EXAMPLE",
    "JobName": "keyphrasesanalysis3",
    "JobStatus": "FAILED",
    "Message": "NO_READ_ACCESS_TO_INPUT: The provided data access role does
not have proper access to the input data.",
    "SubmitTime": "2023-06-09T16:47:04.029000+00:00",
    "EndTime": "2023-06-09T16:47:18.413000+00:00",
    "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
    }
},
```

```

      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-KP-123456abcdeb0e11022f22a44EXAMPLE/output/output.tar.gz"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListKeyPhrasesDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-pii-entities-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-pii-entities-detection-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan deteksi entitas pii

`list-pii-entities-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi pii asinkron yang sedang berlangsung dan diselesaikan.

```
aws comprehend list-pii-entities-detection-jobs
```

### Output:

```

{
  "PiiEntitiesDetectionJobPropertiesList": [
    {
      "JobId": "6f9db0c42d0c810e814670ee4EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/6f9db0c42d0c810e814670ee4EXAMPLE",
      "JobName": "example-pii-detection-job",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T21:02:46.241000+00:00",
      "EndTime": "2023-06-09T21:12:52.602000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",

```

```

        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/111122223333-
PII-6f9db0c42d0c810e814670ee4EXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "Mode": "ONLY_OFFSETS"
},
{
    "JobId": "d927562638cfa739331a99b3cEXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/d927562638cfa739331a99b3cEXAMPLE",
    "JobName": "example-pii-detection-job-2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-09T21:20:58.211000+00:00",
    "EndTime": "2023-06-09T21:31:06.027000+00:00",
    "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-PII-d927562638cfa739331a99b3cEXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "Mode": "ONLY_OFFSETS"
}
]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListPiiEntitiesDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-sentiment-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-sentiment-detection-jobs`.



## AWS CLI

Untuk membuat daftar semua pekerjaan deteksi sentimen

`list-sentiment-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi sentimen asinkron yang sedang berlangsung dan diselesaikan.

```
aws comprehend list-sentiment-detection-jobs
```

Output:

```
{
  "SentimentDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "example-sentiment-detection-job",
      "JobStatus": "IN_PROGRESS",
      "SubmitTime": "2023-06-09T22:42:20.545000+00:00",
      "EndTime": "2023-06-09T22:52:27.416000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    },
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE2",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE2",
      "JobName": "example-sentiment-detection-job-2",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
      "EndTime": "2023-06-09T23:26:00.168000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData2",
```

```

        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListSentimentDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk sumber daya Amazon Comprehend.

```

aws comprehend list-tags-for-resource \
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1

```

Output:

```

{
  "ResourceArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1",
  "Tags": [
    {
      "Key": "Department",
      "Value": "Finance"
    }
  ],
}

```

```
{
  "Key": "location",
  "Value": "Seattle"
}
]
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## list-targeted-sentiment-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-targeted-sentiment-detection-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan deteksi sentimen yang ditargetkan

`list-targeted-sentiment-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi sentimen bertarget asinkron yang sedang berlangsung dan diselesaikan.

```
aws comprehend list-targeted-sentiment-detection-jobs
```

Output:

```
{
  "TargetedSentimentDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "example-targeted-sentiment-detection-job",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T22:42:20.545000+00:00",
      "EndTime": "2023-06-09T22:52:27.416000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
    },
  ],
}
```

```

    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-I0role"
  },
  {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE2",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE2",
    "JobName": "example-targeted-sentiment-detection-job-2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
    "EndTime": "2023-06-09T23:26:00.168000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData2",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE2/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListTargetedSentimentDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-topics-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-topics-detection-jobs`.

### AWS CLI

Untuk mencantumkan semua pekerjaan deteksi topik

`list-topics-detection-jobs` Contoh berikut mencantumkan semua pekerjaan deteksi topik asinkron yang sedang berlangsung dan diselesaikan.

```
aws comprehend list-topics-detection-jobs
```

Output:

```
{
  "TopicsDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "topic-analysis-1",
      "JobStatus": "IN_PROGRESS",
      "SubmitTime": "2023-06-09T18:40:35.384000+00:00",
      "EndTime": "2023-06-09T18:46:41.936000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
      },
      "NumberOfTopics": 10,
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    },
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE2",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-job/123456abcdeb0e11022f22a11EXAMPLE2",
      "JobName": "topic-analysis-2",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T18:44:43.414000+00:00",
      "EndTime": "2023-06-09T18:50:50.872000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
```

```

        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
    },
    "NumberOfTopics": 10,
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  },
  {
    "JobId": "123456abcdeb0e11022f22a1EXAMPLE3",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a1EXAMPLE3",
    "JobName": "topic-analysis-2",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T18:50:56.737000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a1EXAMPLE3/output/output.tar.gz"
    },
    "NumberOfTopics": 10,
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [ListTopicsDetectionJobs](#) di Referensi AWS CLI Perintah.

## put-resource-policy

Contoh kode berikut menunjukkan cara menggunakan `put-resource-policy`.

### AWS CLI

Untuk melampirkan kebijakan berbasis sumber daya

`put-resource-policy` Contoh berikut melampirkan kebijakan berbasis sumber daya ke model sehingga dapat diimpor oleh akun lain. AWS Kebijakan dilampirkan ke model dalam akun 111122223333 dan memungkinkan akun 444455556666 mengimpor model.

```
aws comprehend put-resource-policy \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1 \  
  --resource-policy '{"Version":"2012-10-17","Statement":  
[{"Effect":"Allow","Action":"comprehend:ImportModel","Resource":"*","Principal":  
{"AWS":["arn:aws:iam::444455556666:root"]}]}'
```

Ouput:

```
{  
  "PolicyRevisionId": "aaa111d069d07afaa2aa3106aEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menyalin model kustom antar AWS akun di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [PutResourcePolicy](#) di Referensi AWS CLI Perintah.

## start-document-classification-job

Contoh kode berikut menunjukkan cara menggunakan `start-document-classification-job`.

### AWS CLI

Untuk memulai pekerjaan klasifikasi dokumen

`start-document-classification-job` Contoh berikut memulai pekerjaan klasifikasi dokumen dengan model kustom pada semua file di alamat yang ditentukan oleh `--input-data-config` tag. Dalam contoh ini, bucket input S3 berisi `SampleSMStext1.txt`, `SampleSMStext2.txt`, dan `SampleSMStext3.txt`. Model ini sebelumnya dilatih pada klasifikasi dokumen spam dan non-spam, atau, “ham”, pesan SMS. Ketika pekerjaan selesai, `output.tar.gz` diletakkan di lokasi yang ditentukan oleh `--output-data-config` tag. `output.tar.gz` berisi `predictions.jsonl` yang mencantumkan klasifikasi setiap dokumen. Output Json dicetak pada satu baris per file, tetapi diformat di sini untuk keterbacaan.

```
aws comprehend start-document-classification-job \  

```

```
--job-name exampleclassificationjob \  
--input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET-INPUT/jobdata/" \  
--output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
--data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role \  
--document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/mymodel/version/12
```

Isi dari SampleSMStext1.txt:

```
"CONGRATULATIONS! TXT 2155550100 to win $5000"
```

Isi dari SampleSMStext2.txt:

```
"Hi, when do you want me to pick you up from practice?"
```

Isi dari SampleSMStext3.txt:

```
"Plz send bank account # to 2155550100 to claim prize!!"
```

Output:

```
{  
  "JobId": "e758dd56b824aa717ceab551fEXAMPLE",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:document-classification-  
job/e758dd56b824aa717ceab551fEXAMPLE",  
  "JobStatus": "SUBMITTED"  
}
```

Isi dari predictions.jsonl:

```
{"File": "SampleSMStext1.txt", "Line": "0", "Classes": [{"Name": "spam", "Score":  
0.9999}, {"Name": "ham", "Score": 0.0001}]}  
{"File": "SampleSMStext2.txt", "Line": "0", "Classes": [{"Name": "ham", "Score":  
0.9994}, {"Name": "spam", "Score": 0.0006}]}  
{"File": "SampleSMStext3.txt", "Line": "0", "Classes": [{"Name": "spam", "Score":  
0.9999}, {"Name": "ham", "Score": 0.0001}]}
```

Untuk informasi selengkapnya, lihat [Klasifikasi Kustom](#) di Panduan Pengembang Amazon Comprehend.



- Untuk detail API, lihat [StartDocumentClassificationJob](#) di Referensi AWS CLI Perintah.

## start-dominant-language-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-dominant-language-detection-job`.

### AWS CLI

Untuk memulai pekerjaan deteksi bahasa asinkron

`start-dominant-language-detection-job` Contoh berikut memulai pekerjaan deteksi bahasa asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag `--input-data-config` Bucket S3 dalam contoh ini berisi `Sampletext1.txt`. Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi `output.txt` yang berisi bahasa dominan dari masing-masing file teks serta skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi.

```
aws comprehend start-dominant-language-detection-job \
  --job-name example_language_analysis_job \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
  --language-code en
```

Isi dari `Sampletext1.txt`:

```
"Physics is the natural science that involves the study of matter and its motion and
behavior through space and time, along with related concepts such as energy and
force."
```

Output:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
```

```
}

```

Isi dari output.txt:

```
{"File": "Sampletext1.txt", "Languages": [{"LanguageCode": "en", "Score":
0.9913753867149353}], "Line": 0}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StartDominantLanguageDetectionJob](#) di Referensi AWS CLI Perintah.

## start-entities-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-entities-detection-job`.

### AWS CLI

Contoh 1: Untuk memulai pekerjaan deteksi entitas standar menggunakan model yang telah dilatih sebelumnya

`start-entities-detection-job` Contoh berikut memulai pekerjaan deteksi entitas asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Bucket S3 dalam contoh ini berisi `Sampletext1.txt`, `Sampletext2.txt`, dan `Sampletext3.txt`. Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi daftar semua entitas bernama `output.txt` yang terdeteksi dalam setiap file teks serta skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi. Output Json dicetak pada satu baris per file input, tetapi diformat di sini untuk keterbacaan.

```
aws comprehend start-entities-detection-job \
  --job-name entitiestest \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
  --language-code en
```

Isi dari `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July 31st."
```

Isi dari `Sampletext2.txt`:

```
"Dear Max, based on your autopay settings for your account example1.org account, we will withdraw your payment on the due date from your bank account number XXXXXX1111 with the routing number XXXXX0000. "
```

Isi dari `Sampletext3.txt`:

```
"Jane, please submit any customer feedback from this weekend to AnySpa, 123 Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Output:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Isi `output.txt` dengan indentasi garis untuk keterbacaan:

```
{
  "Entities": [
    {
      "BeginOffset": 6,
      "EndOffset": 15,
      "Score": 0.9994006636420306,
      "Text": "Zhang Wei",
      "Type": "PERSON"
    },
    {
      "BeginOffset": 22,
      "EndOffset": 26,
      "Score": 0.9976647915128143,
      "Text": "John",
      "Type": "PERSON"
    }
  ]
}
```

```
  },
  {
    "BeginOffset": 33,
    "EndOffset": 67,
    "Score": 0.9984608700836206,
    "Text": "AnyCompany Financial Services, LLC",
    "Type": "ORGANIZATION"
  },
  {
    "BeginOffset": 88,
    "EndOffset": 107,
    "Score": 0.9868521019555556,
    "Text": "1111-XXXX-1111-XXXX",
    "Type": "OTHER"
  },
  {
    "BeginOffset": 133,
    "EndOffset": 139,
    "Score": 0.998242565709204,
    "Text": "$24.53",
    "Type": "QUANTITY"
  },
  {
    "BeginOffset": 155,
    "EndOffset": 164,
    "Score": 0.9993039263159287,
    "Text": "July 31st",
    "Type": "DATE"
  }
],
"File": "SampleText1.txt",
"Line": 0
}
{
  "Entities": [
    {
      "BeginOffset": 5,
      "EndOffset": 8,
      "Score": 0.9866232147545232,
      "Text": "Max",
      "Type": "PERSON"
    },
    {
      "BeginOffset": 156,
```

```
"EndOffset": 166,
"Score": 0.9797723450933329,
"Text": "XXXXXX1111",
"Type": "OTHER"
},
{
"BeginOffset": 191,
"EndOffset": 200,
"Score": 0.9247838572396843,
"Text": "XXXXX0000",
"Type": "OTHER"
}
],
"File": "SampleText2.txt",
"Line": 0
}
{
"Entities": [
{
"Score": 0.9990532994270325,
"Type": "PERSON",
"Text": "Jane",
"BeginOffset": 0,
"EndOffset": 4
},
{
"Score": 0.9519651532173157,
"Type": "DATE",
"Text": "this weekend",
"BeginOffset": 47,
"EndOffset": 59
},
{
"Score": 0.5566426515579224,
"Type": "ORGANIZATION",
"Text": "AnySpa",
"BeginOffset": 63,
"EndOffset": 69
},
{
"Score": 0.8059805631637573,
"Type": "LOCATION",
"Text": "123 Main St, Anywhere",
"BeginOffset": 71,
```

```
"EndOffset": 92
},
{
  "Score": 0.998830258846283,
  "Type": "PERSON",
  "Text": "Alice",
  "BeginOffset": 114,
  "EndOffset": 119
},
{
  "Score": 0.997818112373352,
  "Type": "OTHER",
  "Text": "AnySpa@example.com",
  "BeginOffset": 123,
  "EndOffset": 138
}
],
"File": "SampleText3.txt",
"Line": 0
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

Contoh 2: Untuk memulai pekerjaan deteksi entitas kustom

`start-entities-detection-job` Contoh berikut memulai pekerjaan deteksi entitas kustom asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Dalam contoh ini, bucket S3 dalam contoh ini berisi `SampleFeedback1.txt`, `SampleFeedback2.txt`, dan `SampleFeedback3.txt`. Model pengenalan entitas dilatih tentang Umpan Balik dukungan pelanggan untuk mengenali nama perangkat. Ketika pekerjaan selesai, folder `output`, diletakkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi `output.txt`, yang mencantumkan semua entitas bernama yang terdeteksi dalam setiap file teks serta skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi. Output Json dicetak pada satu baris per file, tetapi diformat di sini untuk keterbacaan.

```
aws comprehend start-entities-detection-job \
  --job-name customentitiestest \
  --entity-recognizer-arn "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/entityrecognizer" \
```

```
--language-code en \  
--input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/jobdata/" \  
--output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
--data-access-role-arn "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-I0role"
```

Isi dari SampleFeedback1.txt:

```
"I've been on the AnyPhone app have had issues for 24 hours when trying to pay bill.  
Cannot make payment. Sigh. | Oh man! Lets get that app up and running. DM me, and  
we can get to work!"
```

Isi dari SampleFeedback2.txt:

```
"Hi, I have a discrepancy with my new bill. Could we get it sorted out? A rep added  
stuff I didnt sign up for when I did my AnyPhone 10 upgrade. | We can absolutely  
get this sorted!"
```

Isi dari SampleFeedback3.txt:

```
"Is the by 1 get 1 free AnySmartPhone promo still going on? | Hi Christian! It ended  
yesterday, send us a DM if you have any questions and we can take a look at your  
options!"
```

Output:

```
{  
  "JobId": "019ea9edac758806850fa8a79ff83021",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-  
job/019ea9edac758806850fa8a79ff83021",  
  "JobStatus": "SUBMITTED"  
}
```

Isi output.txt dengan indentasi garis untuk keterbacaan:

```
{  
  "Entities": [  
    {  
      "BeginOffset": 17,  
      "EndOffset": 25,
```

```
    "Score": 0.9999728210205924,
    "Text": "AnyPhone",
    "Type": "DEVICE"
  }
],
"File": "SampleFeedback1.txt",
"Line": 0
}
{
"Entities": [
  {
    "BeginOffset": 123,
    "EndOffset": 133,
    "Score": 0.9999892116761524,
    "Text": "AnyPhone 10",
    "Type": "DEVICE"
  }
],
"File": "SampleFeedback2.txt",
"Line": 0
}
{
"Entities": [
  {
    "BeginOffset": 23,
    "EndOffset": 35,
    "Score": 0.9999971389852362,
    "Text": "AnySmartPhone",
    "Type": "DEVICE"
  }
],
"File": "SampleFeedback3.txt",
"Line": 0
}
```

Untuk informasi selengkapnya, lihat [Pengenalan entitas khusus](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [StartEntitiesDetectionJob](#) di Referensi AWS CLI Perintah.

## **start-events-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `start-events-detection-job`.



## AWS CLI

Untuk memulai pekerjaan deteksi peristiwa asinkron

`start-events-detection-job` Contoh berikut memulai pekerjaan deteksi peristiwa asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Jenis acara target yang mungkin termasuk `BANKRUPTCY`, `EMPLOYMENT`, `CORPORATE_ACQUISITION`, `INVESTMENT_GENERAL`, `CORPORATE_MERGER`, dan `STOCK_SPLIT`. Bucket S3 dalam contoh ini berisi `SampleText1.txt`, `SampleText2.txt`, dan `SampleText3.txt`. Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi `SampleText1.txt.out`, `SampleText2.txt.out`, dan `SampleText3.txt.out`. Output JSON dicetak pada satu baris per file, tetapi diformat di sini untuk keterbacaan.

```
aws comprehend start-events-detection-job \  
  --job-name events-detection-1 \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/EventsData" \  
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-servicerole \  
  --language-code en \  
  --target-event-types "BANKRUPTCY" "EMPLOYMENT" "CORPORATE_ACQUISITION"  
"CORPORATE_MERGER" "INVESTMENT_GENERAL"
```

Isi dari `SampleText1.txt`:

```
"Company AnyCompany grew by increasing sales and through acquisitions. After  
purchasing competing firms in 2020, AnyBusiness, a part of the AnyBusinessGroup,  
gave Jane Does firm a going rate of one cent a gallon or forty-two cents a barrel."
```

Isi dari `SampleText2.txt`:

```
"In 2021, AnyCompany officially purchased AnyBusiness for 100 billion dollars,  
surprising and exciting the shareholders."
```

Isi dari `SampleText3.txt`:

```
"In 2022, AnyCompany stock crashed 50. Eventually later that year they filed for  
bankruptcy."
```

**Output:**

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Isi `SampleText1.txt.out` dengan indentasi garis untuk keterbacaan:

```
{
  "Entities": [
    {
      "Mentions": [
        {
          "BeginOffset": 8,
          "EndOffset": 18,
          "Score": 0.99977,
          "Text": "AnyCompany",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        },
        {
          "BeginOffset": 112,
          "EndOffset": 123,
          "Score": 0.999747,
          "Text": "AnyBusiness",
          "Type": "ORGANIZATION",
          "GroupScore": 0.979826
        },
        {
          "BeginOffset": 171,
          "EndOffset": 175,
          "Score": 0.999615,
          "Text": "firm",
          "Type": "ORGANIZATION",
          "GroupScore": 0.871647
        }
      ]
    },
    {
      "Mentions": [
```

```
    {
      "BeginOffset": 97,
      "EndOffset": 102,
      "Score": 0.987687,
      "Text": "firms",
      "Type": "ORGANIZATION",
      "GroupScore": 1
    }
  ],
},
{
  "Mentions": [
    {
      "BeginOffset": 103,
      "EndOffset": 110,
      "Score": 0.999458,
      "Text": "in 2020",
      "Type": "DATE",
      "GroupScore": 1
    }
  ]
},
{
  "Mentions": [
    {
      "BeginOffset": 160,
      "EndOffset": 168,
      "Score": 0.999649,
      "Text": "John Doe",
      "Type": "PERSON",
      "GroupScore": 1
    }
  ]
}
],
"Events": [
  {
    "Type": "CORPORATE_ACQUISITION",
    "Arguments": [
      {
        "EntityIndex": 0,
        "Role": "INVESTOR",
        "Score": 0.99977
      }
    ]
  }
]
```

```
],
  "Triggers": [
    {
      "BeginOffset": 56,
      "EndOffset": 68,
      "Score": 0.999967,
      "Text": "acquisitions",
      "Type": "CORPORATE_ACQUISITION",
      "GroupScore": 1
    }
  ]
},
{
  "Type": "CORPORATE_ACQUISITION",
  "Arguments": [
    {
      "EntityIndex": 1,
      "Role": "INVESTEES",
      "Score": 0.987687
    },
    {
      "EntityIndex": 2,
      "Role": "DATE",
      "Score": 0.999458
    },
    {
      "EntityIndex": 3,
      "Role": "INVESTOR",
      "Score": 0.999649
    }
  ]
},
  "Triggers": [
    {
      "BeginOffset": 76,
      "EndOffset": 86,
      "Score": 0.999973,
      "Text": "purchasing",
      "Type": "CORPORATE_ACQUISITION",
      "GroupScore": 1
    }
  ]
}
],
"File": "SampleText1.txt",
```

```
"Line": 0
}
```

Isi dari SampleText2.txt.out:

```
{
  "Entities": [
    {
      "Mentions": [
        {
          "BeginOffset": 0,
          "EndOffset": 7,
          "Score": 0.999473,
          "Text": "In 2021",
          "Type": "DATE",
          "GroupScore": 1
        }
      ]
    },
    {
      "Mentions": [
        {
          "BeginOffset": 9,
          "EndOffset": 19,
          "Score": 0.999636,
          "Text": "AnyCompany",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        }
      ]
    },
    {
      "Mentions": [
        {
          "BeginOffset": 45,
          "EndOffset": 56,
          "Score": 0.999712,
          "Text": "AnyBusiness",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        }
      ]
    }
  ],
}
```

```
{
  "Mentions": [
    {
      "BeginOffset": 61,
      "EndOffset": 80,
      "Score": 0.998886,
      "Text": "100 billion dollars",
      "Type": "MONETARY_VALUE",
      "GroupScore": 1
    }
  ]
},
"Events": [
  {
    "Type": "CORPORATE_ACQUISITION",
    "Arguments": [
      {
        "EntityIndex": 3,
        "Role": "AMOUNT",
        "Score": 0.998886
      },
      {
        "EntityIndex": 2,
        "Role": "INVESTEES",
        "Score": 0.999712
      },
      {
        "EntityIndex": 0,
        "Role": "DATE",
        "Score": 0.999473
      },
      {
        "EntityIndex": 1,
        "Role": "INVESTOR",
        "Score": 0.999636
      }
    ]
  },
  {
    "Type": "CORPORATE_ACQUISITION",
    "Arguments": [
      {
        "EntityIndex": 3,
        "Role": "AMOUNT",
        "Score": 0.998886
      },
      {
        "EntityIndex": 2,
        "Role": "INVESTEES",
        "Score": 0.999712
      },
      {
        "EntityIndex": 0,
        "Role": "DATE",
        "Score": 0.999473
      },
      {
        "EntityIndex": 1,
        "Role": "INVESTOR",
        "Score": 0.999636
      }
    ]
  }
],
"Triggers": [
  {
    "BeginOffset": 31,
    "EndOffset": 40,
    "Score": 0.99995,
    "Text": "purchased",
  }
]
```

```
        "Type": "CORPORATE_ACQUISITION",
        "GroupScore": 1
      }
    ]
  },
  "File": "SampleText2.txt",
  "Line": 0
}
```

Isi dari SampleText3.txt.out:

```
{
  "Entities": [
    {
      "Mentions": [
        {
          "BeginOffset": 9,
          "EndOffset": 19,
          "Score": 0.999774,
          "Text": "AnyCompany",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        },
        {
          "BeginOffset": 66,
          "EndOffset": 70,
          "Score": 0.995717,
          "Text": "they",
          "Type": "ORGANIZATION",
          "GroupScore": 0.997626
        }
      ]
    },
    {
      "Mentions": [
        {
          "BeginOffset": 50,
          "EndOffset": 65,
          "Score": 0.999656,
          "Text": "later that year",
          "Type": "DATE",
          "GroupScore": 1
        }
      ]
    }
  ]
}
```

```
    }
  ]
}
],
"Events": [
  {
    "Type": "BANKRUPTCY",
    "Arguments": [
      {
        "EntityIndex": 1,
        "Role": "DATE",
        "Score": 0.999656
      },
      {
        "EntityIndex": 0,
        "Role": "FILER",
        "Score": 0.995717
      }
    ]
  },
  {
    "BeginOffset": 81,
    "EndOffset": 91,
    "Score": 0.999936,
    "Text": "bankruptcy",
    "Type": "BANKRUPTCY",
    "GroupScore": 1
  }
]
}
],
"File": "SampleText3.txt",
"Line": 0
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StartEventsDetectionJob](#) di Referensi AWS CLI Perintah.

## start-flywheel-iteration

Contoh kode berikut menunjukkan cara menggunakan `start-flywheel-iteration`.



## AWS CLI

Untuk memulai iterasi flywheel

`start-flywheel-iteration` Contoh berikut memulai iterasi flywheel. Operasi ini menggunakan dataset baru di flywheel untuk melatih versi model baru.

```
aws comprehend start-flywheel-iteration \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel
```

Output:

```
{  
  "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel",  
  "FlywheelIterationId": "12345123TEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [ikhtisar Flywheel di Panduan](#) Pengembang Amazon Comprehend.

- Untuk detail API, lihat [StartFlywheelIteration](#) di Referensi AWS CLI Perintah.

## **start-key-phrases-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `start-key-phrases-detection-job`.

### AWS CLI

Untuk memulai pekerjaan deteksi frasa kunci

`start-key-phrases-detection-job` Contoh berikut memulai pekerjaan deteksi frasa kunci asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Bucket S3 dalam contoh ini berisi `Sampletext1.txt`, `Sampletext2.txt`, dan `Sampletext3.txt`. Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi file `output.txt` yang berisi semua frasa kunci yang terdeteksi dalam setiap file teks dan skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi. Output Json dicetak pada satu baris per file, tetapi diformat di sini untuk keterbacaan.

```
aws comprehend start-key-phrases-detection-job \  
  --job-name keyphrasesanalysisistest1 \  
  --language-code en \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \  
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
  --data-access-role-arn "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role" \  
  --language-code en
```

Isi dari Sampletext1.txt:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card  
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July  
31st."
```

Isi dari Sampletext2.txt:

```
"Dear Max, based on your autopay settings for your account Internet.org account, we  
will withdraw your payment on the due date from your bank account number XXXXXX1111  
with the routing number XXXXX0000. "
```

Isi dari Sampletext3.txt:

```
"Jane, please submit any customer feedback from this weekend to Sunshine Spa, 123  
Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Output:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-  
job/123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "SUBMITTED"  
}
```

Isi output.txt dengan indentasi garis untuk readability:

```
{  
  "File": "SampleText1.txt",  
  "KeyPhrases": [  
    {
```

```
"BeginOffset": 6,
"EndOffset": 15,
"Score": 0.9748965572679326,
"Text": "Zhang Wei"
},
{
"BeginOffset": 22,
"EndOffset": 26,
"Score": 0.9997344722354619,
"Text": "John"
},
{
"BeginOffset": 28,
"EndOffset": 62,
"Score": 0.9843791074032948,
"Text": "Your AnyCompany Financial Services"
},
{
"BeginOffset": 64,
"EndOffset": 107,
"Score": 0.8976122401721824,
"Text": "LLC credit card account 1111-XXXX-1111-XXXX"
},
{
"BeginOffset": 112,
"EndOffset": 129,
"Score": 0.9999612982629748,
"Text": "a minimum payment"
},
{
"BeginOffset": 133,
"EndOffset": 139,
"Score": 0.99975728947036,
"Text": "$24.53"
},
{
"BeginOffset": 155,
"EndOffset": 164,
"Score": 0.9940866241449973,
"Text": "July 31st"
}
],
"Line": 0
}
```

```
{
  "File": "SampleText2.txt",
  "KeyPhrases": [
    {
      "BeginOffset": 0,
      "EndOffset": 8,
      "Score": 0.9974021100118472,
      "Text": "Dear Max"
    },
    {
      "BeginOffset": 19,
      "EndOffset": 40,
      "Score": 0.9961120519515884,
      "Text": "your autopay settings"
    },
    {
      "BeginOffset": 45,
      "EndOffset": 78,
      "Score": 0.9980620070116009,
      "Text": "your account Internet.org account"
    },
    {
      "BeginOffset": 97,
      "EndOffset": 109,
      "Score": 0.999919660140754,
      "Text": "your payment"
    },
    {
      "BeginOffset": 113,
      "EndOffset": 125,
      "Score": 0.9998370719754205,
      "Text": "the due date"
    },
    {
      "BeginOffset": 131,
      "EndOffset": 166,
      "Score": 0.9955068678502509,
      "Text": "your bank account number XXXXXX1111"
    },
    {
      "BeginOffset": 172,
      "EndOffset": 200,
      "Score": 0.8653433315829526,
      "Text": "the routing number XXXXX0000"
    }
  ]
}
```

```
    }
  ],
  "Line": 0
}
{
  "File": "SampleText3.txt",
  "KeyPhrases": [
    {
      "BeginOffset": 0,
      "EndOffset": 4,
      "Score": 0.9142947833681668,
      "Text": "Jane"
    },
    {
      "BeginOffset": 20,
      "EndOffset": 41,
      "Score": 0.9984325676596763,
      "Text": "any customer feedback"
    },
    {
      "BeginOffset": 47,
      "EndOffset": 59,
      "Score": 0.9998782448150636,
      "Text": "this weekend"
    },
    {
      "BeginOffset": 63,
      "EndOffset": 75,
      "Score": 0.99866741830757,
      "Text": "Sunshine Spa"
    },
    {
      "BeginOffset": 77,
      "EndOffset": 88,
      "Score": 0.9695803485466054,
      "Text": "123 Main St"
    },
    {
      "BeginOffset": 108,
      "EndOffset": 116,
      "Score": 0.9997065928550928,
      "Text": "comments"
    }
  ]
}
```

```

    "BeginOffset": 120,
    "EndOffset": 125,
    "Score": 0.9993466833825161,
    "Text": "Alice"
  },
  {
    "BeginOffset": 129,
    "EndOffset": 144,
    "Score": 0.9654563612885667,
    "Text": "AnySpa@example.com"
  }
],
"Line": 0
}

```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembangan Amazon Comprehend](#).

- Untuk detail API, lihat [StartKeyPhrasesDetectionJob](#) di Referensi AWS CLI Perintah.

## start-pii-entities-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-pii-entities-detection-job`.

### AWS CLI

Untuk memulai pekerjaan deteksi PII asinkron

`start-pii-entities-detection-job` Contoh berikut memulai pekerjaan deteksi entitas informasi pribadi asinkron (PII) untuk semua file yang terletak di alamat yang ditentukan oleh tag `--input-data-config` Bucket S3 dalam contoh ini berisi `SampleText1.txt`, `SampleText2.txt`, dan `SampleText3.txt`. Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi `SampleText1.txt.out`, `SampleText2.txt.out`, dan `SampleText3.txt.out` yang mencantumkan entitas bernama dalam setiap file teks. Output Json dicetak pada satu baris per file, tetapi diformat di sini untuk keterbacaan.

```

aws comprehend start-pii-entities-detection-job \
  --job-name entities_test \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \

```

```
--data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
--language-code en \
--mode ONLY_OFFSETS
```

Isi dari Sampletext1.txt:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
31st."
```

Isi dari Sampletext2.txt:

```
"Dear Max, based on your autopay settings for your account Internet.org account, we
will withdraw your payment on the due date from your bank account number XXXXXX1111
with the routing number XXXXX0000. "
```

Isi dari Sampletext3.txt:

```
"Jane, please submit any customer feedback from this weekend to Sunshine Spa, 123
Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Output:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pai-entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Isi SampleText1.txt.out dengan indentasi garis untuk keterbacaan:

```
{
  "Entities": [
    {
      "BeginOffset": 6,
      "EndOffset": 15,
      "Type": "NAME",
      "Score": 0.9998490510222595
    }
  ]
}
```

```
    },
    {
      "BeginOffset": 22,
      "EndOffset": 26,
      "Type": "NAME",
      "Score": 0.9998937958019426
    },
    {
      "BeginOffset": 88,
      "EndOffset": 107,
      "Type": "CREDIT_DEBIT_NUMBER",
      "Score": 0.9554297245278491
    },
    {
      "BeginOffset": 155,
      "EndOffset": 164,
      "Type": "DATE_TIME",
      "Score": 0.9999720462925257
    }
  ],
  "File": "SampleText1.txt",
  "Line": 0
}
```

Isi SampleText2.txt.out dengan indentasi garis untuk keterbacaan:

```
{
  "Entities": [
    {
      "BeginOffset": 5,
      "EndOffset": 8,
      "Type": "NAME",
      "Score": 0.9994390774924007
    },
    {
      "BeginOffset": 58,
      "EndOffset": 70,
      "Type": "URL",
      "Score": 0.9999958276922101
    },
    {
      "BeginOffset": 156,
      "EndOffset": 166,

```



```
    "Type": "BANK_ACCOUNT_NUMBER",
    "Score": 0.9999721058045592
  },
  {
    "BeginOffset": 191,
    "EndOffset": 200,
    "Type": "BANK_ROUTING",
    "Score": 0.9998968945989909
  }
],
"File": "SampleText2.txt",
"Line": 0
}
```

Isi SampleText3.txt.out dengan indentasi garis untuk keterbacaan:

```
{
  "Entities": [
    {
      "BeginOffset": 0,
      "EndOffset": 4,
      "Type": "NAME",
      "Score": 0.999949934606805
    },
    {
      "BeginOffset": 77,
      "EndOffset": 88,
      "Type": "ADDRESS",
      "Score": 0.9999035300466904
    },
    {
      "BeginOffset": 120,
      "EndOffset": 125,
      "Type": "NAME",
      "Score": 0.9998203838716296
    },
    {
      "BeginOffset": 129,
      "EndOffset": 144,
      "Type": "EMAIL",
      "Score": 0.9998313473105228
    }
  ],
}
```

```
"File": "SampleText3.txt",  
"Line": 0  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StartPiiEntitiesDetectionJob](#) di Referensi AWS CLI Perintah.

## start-sentiment-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-sentiment-detection-job`.

### AWS CLI

Untuk memulai pekerjaan analisis sentimen asinkron

`start-sentiment-detection-job` Contoh berikut memulai pekerjaan deteksi analisis sentimen asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Folder bucket S3 dalam contoh ini berisi `SampleMovieReview1.txt`, `SampleMovieReview2.txt`, dan `SampleMovieReview3.txt`. Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. Folder berisi file, `output.txt`, yang berisi sentimen yang berlaku untuk setiap file teks dan skor kepercayaan model yang telah dilatih sebelumnya untuk setiap prediksi. Output Json dicetak pada satu baris per file, tetapi diformat di sini untuk keterbacaan.

```
aws comprehend start-sentiment-detection-job \  
  --job-name example-sentiment-detection-job \  
  --language-code en \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/MovieData" \  
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role
```

Isi dari `SampleMovieReview1.txt`:

```
"The film, AnyMovie2, is fairly predictable and just okay."
```

Isi dari `SampleMovieReview2.txt`:

```
"AnyMovie2 is the essential sci-fi film that I grew up watching when I was a kid. I highly recommend this movie."
```

Isi dari SampleMovieReview3.txt:

```
"Don't get fooled by the 'awards' for AnyMovie2. All parts of the film were poorly stolen from other modern directors."
```

Output:

```
{
  "JobId": "0b5001e25f62ebb40631a9a1a7fde7b3",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-detection-job/0b5001e25f62ebb40631a9a1a7fde7b3",
  "JobStatus": "SUBMITTED"
}
```

Isi output.txt dengan garis indentasi agar mudah dibaca:

```
{
  "File": "SampleMovieReview1.txt",
  "Line": 0,
  "Sentiment": "MIXED",
  "SentimentScore": {
    "Mixed": 0.6591159105300903,
    "Negative": 0.26492202281951904,
    "Neutral": 0.035430654883384705,
    "Positive": 0.04053137078881264
  }
}
{
  "File": "SampleMovieReview2.txt",
  "Line": 0,
  "Sentiment": "POSITIVE",
  "SentimentScore": {
    "Mixed": 0.000008718466233403888,
    "Negative": 0.00006134175055194646,
    "Neutral": 0.0002941041602753103,
    "Positive": 0.9996358156204224
  }
}
{
```

```
"File": "SampleMovieReview3.txt",
  "Line": 0,
  "Sentiment": "NEGATIVE",
  "SentimentScore": {
    "Mixed": 0.004146667663007975,
    "Negative": 0.9645107984542847,
    "Neutral": 0.016559595242142677,
    "Positive": 0.014782938174903393
  }
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StartSentimentDetectionJob](#) di Referensi AWS CLI Perintah.

## start-targeted-sentiment-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-targeted-sentiment-detection-job`.

### AWS CLI

Untuk memulai pekerjaan analisis sentimen bertarget asinkron

`start-targeted-sentiment-detection-job` Contoh berikut memulai pekerjaan deteksi analisis sentimen bertarget asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Folder bucket S3 dalam contoh ini berisi `SampleMovieReview1.txt`, `SampleMovieReview2.txt`, dan `SampleMovieReview3.txt`. Ketika pekerjaan selesai, `output.tar.gz` ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. `output.tar.gz` berisi file `SampleMovieReview1.txt.out`, `SampleMovieReview2.txt.out`, dan `SampleMovieReview3.txt.out`, yang masing-masing berisi semua entitas bernama dan sentimen terkait untuk satu file teks input.

```
aws comprehend start-targeted-sentiment-detection-job \
  --job-name targeted_movie_review_analysis1 \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/MovieData" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
```

```
--data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role
```

Isi dari SampleMovieReview1.txt:

```
"The film, AnyMovie, is fairly predictable and just okay."
```

Isi dari SampleMovieReview2.txt:

```
"AnyMovie is the essential sci-fi film that I grew up watching when I was a kid. I  
highly recommend this movie."
```

Isi dari SampleMovieReview3.txt:

```
"Don't get fooled by the 'awards' for AnyMovie. All parts of the film were poorly  
stolen from other modern directors."
```

Output:

```
{  
  "JobId": "0b5001e25f62ebb40631a9a1a7fde7b3",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-  
detection-job/0b5001e25f62ebb40631a9a1a7fde7b3",  
  "JobStatus": "SUBMITTED"  
}
```

Isi SampleMovieReview1.txt.out dengan indentasi garis untuk keterbacaan:

```
{  
  "Entities": [  
    {  
      "DescriptiveMentionIndex": [  
        0  
      ],  
      "Mentions": [  
        {  
          "BeginOffset": 4,  
          "EndOffset": 8,  
          "Score": 0.994972,  
          "GroupScore": 1,  
          "Text": "film",
```

```
    "Type": "MOVIE",
    "MentionSentiment": {
      "Sentiment": "NEUTRAL",
      "SentimentScore": {
        "Mixed": 0,
        "Negative": 0,
        "Neutral": 1,
        "Positive": 0
      }
    }
  ],
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "BeginOffset": 10,
      "EndOffset": 18,
      "Score": 0.631368,
      "GroupScore": 1,
      "Text": "AnyMovie",
      "Type": "ORGANIZATION",
      "MentionSentiment": {
        "Sentiment": "POSITIVE",
        "SentimentScore": {
          "Mixed": 0.001729,
          "Negative": 0.000001,
          "Neutral": 0.000318,
          "Positive": 0.997952
        }
      }
    }
  ]
}
],
"File": "SampleMovieReview1.txt",
"Line": 0
}
```

Isi indentasi SampleMovieReview2.txt.out baris untuk keterbacaan:

```
{
  "Entities": [
    {
      "DescriptiveMentionIndex": [
        0
      ],
      "Mentions": [
        {
          "BeginOffset": 0,
          "EndOffset": 8,
          "Score": 0.854024,
          "GroupScore": 1,
          "Text": "AnyMovie",
          "Type": "MOVIE",
          "MentionSentiment": {
            "Sentiment": "POSITIVE",
            "SentimentScore": {
              "Mixed": 0,
              "Negative": 0,
              "Neutral": 0.000007,
              "Positive": 0.999993
            }
          }
        }
      ],
        {
          "BeginOffset": 104,
          "EndOffset": 109,
          "Score": 0.999129,
          "GroupScore": 0.502937,
          "Text": "movie",
          "Type": "MOVIE",
          "MentionSentiment": {
            "Sentiment": "POSITIVE",
            "SentimentScore": {
              "Mixed": 0,
              "Negative": 0,
              "Neutral": 0,
              "Positive": 1
            }
          }
        }
      ],
        {
          "BeginOffset": 33,
```

```
    "EndOffset": 37,
    "Score": 0.999823,
    "GroupScore": 0.999252,
    "Text": "film",
    "Type": "MOVIE",
    "MentionSentiment": {
      "Sentiment": "POSITIVE",
      "SentimentScore": {
        "Mixed": 0,
        "Negative": 0,
        "Neutral": 0.000001,
        "Positive": 0.999999
      }
    }
  }
],
{
  "DescriptiveMentionIndex": [
    0,
    1,
    2
  ],
  "Mentions": [
    {
      "BeginOffset": 43,
      "EndOffset": 44,
      "Score": 0.999997,
      "GroupScore": 1,
      "Text": "I",
      "Type": "PERSON",
      "MentionSentiment": {
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
          "Mixed": 0,
          "Negative": 0,
          "Neutral": 1,
          "Positive": 0
        }
      }
    }
  ],
  {
    "BeginOffset": 80,
    "EndOffset": 81,
```



```
    "Score": 0.999996,
    "GroupScore": 0.52523,
    "Text": "I",
    "Type": "PERSON",
    "MentionSentiment": {
      "Sentiment": "NEUTRAL",
      "SentimentScore": {
        "Mixed": 0,
        "Negative": 0,
        "Neutral": 1,
        "Positive": 0
      }
    }
  },
  {
    "BeginOffset": 67,
    "EndOffset": 68,
    "Score": 0.999994,
    "GroupScore": 0.999499,
    "Text": "I",
    "Type": "PERSON",
    "MentionSentiment": {
      "Sentiment": "NEUTRAL",
      "SentimentScore": {
        "Mixed": 0,
        "Negative": 0,
        "Neutral": 1,
        "Positive": 0
      }
    }
  }
],
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "BeginOffset": 75,
      "EndOffset": 78,
      "Score": 0.999978,
      "GroupScore": 1,
      "Text": "kid",
```

```

        "Type": "PERSON",
        "MentionSentiment": {
            "Sentiment": "NEUTRAL",
            "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
            }
        }
    ]
}
],
"File": "SampleMovieReview2.txt",
"Line": 0
}

```

Isi SampleMovieReview3.txt.out dengan indentasi garis untuk readibilitas:

```

{
  "Entities": [
    {
      "DescriptiveMentionIndex": [
        1
      ],
      "Mentions": [
        {
          "BeginOffset": 64,
          "EndOffset": 68,
          "Score": 0.992953,
          "GroupScore": 0.999814,
          "Text": "film",
          "Type": "MOVIE",
          "MentionSentiment": {
            "Sentiment": "NEUTRAL",
            "SentimentScore": {
              "Mixed": 0.000004,
              "Negative": 0.010425,
              "Neutral": 0.989543,
              "Positive": 0.000027
            }
          }
        }
      ]
    }
  ]
}

```

```
    },
    {
      "BeginOffset": 37,
      "EndOffset": 45,
      "Score": 0.999782,
      "GroupScore": 1,
      "Text": "AnyMovie",
      "Type": "ORGANIZATION",
      "MentionSentiment": {
        "Sentiment": "POSITIVE",
        "SentimentScore": {
          "Mixed": 0.000095,
          "Negative": 0.039847,
          "Neutral": 0.000673,
          "Positive": 0.959384
        }
      }
    }
  ]
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "BeginOffset": 47,
      "EndOffset": 50,
      "Score": 0.999991,
      "GroupScore": 1,
      "Text": "All",
      "Type": "QUANTITY",
      "MentionSentiment": {
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
          "Mixed": 0.000001,
          "Negative": 0.000001,
          "Neutral": 0.999998,
          "Positive": 0
        }
      }
    }
  ]
},
```

```
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "BeginOffset": 106,
      "EndOffset": 115,
      "Score": 0.542083,
      "GroupScore": 1,
      "Text": "directors",
      "Type": "PERSON",
      "MentionSentiment": {
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
          "Mixed": 0,
          "Negative": 0,
          "Neutral": 1,
          "Positive": 0
        }
      }
    }
  ]
},
"File": "SampleMovieReview3.txt",
"Line": 0
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StartTargetedSentimentDetectionJob](#) di Referensi AWS CLI Perintah.

## start-topics-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-topics-detection-job`.

### AWS CLI

Untuk memulai pekerjaan analisis deteksi topik

`start-topics-detection-job` Contoh berikut memulai pekerjaan deteksi topik asinkron untuk semua file yang terletak di alamat yang ditentukan oleh tag. `--input-data-config` Ketika pekerjaan selesai, folder, `output`, ditempatkan di lokasi yang ditentukan oleh `--output-data-config` tag. `output` berisi `topic-terms.csv` dan `doc-topics.csv`. File keluaran pertama, `topic-terms.csv`, adalah daftar topik dalam koleksi. Untuk setiap topik, daftar tersebut mencakup, secara default, istilah teratas berdasarkan topik sesuai dengan bobotnya. File kedua, `doc-topics.csv`, mencantumkan dokumen yang terkait dengan topik dan proporsi dokumen yang berkaitan dengan topik tersebut.

```
aws comprehend start-topics-detection-job \
  --job-name example_topics_detection_job \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
  --language-code en
```

Output:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Untuk informasi selengkapnya, lihat [Pemodelan Topik](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [StartTopicsDetectionJob](#) di Referensi AWS CLI Perintah.

## **stop-dominant-language-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `stop-dominant-language-detection-job`.

AWS CLI

Untuk menghentikan pekerjaan deteksi bahasa dominan asinkron

`stop-dominant-language-detection-job` Contoh berikut menghentikan pekerjaan deteksi bahasa dominan asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-dominant-language-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopDominantLanguageDetectionJob](#) di Referensi AWS CLI Perintah.

## **stop-entities-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `stop-entities-detection-job`.

AWS CLI

Untuk menghentikan pekerjaan deteksi entitas asinkron

`stop-entities-detection-job` Contoh berikut menghentikan pekerjaan deteksi entitas asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-entities-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
```

```
"JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
"JobStatus": "STOP_REQUESTED"  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopEntitiesDetectionJob](#) di Referensi AWS CLI Perintah.

## stop-events-detection-job

Contoh kode berikut menunjukkan cara menggunakan `stop-events-detection-job`.

### AWS CLI

Untuk menghentikan pekerjaan deteksi peristiwa asinkron

`stop-events-detection-job` Contoh berikut menghentikan pekerjaan deteksi peristiwa asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-events-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopEventsDetectionJob](#) di Referensi AWS CLI Perintah.

## stop-key-phrases-detection-job

Contoh kode berikut menunjukkan cara menggunakan `stop-key-phrases-detection-job`.

## AWS CLI

Untuk menghentikan pekerjaan deteksi frase kunci asinkron

`stop-key-phrases-detection-job` Contoh berikut menghentikan pekerjaan deteksi frase kunci asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-key-phrases-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopKeyPhrasesDetectionJob](#) di Referensi AWS CLI Perintah.

## **stop-pii-entities-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `stop-pii-entities-detection-job`.

## AWS CLI

Untuk menghentikan pekerjaan deteksi entitas pii asinkron

`stop-pii-entities-detection-job` Contoh berikut menghentikan pekerjaan deteksi entitas pii asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-pii-entities-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```



**Output:**

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "STOP_REQUESTED"
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopPiiEntitiesDetectionJob](#) di Referensi AWS CLI Perintah.

**stop-sentiment-detection-job**

Contoh kode berikut menunjukkan cara menggunakan `stop-sentiment-detection-job`.

**AWS CLI**

Untuk menghentikan pekerjaan deteksi sentimen asinkron

`stop-sentiment-detection-job` Contoh berikut menghentikan pekerjaan deteksi sentimen asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-sentiment-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

**Output:**

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "STOP_REQUESTED"
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopSentimentDetectionJob](#) di Referensi AWS CLI Perintah.

## stop-targeted-sentiment-detection-job

Contoh kode berikut menunjukkan cara menggunakan `stop-targeted-sentiment-detection-job`.

### AWS CLI

Untuk menghentikan pekerjaan deteksi sentimen bertarget asinkron

`stop-targeted-sentiment-detection-job` Contoh berikut menghentikan pekerjaan deteksi sentimen bertarget asinkron yang sedang berlangsung. Jika status pekerjaan saat ini adalah `IN_PROGRESS` pekerjaan ditandai untuk pemutusan hubungan kerja dan dimasukkan ke dalam `STOP_REQUESTED` negara bagian. Jika pekerjaan selesai sebelum dapat dihentikan, itu dimasukkan ke dalam `COMPLETED` negara.

```
aws comprehend stop-targeted-sentiment-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Untuk informasi selengkapnya, lihat [Analisis asinkron untuk Amazon Comprehend insight di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [StopTargetedSentimentDetectionJob](#) di Referensi AWS CLI Perintah.

## stop-training-document-classifier

Contoh kode berikut menunjukkan cara menggunakan `stop-training-document-classifier`.

### AWS CLI

Untuk menghentikan pelatihan model pengklasifikasi dokumen

`stop-training-document-classifier` Contoh berikut menghentikan pelatihan model pengklasifikasi dokumen saat sedang berlangsung.

```
aws comprehend stop-training-document-classifier
```

```
--document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan mengelola model kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [StopTrainingDocumentClassifier](#) di Referensi AWS CLI Perintah.

## stop-training-entity-recognizer

Contoh kode berikut menunjukkan cara menggunakan `stop-training-entity-recognizer`.

### AWS CLI

Untuk menghentikan pelatihan model pengenalan entitas

`stop-training-entity-recognizer` Contoh berikut menghentikan pelatihan model pengenalan entitas saat dalam proses.

```
aws comprehend stop-training-entity-recognizer
  --entity-recognizer-arn "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/examplerrecognizer1"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan mengelola model kustom](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [StopTrainingEntityRecognizer](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Contoh 1: Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan satu tag ke sumber daya Amazon Comprehend.

```
aws comprehend tag-resource \
```

```
--resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1 \
--tags Key=Location,Value=Seattle
```

Perintah ini tidak memiliki output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan Pengembang Amazon Comprehend.

Contoh 2: Untuk menambahkan beberapa tag ke sumber daya

`tag-resource` Contoh berikut menambahkan beberapa tag ke sumber daya Amazon Comprehend.

```
aws comprehend tag-resource \
--resource-arn "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1" \
--tags Key=location,Value=Seattle Key=Department,Value=Finance
```

Perintah ini tidak memiliki output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Contoh 1: Untuk menghapus satu tag dari sumber daya

`untag-resource` Contoh berikut menghapus satu tag dari sumber daya Amazon Comprehend.

```
aws comprehend untag-resource \
--resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1
--tag-keys Location
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan Pengembang Amazon Comprehend.

Contoh 2: Untuk menghapus beberapa tag dari sumber daya

`untag-resource` Contoh berikut menghapus beberapa tag dari sumber daya Amazon Comprehend.

```
aws comprehend untag-resource \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1  
  --tag-keys Location Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan Pengembang Amazon Comprehend.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-endpoint

Contoh kode berikut menunjukkan cara menggunakan `update-endpoint`.

### AWS CLI

Contoh 1: Untuk memperbarui unit inferensi titik akhir

`update-endpoint` Contoh berikut memperbarui informasi tentang titik akhir. Dalam contoh ini, jumlah unit inferensi meningkat.

```
aws comprehend update-endpoint \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint  
  --desired-inference-units 2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

Contoh 2: Untuk memperbarui model aksi titik akhir

`update-endpoint` Contoh berikut memperbarui informasi tentang titik akhir. Dalam contoh ini, model aktif diubah.

```
aws comprehend update-endpoint \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint \  
  --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier-new
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola titik akhir Amazon Comprehend di Panduan Pengembang Amazon Comprehend](#).

- Untuk detail API, lihat [UpdateEndpoint](#) di Referensi AWS CLI Perintah.

## update-flywheel

Contoh kode berikut menunjukkan cara menggunakan `update-flywheel`.

### AWS CLI

Untuk memperbarui konfigurasi flywheel

`update-flywheel` Contoh berikut memperbarui konfigurasi flywheel. Dalam contoh ini, model aktif untuk flywheel diperbarui.

```
aws comprehend update-flywheel \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel-1 \  
  --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier/version/new-example-classifier-model
```

Output:

```
{  
  "FlywheelProperties": {  
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity",  
    "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier/version/new-example-classifier-model",
```

```
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "TaskConfig": {
      "LanguageCode": "en",
      "DocumentClassificationConfig": {
        "Mode": "MULTI_CLASS"
      }
    },
    "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/
schemaVersion=1/20230616T200543Z/",
    "DataSecurityConfig": {},
    "Status": "ACTIVE",
    "ModelType": "DOCUMENT_CLASSIFIER",
    "CreationTime": "2023-06-16T20:05:43.242000+00:00",
    "LastModifiedTime": "2023-06-19T04:00:43.027000+00:00",
    "LatestFlywheelIteration": "20230619T040032Z"
  }
}
```

Untuk informasi selengkapnya, lihat [ikhtisar Flywheel di Panduan](#) Pengembang Amazon Comprehend.

- Untuk detail API, lihat [UpdateFlywheel](#) di Referensi AWS CLI Perintah.

## Amazon Comprehend Medical contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum AWS Command Line Interface dengan menggunakan Amazon Comprehend Medical.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **describe-entities-detection-v2-job**

Contoh kode berikut menunjukkan cara menggunakan `describe-entities-detection-v2-job`.

#### AWS CLI

Untuk menggambarkan pekerjaan deteksi entitas

`describe-entities-detection-v2-job` Contoh berikut menampilkan properti yang terkait dengan pekerjaan deteksi entitas asinkron.

```
aws comprehendmedical describe-entities-detection-v2-job \  
  --job-id "ab9887877365fe70299089371c043b96"
```

#### Output:

```
{  
  "ComprehendMedicalAsyncJobProperties": {  
    "JobId": "ab9887877365fe70299089371c043b96",  
    "JobStatus": "COMPLETED",  
    "SubmitTime": "2020-03-18T21:20:15.614000+00:00",  
    "EndTime": "2020-03-18T21:27:07.350000+00:00",  
    "ExpirationTime": "2020-07-16T21:20:15+00:00",  
    "InputDataConfig": {  
      "S3Bucket": "comp-med-input",  
      "S3Key": ""  
    },  
    "OutputDataConfig": {  
      "S3Bucket": "comp-med-output",  
      "S3Key": "867139942017-EntitiesDetection-  
ab9887877365fe70299089371c043b96/"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/  
ComprehendMedicalBatchProcessingRole",  
    "ModelVersion": "DetectEntitiesModelV20190930"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.



- Untuk detail API, lihat [DescribeEntitiesDetectionV2Job](#) di Referensi AWS CLI Perintah.

## describe-icd10-cm-inference-job

Contoh kode berikut menunjukkan cara menggunakan `describe-icd10-cm-inference-job`.

### AWS CLI

Untuk menggambarkan pekerjaan inferensi ICD-10-CM

`describe-icd10-cm-inference-job` Contoh berikut menjelaskan properti pekerjaan inferensi yang diminta dengan `job-id` yang ditentukan.

```
aws comprehendmedical describe-icd10-cm-inference-job \  
  --job-id "5780034166536cdb52ffa3295a1b00a7"
```

### Output:

```
{  
  "ComprehendMedicalAsyncJobProperties": {  
    "JobId": "5780034166536cdb52ffa3295a1b00a7",  
    "JobStatus": "COMPLETED",  
    "SubmitTime": "2020-05-18T21:20:15.614000+00:00",  
    "EndTime": "2020-05-18T21:27:07.350000+00:00",  
    "ExpirationTime": "2020-09-16T21:20:15+00:00",  
    "InputDataConfig": {  
      "S3Bucket": "comp-med-input",  
      "S3Key": "AKIAIOSFODNN7EXAMPLE"  
    },  
    "OutputDataConfig": {  
      "S3Bucket": "comp-med-output",  
      "S3Key": "AKIAIOSFODNN7EXAMPLE"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/  
ComprehendMedicalBatchProcessingRole",  
    "ModelVersion": "0.1.0"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [DescribeJob](#) di Referensi AWS CLI Perintah.

## describe-phi-detection-job

Contoh kode berikut menunjukkan cara menggunakan `describe-phi-detection-job`.

### AWS CLI

Untuk menggambarkan pekerjaan deteksi PHI

Contoh berikut menampilkan properti yang terkait dengan pekerjaan deteksi informasi kesehatan yang dilindungi asinkron (PHI).

```
aws comprehendmedical describe-phi-detection-job \
  --job-id "4750034166536cdb52ffa3295a1b00a3"
```

### Output:

```
{
  "ComprehendMedicalAsyncJobProperties": {
    "JobId": "4750034166536cdb52ffa3295a1b00a3",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
    "EndTime": "2020-03-19T20:45:07.894000+00:00",
    "ExpirationTime": "2020-07-17T20:38:37+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": ""
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "867139942017-PHIDetection-4750034166536cdb52ffa3295a1b00a3/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "PHIModelV20190903"
  }
}
```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [DescribePhiDetectionJob](#) di Referensi AWS CLI Perintah.

## describe-rx-norm-inference-job

Contoh kode berikut menunjukkan cara menggunakan `describe-rx-norm-inference-job`.

### AWS CLI

Untuk menggambarkan pekerjaan RxNorm inferensi

`describe-rx-norm-inference-job` Contoh berikut menjelaskan properti pekerjaan inferensi yang diminta dengan `job-id` yang ditentukan.

```
aws comprehendmedical describe-rx-norm-inference-job \
  --job-id "eg8199877365fc70299089371c043b96"
```

Output:

```
{
  "ComprehendMedicalAsyncJobProperties": {
    "JobId": "g8199877365fc70299089371c043b96",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-05-18T21:20:15.614000+00:00",
    "EndTime": "2020-05-18T21:27:07.350000+00:00",
    "ExpirationTime": "2020-09-16T21:20:15+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.0.0"
  }
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [DescribeRxNormInferenceJob](#) di Referensi AWS CLI Perintah.

## describe-snomedct-inference-job

Contoh kode berikut menunjukkan cara menggunakan `describe-snomedct-inference-job`.

### AWS CLI

Untuk menggambarkan pekerjaan inferensi CT SNOMED

`describe-snomedct-inference-job` Contoh berikut menjelaskan properti pekerjaan inferensi yang diminta dengan `job-id` yang ditentukan.

```
aws comprehendmedical describe-snomedct-inference-job \  
  --job-id "2630034166536cdb52ffa3295a1b00a7"
```

Output:

```
{  
  "ComprehendMedicalAsyncJobProperties": {  
    "JobId": "2630034166536cdb52ffa3295a1b00a7",  
    "JobStatus": "COMPLETED",  
    "SubmitTime": "2021-12-18T21:20:15.614000+00:00",  
    "EndTime": "2021-12-18T21:27:07.350000+00:00",  
    "ExpirationTime": "2022-05-16T21:20:15+00:00",  
    "InputDataConfig": {  
      "S3Bucket": "comp-med-input",  
      "S3Key": "AKIAIOSFODNN7EXAMPLE"  
    },  
    "OutputDataConfig": {  
      "S3Bucket": "comp-med-output",  
      "S3Key": "AKIAIOSFODNN7EXAMPLE"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/  
ComprehendMedicalBatchProcessingRole",  
    "ModelVersion": "0.1.0"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [DescribeSnomedctInferenceJob](#) di Referensi AWS CLI Perintah.

## detect-entities-v2

Contoh kode berikut menunjukkan cara menggunakan `detect-entities-v2`.

### AWS CLI

Contoh 1: Untuk mendeteksi entitas langsung dari teks

`detect-entities-v2` Contoh berikut menunjukkan entitas yang terdeteksi dan memberi label sesuai dengan jenis, langsung dari teks masukan.

```
aws comprehendmedical detect-entities-v2 \  
  --text "Sleeping trouble on present dosage of Clonidine. Severe rash on face and  
  leg, slightly itchy."
```

Output:

```
{  
  "Id": 0,  
  "BeginOffset": 38,  
  "EndOffset": 47,  
  "Score": 0.9942955374717712,  
  "Text": "Clonidine",  
  "Category": "MEDICATION",  
  "Type": "GENERIC_NAME",  
  "Traits": []  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Entitas Versi 2](#) di Panduan Pengembang Medis Amazon Comprehend.

Contoh 2: Untuk mendeteksi entitas dari jalur file

`detect-entities-v2` Contoh berikut menunjukkan entitas yang terdeteksi dan memberi label sesuai dengan jenis dari jalur file.

```
aws comprehendmedical detect-entities-v2 \  
  --text file://medical_entities.txt
```

Isi dari `medical_entities.txt`:

```
{
```

```
"Sleeping trouble on present dosage of Clonidine. Severe rash on face and leg,
slightly itchy."
}
```

Output:

```
{
  "Id": 0,
  "BeginOffset": 38,
  "EndOffset": 47,
  "Score": 0.9942955374717712,
  "Text": "Clonidine",
  "Category": "MEDICATION",
  "Type": "GENERIC_NAME",
  "Traits": []
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Entitas Versi 2](#) di Panduan Pengembang Medis Amazon Comprehend.

- Untuk detail API, lihat [DetectEntitiesV2](#) di Referensi AWS CLI Perintah.

## detect-phi

Contoh kode berikut menunjukkan cara menggunakan `detect-phi`.

### AWS CLI

Contoh 1: Untuk mendeteksi informasi kesehatan yang dilindungi (PHI) langsung dari teks

`detect-phi` Contoh berikut menampilkan entitas informasi kesehatan terlindungi (PHI) yang terdeteksi langsung dari teks masukan.

```
aws comprehendmedical detect-phi \
  --text "Patient Carlos Salazar presented with rash on his upper extremities and
dry cough. He lives at 100 Main Street, Anytown, USA where he works from his home
as a carpenter."
```

Output:

```
{
  "Entities": [
```

```

    {
      "Id": 0,
      "BeginOffset": 8,
      "EndOffset": 21,
      "Score": 0.9914507269859314,
      "Text": "Carlos Salazar",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "NAME",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 94,
      "EndOffset": 109,
      "Score": 0.871849775314331,
      "Text": "100 Main Street, Anytown, USA",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "ADDRESS",
      "Traits": []
    },
    {
      "Id": 2,
      "BeginOffset": 145,
      "EndOffset": 154,
      "Score": 0.8302185535430908,
      "Text": "carpenter",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "PROFESSION",
      "Traits": []
    }
  ],
  "ModelVersion": "0.0.0"
}

```

Untuk informasi selengkapnya, lihat [Deteksi PHI](#) di Amazon Comprehend Medical Developer Guide.

Contoh 2: Untuk mendeteksi melindungi informasi kesehatan (PHI) langsung dari jalur file

`detect-phi` Contoh berikut menunjukkan entitas informasi kesehatan terlindungi (PHI) yang terdeteksi dari jalur file.

```
aws comprehendmedical detect-phi \
```

```
--text file://phi.txt
```

Isi dari phi.txt:

```
"Patient Carlos Salazar presented with a rash on his upper extremities and a dry cough. He lives at 100 Main Street, Anytown, USA, where he works from his home as a carpenter."
```

Output:

```
{
  "Entities": [
    {
      "Id": 0,
      "BeginOffset": 8,
      "EndOffset": 21,
      "Score": 0.9914507269859314,
      "Text": "Carlos Salazar",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "NAME",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 94,
      "EndOffset": 109,
      "Score": 0.871849775314331,
      "Text": "100 Main Street, Anytown, USA",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "ADDRESS",
      "Traits": []
    },
    {
      "Id": 2,
      "BeginOffset": 145,
      "EndOffset": 154,
      "Score": 0.8302185535430908,
      "Text": "carpenter",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "PROFESSION",
      "Traits": []
    }
  ],
}
```



```
"ModelVersion": "0.0.0"
}
```

Untuk informasi selengkapnya, lihat [Deteksi PHI](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [DetectPhi](#) di Referensi AWS CLI Perintah.

## infer-icd10-cm

Contoh kode berikut menunjukkan cara menggunakan `infer-icd10-cm`.

### AWS CLI

Contoh 1: Untuk mendeteksi entitas kondisi medis dan menautkan ke Ontologi ICD-10-CM langsung dari teks

`infer-icd10-cm` Contoh berikut memberi label pada entitas kondisi medis yang terdeteksi dan menghubungkan entitas tersebut dengan kode dalam edisi 2019 dari International Classification of Diseases Clinical Modification (ICD-10-CM).

```
aws comprehendmedical infer-icd10-cm \
  --text "The patient complains of abdominal pain, has a long-standing history of
  diabetes treated with Micronase daily."
```

Output:

```
{
  "Entities": [
    {
      "Id": 0,
      "Text": "abdominal pain",
      "Category": "MEDICAL_CONDITION",
      "Type": "DX_NAME",
      "Score": 0.9475538730621338,
      "BeginOffset": 28,
      "EndOffset": 42,
      "Attributes": [],
      "Traits": [
        {
          "Name": "SYMPTOM",
          "Score": 0.6724207401275635
        }
      ]
    }
  ]
}
```

```
    }
  ],
  "ICD10CMConcepts": [
    {
      "Description": "Unspecified abdominal pain",
      "Code": "R10.9",
      "Score": 0.6904221177101135
    },
    {
      "Description": "Epigastric pain",
      "Code": "R10.13",
      "Score": 0.1364113688468933
    },
    {
      "Description": "Generalized abdominal pain",
      "Code": "R10.84",
      "Score": 0.12508003413677216
    },
    {
      "Description": "Left lower quadrant pain",
      "Code": "R10.32",
      "Score": 0.10063883662223816
    },
    {
      "Description": "Lower abdominal pain, unspecified",
      "Code": "R10.30",
      "Score": 0.09933677315711975
    }
  ]
},
{
  "Id": 1,
  "Text": "diabetes",
  "Category": "MEDICAL_CONDITION",
  "Type": "DX_NAME",
  "Score": 0.9899052977561951,
  "BeginOffset": 75,
  "EndOffset": 83,
  "Attributes": [],
  "Traits": [
    {
      "Name": "DIAGNOSIS",
      "Score": 0.9258432388305664
    }
  ]
}
```

```

    ],
    "ICD10CMConcepts": [
      {
        "Description": "Type 2 diabetes mellitus without complications",
        "Code": "E11.9",
        "Score": 0.7158446311950684
      },
      {
        "Description": "Family history of diabetes mellitus",
        "Code": "Z83.3",
        "Score": 0.5704703330993652
      },
      {
        "Description": "Family history of other endocrine, nutritional
and metabolic diseases",
        "Code": "Z83.49",
        "Score": 0.19856023788452148
      },
      {
        "Description": "Type 1 diabetes mellitus with ketoacidosis
without coma",
        "Code": "E10.10",
        "Score": 0.13285516202449799
      },
      {
        "Description": "Type 2 diabetes mellitus with hyperglycemia",
        "Code": "E11.65",
        "Score": 0.0993388369679451
      }
    ]
  }
],
  "ModelVersion": "0.1.0"
}

```

Untuk informasi lebih lanjut, lihat [Menyimpulkan ICD10-CM di Panduan Pengembang Medis Amazon Comprehend Medical](#).

Contoh 2: Untuk mendeteksi entitas kondisi medis dan menautkan ke Ontologi ICD-10-CM dari jalur file

`infer-icd-10-cm` Contoh berikut memberi label pada entitas kondisi medis yang terdeteksi dan menghubungkan entitas tersebut dengan kode dalam edisi 2019 dari International Classification of Diseases Clinical Modification (ICD-10-CM).

```
aws comprehendmedical infer-icd10-cm \  
  --text file://icd10cm.txt
```

Isi dari `icd10cm.txt`:

```
{  
  "The patient complains of abdominal pain, has a long-standing history of  
  diabetes treated with Micronase daily."  
}
```

Output:

```
{  
  "Entities": [  
    {  
      "Id": 0,  
      "Text": "abdominal pain",  
      "Category": "MEDICAL_CONDITION",  
      "Type": "DX_NAME",  
      "Score": 0.9475538730621338,  
      "BeginOffset": 28,  
      "EndOffset": 42,  
      "Attributes": [],  
      "Traits": [  
        {  
          "Name": "SYMPTOM",  
          "Score": 0.6724207401275635  
        }  
      ],  
      "ICD10CMConcepts": [  
        {  
          "Description": "Unspecified abdominal pain",  
          "Code": "R10.9",  
          "Score": 0.6904221177101135  
        },  
        {  
          "Description": "Epigastric pain",  
          "Code": "R10.13",  
        }  
      ]  
    }  
  ]  
}
```

```

        "Score": 0.1364113688468933
      },
      {
        "Description": "Generalized abdominal pain",
        "Code": "R10.84",
        "Score": 0.12508003413677216
      },
      {
        "Description": "Left lower quadrant pain",
        "Code": "R10.32",
        "Score": 0.10063883662223816
      },
      {
        "Description": "Lower abdominal pain, unspecified",
        "Code": "R10.30",
        "Score": 0.09933677315711975
      }
    ]
  },
  {
    "Id": 1,
    "Text": "diabetes",
    "Category": "MEDICAL_CONDITION",
    "Type": "DX_NAME",
    "Score": 0.9899052977561951,
    "BeginOffset": 75,
    "EndOffset": 83,
    "Attributes": [],
    "Traits": [
      {
        "Name": "DIAGNOSIS",
        "Score": 0.9258432388305664
      }
    ],
    "ICD10CMConcepts": [
      {
        "Description": "Type 2 diabetes mellitus without complications",
        "Code": "E11.9",
        "Score": 0.7158446311950684
      },
      {
        "Description": "Family history of diabetes mellitus",
        "Code": "Z83.3",
        "Score": 0.5704703330993652
      }
    ]
  }
]

```

```

        },
        {
            "Description": "Family history of other endocrine, nutritional
and metabolic diseases",
            "Code": "Z83.49",
            "Score": 0.19856023788452148
        },
        {
            "Description": "Type 1 diabetes mellitus with ketoacidosis
without coma",
            "Code": "E10.10",
            "Score": 0.13285516202449799
        },
        {
            "Description": "Type 2 diabetes mellitus with hyperglycemia",
            "Code": "E11.65",
            "Score": 0.0993388369679451
        }
    ]
}
],
"ModelVersion": "0.1.0"
}

```

Untuk informasi lebih lanjut, lihat [Infer-ICD10-CM](#) di Panduan Pengembang Medis Amazon Comprehend Medical.

- Untuk detail API, lihat [InferIcd10Cm](#) di Referensi AWS CLI Perintah.

## **infer-rx-norm**

Contoh kode berikut menunjukkan cara menggunakan `infer-rx-norm`.

### AWS CLI

Contoh 1: Untuk mendeteksi entitas obat dan menautkan RxNorm langsung dari teks

`infer-rx-norm` Contoh berikut menunjukkan dan memberi label entitas obat yang terdeteksi dan menghubungkan entitas tersebut ke pengidentifikasi konsep (RxCui) dari database Perpustakaan Kedokteran Nasional. RxNorm

```
aws comprehendmedical infer-rx-norm \
```

```
--text "Patient reports taking Levothyroxine 125 micrograms p.o. once daily, but denies taking Synthroid."
```

## Output:

```
{
  "Entities": [
    {
      "Id": 0,
      "Text": "Levothyroxine",
      "Category": "MEDICATION",
      "Type": "GENERIC_NAME",
      "Score": 0.9996285438537598,
      "BeginOffset": 23,
      "EndOffset": 36,
      "Attributes": [
        {
          "Type": "DOSAGE",
          "Score": 0.9892290830612183,
          "RelationshipScore": 0.9997978806495667,
          "Id": 1,
          "BeginOffset": 37,
          "EndOffset": 51,
          "Text": "125 micrograms",
          "Traits": []
        },
        {
          "Type": "ROUTE_OR_MODE",
          "Score": 0.9988924860954285,
          "RelationshipScore": 0.998291552066803,
          "Id": 2,
          "BeginOffset": 52,
          "EndOffset": 56,
          "Text": "p.o.",
          "Traits": []
        },
        {
          "Type": "FREQUENCY",
          "Score": 0.9953463673591614,
          "RelationshipScore": 0.9999889135360718,
          "Id": 3,
          "BeginOffset": 57,
          "EndOffset": 67,

```

```

        "Text": "once daily",
        "Traits": []
    }
],
"Traits": [],
"RxNormConcepts": [
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet",
        "Code": "966224",
        "Score": 0.9912070631980896
    },
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Capsule",
        "Code": "966405",
        "Score": 0.8698278665542603
    },
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
        "Code": "966191",
        "Score": 0.7448257803916931
    },
    {
        "Description": "levothyroxine",
        "Code": "10582",
        "Score": 0.7050482630729675
    },
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Levoxyl]",
        "Code": "966190",
        "Score": 0.6921631693840027
    }
]
},
{
    "Id": 4,
    "Text": "Synthroid",
    "Category": "MEDICATION",
    "Type": "BRAND_NAME",
    "Score": 0.9946461319923401,
    "BeginOffset": 86,
    "EndOffset": 95,
    "Attributes": [],

```



```
    "Traits": [
      {
        "Name": "NEGATION",
        "Score": 0.5167351961135864
      }
    ],
    "RxNormConcepts": [
      {
        "Description": "Synthroid",
        "Code": "224920",
        "Score": 0.9462039470672607
      },
      {
        "Description": "Levothyroxine Sodium 0.088 MG Oral Tablet
[Synthroid]",
        "Code": "966282",
        "Score": 0.8309829235076904
      },
      {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
        "Code": "966191",
        "Score": 0.4945160448551178
      },
      {
        "Description": "Levothyroxine Sodium 0.05 MG Oral Tablet
[Synthroid]",
        "Code": "966247",
        "Score": 0.3674522042274475
      },
      {
        "Description": "Levothyroxine Sodium 0.025 MG Oral Tablet
[Synthroid]",
        "Code": "966158",
        "Score": 0.2588822841644287
      }
    ]
  }
],
  "ModelVersion": "0.0.0"
}
```

Untuk informasi selengkapnya, lihat [Infer RxNorm di Panduan Pengembang](#) Medis Amazon Comprehend Medical.

Contoh 2: Untuk mendeteksi entitas obat dan menautkan ke RxNorm dari jalur file.

`infer-rx-norm` Contoh berikut menunjukkan dan memberi label entitas obat yang terdeteksi dan menghubungkan entitas tersebut ke pengidentifikasi konsep (RxCui) dari database Perpustakaan Kedokteran Nasional. RxNorm

```
aws comprehendmedical infer-rx-norm \  
  --text file://rxnorm.txt
```

Isi dari `rxnorm.txt`:

```
{  
  "Patient reports taking Levothyroxine 125 micrograms p.o. once daily, but denies  
  taking Synthroid."  
}
```

Output:

```
{  
  "Entities": [  
    {  
      "Id": 0,  
      "Text": "Levothyroxine",  
      "Category": "MEDICATION",  
      "Type": "GENERIC_NAME",  
      "Score": 0.9996285438537598,  
      "BeginOffset": 23,  
      "EndOffset": 36,  
      "Attributes": [  
        {  
          "Type": "DOSAGE",  
          "Score": 0.9892290830612183,  
          "RelationshipScore": 0.9997978806495667,  
          "Id": 1,  
          "BeginOffset": 37,  
          "EndOffset": 51,  
          "Text": "125 micrograms",  
          "Traits": []  
        },  
      ],  
    },  
  ],  
}
```

```

    {
      "Type": "ROUTE_OR_MODE",
      "Score": 0.9988924860954285,
      "RelationshipScore": 0.998291552066803,
      "Id": 2,
      "BeginOffset": 52,
      "EndOffset": 56,
      "Text": "p.o.",
      "Traits": []
    },
    {
      "Type": "FREQUENCY",
      "Score": 0.9953463673591614,
      "RelationshipScore": 0.9999889135360718,
      "Id": 3,
      "BeginOffset": 57,
      "EndOffset": 67,
      "Text": "once daily",
      "Traits": []
    }
  ],
  "Traits": [],
  "RxNormConcepts": [
    {
      "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet",
      "Code": "966224",
      "Score": 0.9912070631980896
    },
    {
      "Description": "Levothyroxine Sodium 0.125 MG Oral Capsule",
      "Code": "966405",
      "Score": 0.8698278665542603
    },
    {
      "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
      "Code": "966191",
      "Score": 0.7448257803916931
    },
    {
      "Description": "levothyroxine",
      "Code": "10582",
      "Score": 0.7050482630729675
    },
  ],

```

```

        {
            "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Levoxy1]",
            "Code": "966190",
            "Score": 0.6921631693840027
        }
    ],
    {
        "Id": 4,
        "Text": "Synthroid",
        "Category": "MEDICATION",
        "Type": "BRAND_NAME",
        "Score": 0.9946461319923401,
        "BeginOffset": 86,
        "EndOffset": 95,
        "Attributes": [],
        "Traits": [
            {
                "Name": "NEGATION",
                "Score": 0.5167351961135864
            }
        ],
        "RxNormConcepts": [
            {
                "Description": "Synthroid",
                "Code": "224920",
                "Score": 0.9462039470672607
            },
            {
                "Description": "Levothyroxine Sodium 0.088 MG Oral Tablet
[Synthroid]",
                "Code": "966282",
                "Score": 0.8309829235076904
            },
            {
                "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
                "Code": "966191",
                "Score": 0.4945160448551178
            },
            {
                "Description": "Levothyroxine Sodium 0.05 MG Oral Tablet
[Synthroid]",

```

```

        "Code": "966247",
        "Score": 0.3674522042274475
      },
      {
        "Description": "Levothyroxine Sodium 0.025 MG Oral Tablet
[Synthroid]",
        "Code": "966158",
        "Score": 0.2588822841644287
      }
    ]
  }
],
  "ModelVersion": "0.0.0"
}

```

Untuk informasi selengkapnya, lihat [Infer RxNorm di Panduan Pengembang](#) Medis Amazon Comprehend Medical.

- Untuk detail API, lihat [InferRxNorm](#) di Referensi AWS CLI Perintah.

## **infer-snomedct**

Contoh kode berikut menunjukkan cara menggunakan `infer-snomedct`.

### AWS CLI

Contoh: Untuk mendeteksi entitas dan menautkan ke SNOMED CT Ontology langsung dari teks

`infer-snomedct` Contoh berikut menunjukkan cara mendeteksi entitas medis dan menghubungkannya dengan konsep dari versi 2021-03 dari Nomenklatur Kedokteran Sistematisasi, Istilah Klinis (SNOMED CT).

```

aws comprehendmedical infer-snomedct \
  --text "The patient complains of abdominal pain, has a long-standing history of
diabetes treated with Micronase daily."

```

Output:

```

{
  "Entities": [
    {
      "Id": 3,
      "BeginOffset": 26,

```

```
    "EndOffset": 40,
    "Score": 0.9598260521888733,
    "Text": "abdominal pain",
    "Category": "MEDICAL_CONDITION",
    "Type": "DX_NAME",
    "Traits": [
      {
        "Name": "SYMPTOM",
        "Score": 0.6819021701812744
      }
    ]
  },
  {
    "Id": 4,
    "BeginOffset": 73,
    "EndOffset": 81,
    "Score": 0.9905840158462524,
    "Text": "diabetes",
    "Category": "MEDICAL_CONDITION",
    "Type": "DX_NAME",
    "Traits": [
      {
        "Name": "DIAGNOSIS",
        "Score": 0.9255214333534241
      }
    ]
  },
  {
    "Id": 1,
    "BeginOffset": 95,
    "EndOffset": 104,
    "Score": 0.6371926665306091,
    "Text": "Micronase",
    "Category": "MEDICATION",
    "Type": "BRAND_NAME",
    "Traits": [],
    "Attributes": [
      {
        "Type": "FREQUENCY",
        "Score": 0.9761165380477905,
        "RelationshipScore": 0.9984188079833984,
        "RelationshipType": "FREQUENCY",
        "Id": 2,
        "BeginOffset": 105,
```

```

        "EndOffset": 110,
        "Text": "daily",
        "Category": "MEDICATION",
        "Traits": []
      }
    ]
  },
  "UnmappedAttributes": [],
  "ModelVersion": "1.0.0"
}

```

Untuk informasi selengkapnya, lihat [InfersNoMEDCT](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [InferSnomedct](#) di Referensi AWS CLI Perintah.

## list-entities-detection-v2-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-entities-detection-v2-jobs`.

### AWS CLI

Untuk mencantumkan pekerjaan deteksi entitas

`list-entities-detection-v2-jobs` Contoh berikut mencantumkan pekerjaan deteksi asinkron saat ini.

```
aws comprehendmedical list-entities-detection-v2-jobs
```

Output:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "ab9887877365fe70299089371c043b96",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
      "EndTime": "2020-03-19T20:45:07.894000+00:00",
      "ExpirationTime": "2020-07-17T20:38:37+00:00",
      "InputDataConfig": {
        "S3Bucket": "comp-med-input",

```

```

        "S3Key": ""
    },
    "OutputDataConfig": {
        "S3Bucket": "comp-med-output",
        "S3Key": "867139942017-EntitiesDetection-
ab9887877365fe70299089371c043b96/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "DetectEntitiesModelV20190930"
}
]
}

```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [ListEntitiesDetectionV2Jobs](#) di Referensi AWS CLI Perintah.

## list-icd10-cm-inference-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-icd10-cm-inference-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan inferensi ICD-10-CM saat ini

Contoh berikut menunjukkan bagaimana `list-icd10-cm-inference-jobs` operasi mengembalikan daftar pekerjaan inferensi batch ICD-10-CM asinkron saat ini.

```
aws comprehendmedical list-icd10-cm-inference-jobs
```

### Output:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "5780034166536cdb52ffa3295a1b00a7",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
      "EndTime": "2020-05-19T20:45:07.894000+00:00",
      "ExpirationTime": "2020-09-17T20:38:37+00:00",
    }
  ]
}

```



```

    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.1.0"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [Listcd10 CmlInferenceJobs](#) di Referensi AWS CLI Perintah.

## list-phi-detection-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-phi-detection-jobs`.

### AWS CLI

Untuk membuat daftar pekerjaan deteksi informasi kesehatan yang dilindungi (PHI)

`list-phi-detection-jobs` Contoh berikut mencantumkan pekerjaan deteksi informasi kesehatan terlindungi (PHI) saat ini

```
aws comprehendmedical list-phi-detection-jobs
```

Output:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "4750034166536cdb52ffa3295a1b00a3",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-03-19T20:38:37.594000+00:00",

```

```

    "EndTime": "2020-03-19T20:45:07.894000+00:00",
    "ExpirationTime": "2020-07-17T20:38:37+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": ""
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "867139942017-
PHIDetection-4750034166536cdb52ffa3295a1b00a3/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "PHIModelV20190903"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [ListPhiDetectionJobs](#) di Referensi AWS CLI Perintah.

## list-rx-norm-inference-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-rx-norm-inference-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan inferensi Rx-Norm saat ini

Contoh berikut menunjukkan bagaimana `list-rx-norm-inference-jobs` mengembalikan daftar pekerjaan inferensi batch Rx-Norm asinkron saat ini.

```
aws comprehendmedical list-rx-norm-inference-jobs
```

### Output:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "4980034166536cfb52gga3295a1b00a3",

```

```

    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
    "EndTime": "2020-05-19T20:45:07.894000+00:00",
    "ExpirationTime": "2020-09-17T20:38:37+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.0.0"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [ListRxNormInferenceJobs](#) di Referensi AWS CLI Perintah.

## list-snomedct-inference-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-snomedct-inference-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan inferensi CT SNOMED

Contoh berikut menunjukkan bagaimana `list-snomedct-inference-jobs` operasi mengembalikan daftar pekerjaan inferensi batch SNOMED CT asinkron saat ini.

```
aws comprehendmedical list-snomedct-inference-jobs
```

Output:

```
{
  "ComprehendMedicalAsyncJobPropertiesList": [
```

```

    {
      "JobId": "5780034166536cdb52ffa3295a1b00a7",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
      "EndTime": "2020-05-19T20:45:07.894000+00:00",
      "ExpirationTime": "2020-09-17T20:38:37+00:00",
      "InputDataConfig": {
        "S3Bucket": "comp-med-input",
        "S3Key": "AKIAIOSFODNN7EXAMPLE"
      },
      "OutputDataConfig": {
        "S3Bucket": "comp-med-output",
        "S3Key": "AKIAIOSFODNN7EXAMPLE"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
      "ModelVersion": "0.1.0"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [ListSnomedctInferenceJobs](#) di Referensi AWS CLI Perintah.

## start-entities-detection-v2-job

Contoh kode berikut menunjukkan cara menggunakan `start-entities-detection-v2-job`.

### AWS CLI

Untuk memulai pekerjaan deteksi entitas

`start-entities-detection-v2-job` Contoh berikut memulai pekerjaan deteksi entitas asinkron.

```

aws comprehendmedical start-entities-detection-v2-job \
  --input-data-config "S3Bucket=comp-med-input" \
  --output-data-config "S3Bucket=comp-med-output" \
  --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \

```

```
--language-code en
```

Output:

```
{
  "JobId": "ab9887877365fe70299089371c043b96"
}
```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StartEntitiesDetectionV2Job](#) di Referensi AWS CLI Perintah.

## start-icd10-cm-inference-job

Contoh kode berikut menunjukkan cara menggunakan `start-icd10-cm-inference-job`.

AWS CLI

Untuk memulai pekerjaan inferensi ICD-10-CM

`start-icd10-cm-inference-job` Contoh berikut memulai pekerjaan analisis batch inferensi ICD-10-CM.

```
aws comprehendmedical start-icd10-cm-inference-job \
  --input-data-config "S3Bucket=comp-med-input" \
  --output-data-config "S3Bucket=comp-med-output" \
  --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
  --language-code en
```

Output:

```
{
  "JobId": "ef7289877365fc70299089371c043b96"
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StartIcd10 CmlInferenceJob](#) di Referensi AWS CLI Perintah.

## start-phi-detection-job

Contoh kode berikut menunjukkan cara menggunakan `start-phi-detection-job`.

### AWS CLI

Untuk memulai pekerjaan deteksi PHI

`start-phi-detection-job` Contoh berikut memulai pekerjaan deteksi entitas PHI asinkron.

```
aws comprehendmedical start-phi-detection-job \
  --input-data-config "S3Bucket=comp-med-input" \
  --output-data-config "S3Bucket=comp-med-output" \
  --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
  --language-code en
```

Output:

```
{
  "JobId": "ab9887877365fe70299089371c043b96"
}
```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StartPhiDetectionJob](#) di Referensi AWS CLI Perintah.

## start-rx-norm-inference-job

Contoh kode berikut menunjukkan cara menggunakan `start-rx-norm-inference-job`.

### AWS CLI

Untuk memulai pekerjaan RxNorm inferensi

`start-rx-norm-inference-job` Contoh berikut memulai pekerjaan analisis batch RxNorm inferensi.

```
aws comprehendmedical start-rx-norm-inference-job \
  --input-data-config "S3Bucket=comp-med-input" \
  --output-data-config "S3Bucket=comp-med-output" \
```

```
--data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
--language-code en
```

Output:

```
{
  "JobId": "eg8199877365fc70299089371c043b96"
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StartRxNormInferenceJob](#) di Referensi AWS CLI Perintah.

## start-snomedct-inference-job

Contoh kode berikut menunjukkan cara menggunakan `start-snomedct-inference-job`.

AWS CLI

Untuk memulai pekerjaan inferensi CT SNOMED

`start-snomedct-inference-job` Contoh berikut memulai pekerjaan analisis batch inferensi CT SNOMED.

```
aws comprehendmedical start-snomedct-inference-job \
  --input-data-config "S3Bucket=comp-med-input" \
  --output-data-config "S3Bucket=comp-med-output" \
  --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
  --language-code en
```

Output:

```
{
  "JobId": "dg7289877365fc70299089371c043b96"
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StartSnomedctInferenceJob](#) di Referensi AWS CLI Perintah.

## stop-entities-detection-v2-job

Contoh kode berikut menunjukkan cara menggunakan `stop-entities-detection-v2-job`.

### AWS CLI

Untuk menghentikan pekerjaan deteksi entitas

`stop-entities-detection-v2-job` Contoh berikut menghentikan pekerjaan deteksi entitas asinkron.

```
aws comprehendmedical stop-entities-detection-v2-job \  
  --job-id "ab9887877365fe70299089371c043b96"
```

Output:

```
{  
  "JobId": "ab9887877365fe70299089371c043b96"  
}
```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StopEntitiesDetectionV2Job](#) di Referensi AWS CLI Perintah.

## stop-icd10-cm-inference-job

Contoh kode berikut menunjukkan cara menggunakan `stop-icd10-cm-inference-job`.

### AWS CLI

Untuk menghentikan pekerjaan inferensi ICD-10-CM

`stop-icd10-cm-inference-job` Contoh berikut menghentikan pekerjaan analisis batch inferensi ICD-10-CM.

```
aws comprehendmedical stop-icd10-cm-inference-job \  
  --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Output:



```
{
  "JobId": "ef7289877365fc70299089371c043b96",
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StopIcd10 CmlInferenceJob](#) di Referensi AWS CLI Perintah.

## stop-phi-detection-job

Contoh kode berikut menunjukkan cara menggunakan `stop-phi-detection-job`.

### AWS CLI

Untuk menghentikan pekerjaan deteksi informasi kesehatan yang dilindungi (PHI)

`stop-phi-detection-job` Contoh berikut menghentikan pekerjaan deteksi informasi kesehatan terlindungi asinkron (PHI).

```
aws comprehendmedical stop-phi-detection-job \
  --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Output:

```
{
  "JobId": "ab9887877365fe70299089371c043b96"
}
```

Untuk informasi selengkapnya, lihat [Batch API](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StopPhiDetectionJob](#) di Referensi AWS CLI Perintah.

## stop-rx-norm-inference-job

Contoh kode berikut menunjukkan cara menggunakan `stop-rx-norm-inference-job`.

### AWS CLI

Untuk menghentikan pekerjaan RxNorm inferensi

`stop-rx-norm-inference-job` Contoh berikut menghentikan pekerjaan analisis batch inferensi ICD-10-CM.

```
aws comprehendmedical stop-rx-norm-inference-job \  
  --job-id "eg8199877365fc70299089371c043b96"
```

Output:

```
{  
  "JobId": "eg8199877365fc70299089371c043b96",  
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StopRxNormInferenceJob](#) di Referensi AWS CLI Perintah.

## **stop-snomedct-inference-job**

Contoh kode berikut menunjukkan cara menggunakan `stop-snomedct-inference-job`.

AWS CLI

Untuk menghentikan pekerjaan inferensi CT SNOMED

`stop-snomedct-inference-job` Contoh berikut menghentikan pekerjaan analisis batch inferensi CT SNOMED.

```
aws comprehendmedical stop-snomedct-inference-job \  
  --job-id "8750034166436cdb52ffa3295a1b00a1"
```

Output:

```
{  
  "JobId": "8750034166436cdb52ffa3295a1b00a1",  
}
```

Untuk informasi selengkapnya, lihat [Ontologi yang menghubungkan analisis batch](#) di Amazon Comprehend Medical Developer Guide.

- Untuk detail API, lihat [StopSnomedctInferenceJob](#) di Referensi AWS CLI Perintah.

## AWS Config contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS Config.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-config-rule**

Contoh kode berikut menunjukkan cara menggunakan `delete-config-rule`.

AWS CLI

Untuk menghapus aturan AWS Config

Perintah berikut menghapus aturan AWS Config bernama: `MyConfigRule`

```
aws configservice delete-config-rule --config-rule-name MyConfigRule
```

- Untuk detail API, lihat [DeleteConfigRule](#) di Referensi AWS CLI Perintah.

### **delete-delivery-channel**

Contoh kode berikut menunjukkan cara menggunakan `delete-delivery-channel`.

AWS CLI

Untuk menghapus saluran pengiriman

Perintah berikut menghapus saluran pengiriman default:

```
aws configservice delete-delivery-channel --delivery-channel-name default
```

- Untuk detail API, lihat [DeleteDeliveryChannel](#) di Referensi AWS CLI Perintah.

## **delete-evaluation-results**

Contoh kode berikut menunjukkan cara menggunakan `delete-evaluation-results`.

### AWS CLI

Untuk menghapus hasil evaluasi secara manual

Perintah berikut menghapus hasil evaluasi saat ini untuk aturan AWS terkelola s3: bucket-versioning-enabled

```
aws configservice delete-evaluation-results --config-rule-name s3-bucket-versioning-enabled
```

- Untuk detail API, lihat [DeleteEvaluationResults](#) di Referensi AWS CLI Perintah.

## **deliver-config-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `deliver-config-snapshot`.

### AWS CLI

Untuk memberikan snapshot konfigurasi

Perintah berikut mengirimkan snapshot konfigurasi ke bucket Amazon S3 yang termasuk dalam saluran pengiriman default:

```
aws configservice deliver-config-snapshot --delivery-channel-name default
```

Output:

```
{
  "configSnapshotId": "d0333b00-a683-44af-921e-examplefb794"
}
```

- Untuk detail API, lihat [DeliverConfigSnapshot](#) di Referensi AWS CLI Perintah.

## describe-compliance-by-config-rule

Contoh kode berikut menunjukkan cara menggunakan `describe-compliance-by-config-rule`.

### AWS CLI

Untuk mendapatkan informasi kepatuhan untuk aturan AWS Config Anda

Perintah berikut menampilkan informasi kepatuhan untuk setiap aturan AWS Config yang dilanggar oleh satu atau beberapa sumber daya: AWS

```
aws configservice describe-compliance-by-config-rule --compliance-types
NON_COMPLIANT
```

Dalam output, nilai untuk setiap `CappedCount` atribut menunjukkan berapa banyak sumber daya yang tidak sesuai dengan aturan terkait. Misalnya, output berikut menunjukkan bahwa 3 sumber daya tidak mematuhi aturan bernama `InstanceTypesAreT2micro`.

Output:

```
{
  "ComplianceByConfigRules": [
    {
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 3,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      },
      "ConfigRuleName": "InstanceTypesAreT2micro"
    },
    {
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 10,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      }
    }
  ]
}
```

```

    "ConfigRuleName": "RequiredTagsForVolumes"
  }
]
}

```

- Untuk detail API, lihat [DescribeComplianceByConfigRule](#) di Referensi AWS CLI Perintah.

## describe-compliance-by-resource

Contoh kode berikut menunjukkan cara menggunakan `describe-compliance-by-resource`.

### AWS CLI

Untuk mendapatkan informasi kepatuhan untuk AWS sumber daya Anda

Perintah berikut mengembalikan informasi kepatuhan untuk setiap instans EC2 yang direkam oleh AWS Config dan yang melanggar satu atau beberapa aturan:

```
aws configservice describe-compliance-by-resource --resource-type AWS::EC2::Instance
--compliance-types NON_COMPLIANT
```

Dalam output, nilai untuk setiap `CappedCount` atribut menunjukkan berapa banyak aturan yang dilanggar sumber daya. Misalnya, output berikut menunjukkan bahwa instance `i-1a2b3c4d` melanggar 2 aturan.

Output:

```
{
  "ComplianceByResources": [
    {
      "ResourceType": "AWS::EC2::Instance",
      "ResourceId": "i-1a2b3c4d",
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 2,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      }
    },
    {
      "ResourceType": "AWS::EC2::Instance",
```

```

    "ResourceId": "i-2a2b3c4d ",
    "Compliance": {
      "ComplianceContributorCount": {
        "CappedCount": 3,
        "CapExceeded": false
      },
      "ComplianceType": "NON_COMPLIANT"
    }
  }
]
}

```

- Untuk detail API, lihat [DescribeComplianceByResource](#) di Referensi AWS CLI Perintah.

## describe-config-rule-evaluation-status

Contoh kode berikut menunjukkan cara menggunakan `describe-config-rule-evaluation-status`.

### AWS CLI

Untuk mendapatkan informasi status untuk aturan AWS Config

Perintah berikut mengembalikan informasi status untuk aturan AWS Config bernama: `MyConfigRule`

```
aws configservice describe-config-rule-evaluation-status --config-rule-names
MyConfigRule
```

Output:

```

{
  "ConfigRulesEvaluationStatus": [
    {
      "ConfigRuleArn": "arn:aws:config:us-east-1:123456789012:config-rule/
config-rule-abcdef",
      "FirstActivatedTime": 1450311703.844,
      "ConfigRuleId": "config-rule-abcdef",
      "LastSuccessfulInvocationTime": 1450314643.156,
      "ConfigRuleName": "MyConfigRule"
    }
  ]
}

```

```
}
```

- Untuk detail API, lihat [DescribeConfigRuleEvaluationStatus](#) di Referensi AWS CLI Perintah.

## describe-config-rules

Contoh kode berikut menunjukkan cara menggunakan `describe-config-rules`.

### AWS CLI

Untuk mendapatkan detail tentang aturan AWS Config

Perintah berikut mengembalikan rincian untuk aturan AWS Config bernama:  
`InstanceTypesAreT2micro`

```
aws configservice describe-config-rules --config-rule-names InstanceTypesAreT2micro
```

Output:

```
{
  "ConfigRules": [
    {
      "ConfigRuleState": "ACTIVE",
      "Description": "Evaluates whether EC2 instances are the t2.micro type.",
      "ConfigRuleName": "InstanceTypesAreT2micro",
      "ConfigRuleArn": "arn:aws:config:us-east-1:123456789012:config-rule/
config-rule-abcdef",
      "Source": {
        "Owner": "CUSTOM_LAMBDA",
        "SourceIdentifier": "arn:aws:lambda:us-
east-1:123456789012:function:InstanceTypeCheck",
        "SourceDetails": [
          {
            "EventSource": "aws.config",
            "MessageType": "ConfigurationItemChangeNotification"
          }
        ]
      },
      "InputParameters": "{\"desiredInstanceType\":\"t2.micro\"}",
      "Scope": {
        "ComplianceResourceTypes": [
          "AWS::EC2::Instance"
        ]
      }
    }
  ]
}
```



```

    ],
    "ConfigRuleId": "config-rule-abcdef"
  }
]
}

```

- Untuk detail API, lihat [DescribeConfigRules](#) di Referensi AWS CLI Perintah.

## describe-configuration-recorder-status

Contoh kode berikut menunjukkan cara menggunakan `describe-configuration-recorder-status`.

### AWS CLI

Untuk mendapatkan informasi status untuk perekam konfigurasi

Perintah berikut mengembalikan status perekam konfigurasi default:

```
aws configservice describe-configuration-recorder-status
```

Output:

```

{
  "ConfigurationRecordersStatus": [
    {
      "name": "default",
      "lastStatus": "SUCCESS",
      "recording": true,
      "lastStatusChangeTime": 1452193834.344,
      "lastStartTime": 1441039997.819,
      "lastStopTime": 1441039992.835
    }
  ]
}

```

- Untuk detail API, lihat [DescribeConfigurationRecorderStatus](#) di Referensi AWS CLI Perintah.

## describe-configuration-recorders

Contoh kode berikut menunjukkan cara menggunakan `describe-configuration-recorders`.

## AWS CLI

Untuk mendapatkan detail tentang perekam konfigurasi

Perintah berikut mengembalikan rincian tentang perekam konfigurasi default:

```
aws configservice describe-configuration-records
```

Output:

```
{
  "ConfigurationRecorders": [
    {
      "recordingGroup": {
        "allSupported": true,
        "resourceTypes": [],
        "includeGlobalResourceTypes": true
      },
      "roleARN": "arn:aws:iam::123456789012:role/config-ConfigRole-
A1B2C3D4E5F6",
      "name": "default"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeConfigurationRecorders](#) di Referensi AWS CLI Perintah.

## **describe-delivery-channel-status**

Contoh kode berikut menunjukkan cara menggunakan `describe-delivery-channel-status`.

## AWS CLI

Untuk mendapatkan informasi status untuk saluran pengiriman

Perintah berikut mengembalikan status saluran pengiriman:

```
aws configservice describe-delivery-channel-status
```

Output:

```
{
  "DeliveryChannelsStatus": [
    {
      "configStreamDeliveryInfo": {
        "lastStatusChangeTime": 1452193834.381,
        "lastStatus": "SUCCESS"
      },
      "configHistoryDeliveryInfo": {
        "lastSuccessfulTime": 1450317838.412,
        "lastStatus": "SUCCESS",
        "lastAttemptTime": 1450317838.412
      },
      "configSnapshotDeliveryInfo": {
        "lastSuccessfulTime": 1452185597.094,
        "lastStatus": "SUCCESS",
        "lastAttemptTime": 1452185597.094
      },
      "name": "default"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeDeliveryChannelStatus](#) di Referensi AWS CLI Perintah.

## describe-delivery-channels

Contoh kode berikut menunjukkan cara menggunakan `describe-delivery-channels`.

### AWS CLI

Untuk mendapatkan detail tentang saluran pengiriman

Perintah berikut mengembalikan rincian tentang saluran pengiriman:

```
aws configservice describe-delivery-channels
```

Output:

```
{
  "DeliveryChannels": [
    {
      "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
```

```
        "name": "default",
        "s3BucketName": "config-bucket-123456789012"
    }
]
}
```

- Untuk detail API, lihat [DescribeDeliveryChannels](#) di Referensi AWS CLI Perintah.

## get-compliance-details-by-config-rule

Contoh kode berikut menunjukkan cara menggunakan `get-compliance-details-by-config-rule`.

### AWS CLI

Untuk mendapatkan hasil evaluasi untuk aturan AWS Config

Perintah berikut mengembalikan hasil evaluasi untuk semua sumber daya yang tidak sesuai dengan aturan AWS Config bernama: `InstanceTypesAreT2micro`

```
aws configservice get-compliance-details-by-config-rule --config-rule-name
InstanceTypesAreT2micro --compliance-types NON_COMPLIANT
```

### Output:

```
{
  "EvaluationResults": [
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
          "ResourceType": "AWS::EC2::Instance",
          "ResourceId": "i-1a2b3c4d",
          "ConfigRuleName": "InstanceTypesAreT2micro"
        }
      },
      "ResultRecordedTime": 1450314645.261,
      "ConfigRuleInvokedTime": 1450314642.948,
      "ComplianceType": "NON_COMPLIANT"
    },
    {
      "EvaluationResultIdentifier": {
```

```

    "OrderingTimestamp": 1450314635.065,
    "EvaluationResultQualifier": {
      "ResourceType": "AWS::EC2::Instance",
      "ResourceId": "i-2a2b3c4d",
      "ConfigRuleName": "InstanceTypesAreT2micro"
    }
  },
  "ResultRecordedTime": 1450314645.18,
  "ConfigRuleInvokedTime": 1450314642.902,
  "ComplianceType": "NON_COMPLIANT"
},
{
  "EvaluationResultIdentifier": {
    "OrderingTimestamp": 1450314635.065,
    "EvaluationResultQualifier": {
      "ResourceType": "AWS::EC2::Instance",
      "ResourceId": "i-3a2b3c4d",
      "ConfigRuleName": "InstanceTypesAreT2micro"
    }
  },
  "ResultRecordedTime": 1450314643.346,
  "ConfigRuleInvokedTime": 1450314643.124,
  "ComplianceType": "NON_COMPLIANT"
}
]
}

```

- Untuk detail API, lihat [GetComplianceDetailsByConfigRule](#) di Referensi AWS CLI Perintah.

## get-compliance-details-by-resource

Contoh kode berikut menunjukkan cara menggunakan `get-compliance-details-by-resource`.

### AWS CLI

Untuk mendapatkan hasil evaluasi untuk sumber AWS daya

Perintah berikut mengembalikan hasil evaluasi untuk setiap aturan yang `i-1a2b3c4d` tidak dipatuhi oleh instans EC2:

```
aws configservice get-compliance-details-by-resource --resource-type
AWS::EC2::Instance --resource-id i-1a2b3c4d --compliance-types NON_COMPLIANT
```

**Output:**

```
{
  "EvaluationResults": [
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
          "ResourceType": "AWS::EC2::Instance",
          "ResourceId": "i-1a2b3c4d",
          "ConfigRuleName": "InstanceTypesAreT2micro"
        }
      },
      "ResultRecordedTime": 1450314643.288,
      "ConfigRuleInvokedTime": 1450314643.034,
      "ComplianceType": "NON_COMPLIANT"
    },
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
          "ResourceType": "AWS::EC2::Instance",
          "ResourceId": "i-1a2b3c4d",
          "ConfigRuleName": "RequiredTagForEC2Instances"
        }
      },
      "ResultRecordedTime": 1450314645.261,
      "ConfigRuleInvokedTime": 1450314642.948,
      "ComplianceType": "NON_COMPLIANT"
    }
  ]
}
```

- Untuk detail API, lihat [GetComplianceDetailsByResource](#) di Referensi AWS CLI Perintah.

**get-compliance-summary-by-config-rule**

Contoh kode berikut menunjukkan cara menggunakan `get-compliance-summary-by-config-rule`.

**AWS CLI**

Untuk mendapatkan ringkasan kepatuhan untuk aturan AWS Config Anda

Perintah berikut mengembalikan jumlah aturan yang sesuai dan nomor yang tidak sesuai:

```
aws configservice get-compliance-summary-by-config-rule
```

Dalam output, nilai untuk setiap CappedCount atribut menunjukkan berapa banyak aturan yang sesuai atau tidak sesuai.

Output:

```
{
  "ComplianceSummary": {
    "NonCompliantResourceCount": {
      "CappedCount": 3,
      "CapExceeded": false
    },
    "ComplianceSummaryTimestamp": 1452204131.493,
    "CompliantResourceCount": {
      "CappedCount": 2,
      "CapExceeded": false
    }
  }
}
```

- Untuk detail API, lihat [GetComplianceSummaryByConfigRule](#) di Referensi AWS CLI Perintah.

## get-compliance-summary-by-resource-type

Contoh kode berikut menunjukkan cara menggunakan `get-compliance-summary-by-resource-type`.

AWS CLI

Untuk mendapatkan ringkasan kepatuhan untuk semua jenis sumber daya

Perintah berikut mengembalikan jumlah AWS sumber daya yang tidak sesuai dan nomor yang sesuai:

```
aws configservice get-compliance-summary-by-resource-type
```

Dalam output, nilai untuk setiap CappedCount atribut menunjukkan berapa banyak sumber daya yang sesuai atau tidak sesuai.

**Output:**

```
{
  "ComplianceSummariesByResourceType": [
    {
      "ComplianceSummary": {
        "NonCompliantResourceCount": {
          "CappedCount": 16,
          "CapExceeded": false
        },
        "ComplianceSummaryTimestamp": 1453237464.543,
        "CompliantResourceCount": {
          "CappedCount": 10,
          "CapExceeded": false
        }
      }
    }
  ]
}
```

Untuk mendapatkan ringkasan kepatuhan untuk jenis sumber daya tertentu

Perintah berikut mengembalikan jumlah instans EC2 yang tidak sesuai dan nomor yang sesuai:

```
aws configservice get-compliance-summary-by-resource-type --resource-types
AWS::EC2::Instance
```

Dalam output, nilai untuk setiap CappedCount atribut menunjukkan berapa banyak sumber daya yang sesuai atau tidak sesuai.

**Output:**

```
{
  "ComplianceSummariesByResourceType": [
    {
      "ResourceType": "AWS::EC2::Instance",
      "ComplianceSummary": {
        "NonCompliantResourceCount": {
          "CappedCount": 3,
          "CapExceeded": false
        },
        "ComplianceSummaryTimestamp": 1452204923.518,

```



```

    "CompliantResourceCount": {
      "CappedCount": 7,
      "CapExceeded": false
    }
  }
]
}

```

- Untuk detail API, lihat [GetComplianceSummaryByResourceType](#) di Referensi AWS CLI Perintah.

## get-resource-config-history

Contoh kode berikut menunjukkan cara menggunakan `get-resource-config-history`.

### AWS CLI

Untuk mendapatkan riwayat konfigurasi sumber AWS daya

Perintah berikut mengembalikan daftar item konfigurasi untuk instance EC2 dengan ID: `i-1a2b3c4d`

```
aws configservice get-resource-config-history --resource-type AWS::EC2::Instance --resource-id i-1a2b3c4d
```

- Untuk detail API, lihat [GetResourceConfigHistory](#) di Referensi AWS CLI Perintah.

## get-status

Contoh kode berikut menunjukkan cara menggunakan `get-status`.

### AWS CLI

Untuk mendapatkan status untuk AWS Config

Perintah berikut mengembalikan status saluran pengiriman dan perekam konfigurasi:

```
aws configservice get-status
```

Output:

**Configuration Recorders:**

```
name: default
recorder: ON
last status: SUCCESS
```

**Delivery Channels:**

```
name: default
last stream delivery status: SUCCESS
last history delivery status: SUCCESS
last snapshot delivery status: SUCCESS
```

- Untuk detail API, lihat [GetStatus](#) di Referensi AWS CLI Perintah.

**list-discovered-resources**

Contoh kode berikut menunjukkan cara menggunakan `list-discovered-resources`.

**AWS CLI**

Untuk membuat daftar sumber daya yang telah AWS ditemukan Config

Perintah berikut mencantumkan instance EC2 yang telah ditemukan AWS Config:

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Instance
```

**Output:**

```
{
  "resourceIdentifiers": [
    {
      "resourceType": "AWS::EC2::Instance",
      "resourceId": "i-1a2b3c4d"
    },
    {
      "resourceType": "AWS::EC2::Instance",
      "resourceId": "i-2a2b3c4d"
    },
    {
      "resourceType": "AWS::EC2::Instance",
      "resourceId": "i-3a2b3c4d"
    }
  ]
}
```

```

    }
  ]
}

```

- Untuk detail API, lihat [ListDiscoveredResources](#) di Referensi AWS CLI Perintah.

## put-config-rule

Contoh kode berikut menunjukkan cara menggunakan `put-config-rule`.

### AWS CLI

Untuk menambahkan aturan Config AWS terkelola

Perintah berikut menyediakan kode JSON untuk menambahkan aturan Config AWS terkelola:

```
aws configservice put-config-rule --config-rule file://
RequiredTagsForEC2Instances.json
```

`RequiredTagsForEC2Instances.json` adalah file JSON yang berisi konfigurasi aturan:

```
{
  "ConfigRuleName": "RequiredTagsForEC2Instances",
  "Description": "Checks whether the CostCenter and Owner tags are applied to EC2
instances.",
  "Scope": {
    "ComplianceResourceTypes": [
      "AWS::EC2::Instance"
    ]
  },
  "Source": {
    "Owner": "AWS",
    "SourceIdentifier": "REQUIRED_TAGS"
  },
  "InputParameters": "{\"tag1Key\":\"CostCenter\",\"tag2Key\":\"Owner\"}"
}
```

Untuk `ComplianceResourceTypes` atribut, kode JSON ini membatasi cakupan sumber daya AWS::EC2::Instance tipe, jadi AWS Config hanya akan mengevaluasi instans EC2 terhadap aturan. Karena aturan adalah aturan terkelola, `Owner` atribut diatur ke `AWS`, dan `SourceIdentifier` atribut diatur ke pengidentifikasi aturan, `REQUIRED_TAGS`. Untuk

`InputParameters` atribut, kunci tag yang dibutuhkan aturan, `CostCenter` dan `Owner`, ditentukan.

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi konfigurasi aturan, jalankan `describe-config-rules` perintah, dan tentukan nama aturan.

Untuk menambahkan aturan Config terkelola pelanggan

Perintah berikut menyediakan kode JSON untuk menambahkan aturan Config terkelola pelanggan:

```
aws configservice put-config-rule --config-rule file://InstanceTypesAreT2micro.json
```

`InstanceTypesAreT2micro.json` adalah file JSON yang berisi konfigurasi aturan:

```
{
  "ConfigRuleName": "InstanceTypesAreT2micro",
  "Description": "Evaluates whether EC2 instances are the t2.micro type.",
  "Scope": {
    "ComplianceResourceTypes": [
      "AWS::EC2::Instance"
    ]
  },
  "Source": {
    "Owner": "CUSTOM_LAMBDA",
    "SourceIdentifier": "arn:aws:lambda:us-east-1:123456789012:function:InstanceTypeCheck",
    "SourceDetails": [
      {
        "EventSource": "aws.config",
        "MessageType": "ConfigurationItemChangeNotification"
      }
    ]
  },
  "InputParameters": "{\"desiredInstanceType\":\"t2.micro\"}"
}
```

Untuk `ComplianceResourceTypes` atribut, kode JSON ini membatasi cakupan sumber daya `AWS::EC2::Instance` tipe, jadi AWS Config hanya akan mengevaluasi instans EC2 terhadap aturan. Karena aturan ini adalah aturan yang dikelola pelanggan, `Owner` atribut disetel ke `CUSTOM_LAMBDA`, dan `SourceIdentifier` atribut disetel ke ARN dari fungsi Lambda AWS. `SourceDetails` objek diperlukan. Parameter yang ditentukan untuk `InputParameters` atribut

diteruskan ke fungsi AWS Lambda saat AWS Config memanggilnya untuk mengevaluasi sumber daya terhadap aturan.

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi konfigurasi aturan, jalankan `describe-config-rules` perintah, dan tentukan nama aturan.

- Untuk detail API, lihat [PutConfigRule](#) di Referensi AWS CLI Perintah.

## put-configuration-recorder

Contoh kode berikut menunjukkan cara menggunakan `put-configuration-recorder`.

### AWS CLI

Contoh 1: Untuk merekam semua sumber daya yang didukung

Perintah berikut membuat perekam konfigurasi yang melacak perubahan pada semua jenis sumber daya yang didukung, termasuk tipe sumber daya global:

```
aws configservice put-configuration-recorder \  
  --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/  
config-role \  
  --recording-group allSupported=true,includeGlobalResourceTypes=true
```

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi pengaturan perekam konfigurasi Anda, jalankan `describe-configuration-recorders` perintah.

Contoh 2: Untuk merekam jenis sumber daya tertentu

Perintah berikut membuat perekam konfigurasi yang melacak perubahan hanya pada jenis sumber daya yang ditentukan dalam file JSON untuk opsi `--recording-group`:

```
aws configservice put-configuration-recorder \  
  --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/  
config-role \  
  --recording-group file://recordingGroup.json
```

`RecordingGroup.json` adalah file JSON yang menentukan jenis sumber daya yang akan direkam oleh Config: AWS

```
{  
  "allSupported": false,
```

```
"includeGlobalResourceTypes": false,
"resourceTypes": [
  "AWS::EC2::EIP",
  "AWS::EC2::Instance",
  "AWS::EC2::NetworkAcl",
  "AWS::EC2::SecurityGroup",
  "AWS::CloudTrail::Trail",
  "AWS::EC2::Volume",
  "AWS::EC2::VPC",
  "AWS::IAM::User",
  "AWS::IAM::Policy"
]
}
```

Sebelum Anda dapat menentukan jenis sumber daya untuk kunci `ResourceTypes`, Anda harus menyetel opsi `AllSupported` `includeGlobalResource` dan `Types` ke `false` atau menghilangkannya.

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi pengaturan perekam konfigurasi Anda, jalankan `describe-configuration-records` perintah.

Contoh 3: Untuk memilih semua sumber daya yang didukung kecuali jenis sumber daya tertentu

Perintah berikut membuat perekam konfigurasi yang melacak perubahan pada semua jenis sumber daya yang didukung saat ini dan masa depan tidak termasuk jenis sumber daya yang ditentukan dalam file JSON untuk opsi `--recording-group`:

```
aws configservice put-configuration-recorder \
  --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/
config-role \
  --recording-group file://recordingGroup.json
```

`RecordingGroup.json` adalah file JSON yang menentukan jenis sumber daya yang akan direkam oleh Config: AWS

```
{
  "allSupported": false,
  "exclusionByResourceTypes": {
    "resourceTypes": [
      "AWS::Redshift::ClusterSnapshot",
      "AWS::RDS::DBClusterSnapshot",
      "AWS::CloudFront::StreamingDistribution"
    ]
  }
}
```

```
    },
    "includeGlobalResourceTypes": false,
    "recordingStrategy": {
      "useOnly": "EXCLUSION_BY_RESOURCE_TYPES"
    },
  },
}
```

Sebelum Anda dapat menentukan jenis sumber daya untuk dikecualikan dari perekaman:

1) Anda harus menyetel opsi `AllSupported` dan `includeGlobalResourceTypes` menjadi `false` atau menghilangkannya, dan 2) Anda harus menyetel bidang `useOnly` ke `EXCLUSION_BY_RESOURCE_TYPES`. `RecordingStrategy`

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi pengaturan perekam konfigurasi Anda, jalankan `describe-configuration-records` perintah.

- Untuk detail API, lihat [PutConfigurationRecorder](#) di Referensi AWS CLI Perintah.

## put-delivery-channel

Contoh kode berikut menunjukkan cara menggunakan `put-delivery-channel`.

### AWS CLI

Untuk membuat saluran pengiriman

Perintah berikut menyediakan pengaturan untuk saluran pengiriman sebagai kode JSON:

```
aws configservice put-delivery-channel --delivery-channel file://
deliveryChannel.json
```

`deliveryChannel.json` menentukan atribut saluran pengiriman:

```
{
  "name": "default",
  "s3BucketName": "config-bucket-123456789012",
  "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
  "configSnapshotDeliveryProperties": {
    "deliveryFrequency": "Twelve_Hours"
  }
}
```

Contoh ini menetapkan atribut berikut:

`name`- Nama saluran pengiriman. Secara default, AWS Config menetapkan nama default ke saluran pengiriman baru. Anda tidak dapat memperbarui nama saluran pengiriman dengan perintah `put-delivery-channel`. Untuk langkah-langkah mengubah nama, lihat [Mengganti nama Saluran Pengiriman](#). `s3BucketName` - Nama bucket Amazon S3 tempat AWS Config memberikan snapshot konfigurasi dan file riwayat konfigurasi. Jika Anda menentukan bucket milik AWS akun lain, bucket tersebut harus memiliki kebijakan yang memberikan izin akses ke Config. AWS Untuk informasi selengkapnya, lihat [Izin untuk Bucket Amazon S3](#).

`snsTopicARN`- Nama Sumber Daya Amazon (ARN) dari topik Amazon SNS tempat AWS Config mengirimkan pemberitahuan tentang perubahan konfigurasi. Jika Anda memilih topik dari akun lain, topik tersebut harus memiliki kebijakan yang memberikan izin akses ke Config. AWS Untuk informasi selengkapnya, lihat [Izin untuk Topik Amazon SNS](#).

`configSnapshotDeliveryProperties`- Berisi `deliveryFrequency` atribut, yang menetapkan seberapa sering AWS Config memberikan snapshot konfigurasi dan seberapa sering ia memanggil evaluasi untuk aturan Config periodik.

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi pengaturan saluran pengiriman Anda, jalankan `describe-delivery-channels` perintah.

- Untuk detail API, lihat [PutDeliveryChannel](#) di Referensi AWS CLI Perintah.

## **start-config-rules-evaluation**

Contoh kode berikut menunjukkan cara menggunakan `start-config-rules-evaluation`.

### AWS CLI

Untuk menjalankan evaluasi sesuai permintaan untuk aturan AWS Config

Perintah berikut memulai evaluasi untuk dua aturan AWS terkelola:

```
aws configservice start-config-rules-evaluation --config-rule-names s3-bucket-versioning-enabled cloudtrail-enabled
```

- Untuk detail API, lihat [StartConfigRulesEvaluation](#) di Referensi AWS CLI Perintah.

## **start-configuration-recorder**

Contoh kode berikut menunjukkan cara menggunakan `start-configuration-recorder`.



## AWS CLI

Untuk memulai perekam konfigurasi

Perintah berikut memulai perekam konfigurasi default:

```
aws configservice start-configuration-recorder --configuration-recorder-name default
```

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi bahwa AWS Config merekam sumber daya Anda, jalankan perintah `get-status`.

- Untuk detail API, lihat [StartConfigurationRecorder](#) di Referensi AWS CLI Perintah.

## **stop-configuration-recorder**

Contoh kode berikut menunjukkan cara menggunakan `stop-configuration-recorder`.

## AWS CLI

Untuk menghentikan perekam konfigurasi

Perintah berikut menghentikan perekam konfigurasi default:

```
aws configservice stop-configuration-recorder --configuration-recorder-name default
```

Jika perintah berhasil, AWS Config tidak mengembalikan output. Untuk memverifikasi bahwa AWS Config tidak merekam sumber daya Anda, jalankan perintah `get-status`.

- Untuk detail API, lihat [StopConfigurationRecorder](#) di Referensi AWS CLI Perintah.

## **subscribe**

Contoh kode berikut menunjukkan cara menggunakan `subscribe`.

## AWS CLI

Untuk berlangganan AWS Config

Perintah berikut membuat saluran pengiriman default dan perekam konfigurasi. Perintah ini juga menentukan bucket Amazon S3 dan topik Amazon SNS tempat AWS Config akan memberikan informasi konfigurasi:

```
aws configservice subscribe --s3-bucket config-bucket-123456789012 --
sns-topic arn:aws:sns:us-east-1:123456789012:config-topic --iam-role
arn:aws:iam::123456789012:role/ConfigRole-A1B2C3D4E5F6
```

### Output:

```
Using existing S3 bucket: config-bucket-123456789012
Using existing SNS topic: arn:aws:sns:us-east-1:123456789012:config-topic
Subscribe succeeded:

Configuration Recorders: [
  {
    "recordingGroup": {
      "allSupported": true,
      "resourceTypes": [],
      "includeGlobalResourceTypes": false
    },
    "roleARN": "arn:aws:iam::123456789012:role/ConfigRole-A1B2C3D4E5F6",
    "name": "default"
  }
]

Delivery Channels: [
  {
    "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
    "name": "default",
    "s3BucketName": "config-bucket-123456789012"
  }
]
```

- Untuk detail API, lihat [Berlangganan](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Connect menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum menggunakan AWS Command Line Interface with Amazon Connect.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-user**

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

AWS CLI

Untuk membuat pengguna

`create-user` Contoh berikut menambahkan pengguna dengan atribut yang ditentukan ke instance Amazon Connect yang ditentukan.

```
aws connect create-user \  
  --username Mary \  
  --password Pass@Word1 \  
  --identity-info FirstName=Mary,LastName=Major \  
  --phone-config  
  PhoneType=DESK_PHONE,AutoAccept=true,AfterContactWorkTimeLimit=60,DeskPhoneNumber=  
+15555551212 \  
  --security-profile-id 12345678-1111-2222-aaaa-a1b2c3d4f5g7 \  
  --routing-profile-id 87654321-9999-3434-abcd-x1y2z3a1b2c3 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "UserId": "87654321-2222-1234-1234-111234567891",  
  "UserArn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-  
cdef-EXAMPLE11111/agent/87654321-2222-1234-1234-111234567891"  
}
```

Untuk informasi selengkapnya, lihat [Menambahkan Pengguna](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## **delete-user**

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

### AWS CLI

Untuk menghapus pengguna

`delete-user` Contoh berikut menghapus pengguna tertentu dari instance Amazon Connect yang ditentukan.

```
aws connect delete-user \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --user-id 87654321-2222-1234-1234-111234567891
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Pengguna](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## **describe-user-hierarchy-group**

Contoh kode berikut menunjukkan cara menggunakan `describe-user-hierarchy-group`.

### AWS CLI

Untuk menampilkan detail untuk grup hierarki

`describe-user-hierarchy-group` Contoh berikut menampilkan detail untuk grup hierarki Amazon Connect yang ditentukan.

```
aws connect describe-user-hierarchy-group \  
  --hierarchy-group-id 12345678-1111-2222-800e-aaabbb555gg \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

**Output:**

```
{
  "HierarchyGroup": {
    "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group/12345678-1111-2222-800e-a2b3c4d5f6g7",
    "Name": "Example Corporation",
    "LevelId": "1",
    "HierarchyPath": {
      "LevelOne": {
        "Id": "abcdefgh-3333-4444-8af3-201123456789",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group/abcdefgh-3333-4444-8af3-201123456789",
        "Name": "Example Corporation"
      }
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Mengatur Hierarki Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [DescribeUserHierarchyGroup](#) di Referensi AWS CLI Perintah.

**describe-user-hierarchy-structure**

Contoh kode berikut menunjukkan cara menggunakan `describe-user-hierarchy-structure`.

**AWS CLI**

Untuk menampilkan detail untuk struktur hierarki

`describe-user-hierarchy-structure` Contoh berikut menampilkan detail untuk struktur hierarki untuk instance Amazon Connect yang ditentukan.

```
aws connect describe-user-hierarchy-group \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

**Output:**

```
{
  "HierarchyStructure": {
    "LevelOne": {
      "Id": "12345678-1111-2222-800e-aaabbb555gg",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/1",
      "Name": "Corporation"
    },
    "LevelTwo": {
      "Id": "87654321-2222-3333-ac99-123456789102",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/2",
      "Name": "Services Division"
    },
    "LevelThree": {
      "Id": "abcdefgh-3333-4444-8af3-201123456789",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/3",
      "Name": "EU Site"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Mengatur Hierarki Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [DescribeUserHierarchyStructure](#) di Referensi AWS CLI Perintah.

## describe-user

Contoh kode berikut menunjukkan cara menggunakan `describe-user`.

### AWS CLI

Untuk menampilkan detail untuk pengguna

`describe-user` Contoh berikut menampilkan detail untuk pengguna Amazon Connect yang ditentukan.

```
aws connect describe-user \
  --user-id 0c245dc0-0cf5-4e37-800e-2a7481cc8a60
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

**Output:**

```
{
  "User": {
    "Id": "0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
    "Username": "Jane",
    "IdentityInfo": {
      "FirstName": "Jane",
      "LastName": "Doe",
      "Email": "example.com"
    },
    "PhoneConfig": {
      "PhoneType": "SOFT_PHONE",
      "AutoAccept": false,
      "AfterContactWorkTimeLimit": 0,
      "DeskPhoneNumber": ""
    },
    "DirectoryUserId": "8b444cf6-b368-4f29-ba18-07af27405658",
    "SecurityProfileIds": [
      "b6f85a42-1dc5-443b-b621-de0abf70c9cf"
    ],
    "RoutingProfileId": "0be36ee9-2b5f-4ef4-bcf7-87738e5be0e5",
    "Tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengguna](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [DescribeUser](#) di Referensi AWS CLI Perintah.

**get-contact-attributes**

Contoh kode berikut menunjukkan cara menggunakan `get-contact-attributes`.

**AWS CLI**

Untuk mengambil atribut untuk kontak

`get-contact-attributes` Contoh berikut mengambil atribut yang ditetapkan untuk kontak Amazon Connect yang ditentukan.

```
aws connect get-contact-attributes \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --initial-contact-id 12345678-1111-2222-800e-a2b3c4d5f6g7
```

Output:

```
{  
  "Attributes": {  
    "greetingPlayed": "true"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Atribut Kontak Amazon Connect](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [GetContactAttributes](#) di Referensi AWS CLI Perintah.

## list-contact-flows

Contoh kode berikut menunjukkan cara menggunakan `list-contact-flows`.

AWS CLI

Untuk membuat daftar alur kontak dalam sebuah instance

`list-contact-flows` Contoh berikut mencantumkan alur kontak dalam instance Amazon Connect yang ditentukan.

```
aws connect list-contact-flows \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "ContactFlowSummaryList": [  
    {  
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/12345678-1111-2222-800e-  
a2b3c4d5f6g7",  
    }  
  ]  
}
```



```

        "Name": "Default queue transfer",
        "ContactFlowType": "QUEUE_TRANSFER"
    },
    {
        "Id": "87654321-2222-3333-ac99-123456789102",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/87654321-2222-3333-
ac99-123456789102",
        "Name": "Default agent hold",
        "ContactFlowType": "AGENT_HOLD"
    },
    {
        "Id": "abcdefgh-3333-4444-8af3-201123456789",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/
abcdefgh-3333-4444-8af3-201123456789",
        "Name": "Default customer hold",
        "ContactFlowType": "CUSTOMER_HOLD"
    },
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Alur Kontak Amazon Connect](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListContactFlows](#) di Referensi AWS CLI Perintah.

## list-hours-of-operations

Contoh kode berikut menunjukkan cara menggunakan `list-hours-of-operations`.

### AWS CLI

Untuk membuat daftar jam operasi dalam sebuah instance

`list-hours-of-operations` Contoh berikut mencantumkan jam operasi untuk instans Amazon Connect yang ditentukan.

```
aws connect list-hours-of-operations \
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
  "HoursOfOperationSummaryList": [
    {
      "Id": "d69f1f84-7457-4924-8fbe-e64875546259",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/operating-hours/d69f1f84-7457-4924-8fbe-e64875546259",
      "Name": "Basic Hours"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengatur Jam Operasi untuk Antrian](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListHoursOfOperations](#) di Referensi AWS CLI Perintah.

## list-phone-numbers

Contoh kode berikut menunjukkan cara menggunakan `list-phone-numbers`.

### AWS CLI

Untuk mencantumkan nomor telepon dalam sebuah contoh

`list-phone-numbers` Contoh berikut mencantumkan nomor telepon dalam instans Amazon Connect yang ditentukan.

```
aws connect list-phone-numbers \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

### Output:

```
{
  "PhoneNumberSummaryList": [
    {
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/phone-number/xyz80zxy-xyz1-80zx-
zx80-11111EXAMPLE",
      "PhoneNumber": "+17065551212",
      "PhoneNumberType": "DID",
    }
  ]
}
```

```

        "PhoneNumberCountryCode": "US"
    },
    {
        "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/phone-number/ccc0ccc-xyz1-80zx-
zx80-22222EXAMPLE",
        "PhoneNumber": "+18555551212",
        "PhoneNumberType": "TOLL_FREE",
        "PhoneNumberCountryCode": "US"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Mengatur Nomor Telepon untuk Pusat Kontak Anda](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListPhoneNumbers](#) di Referensi AWS CLI Perintah.

## list-queues

Contoh kode berikut menunjukkan cara menggunakan `list-queues`.

### AWS CLI

Untuk membuat daftar antrian dalam sebuah instance

`list-queues` Contoh berikut mencantumkan antrian dalam instance Amazon Connect yang ditentukan.

```

aws connect list-queues \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Output:

```

{
  "QueueSummaryList": [
    {
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/12345678-1111-2222-800e-
a2b3c4d5f6g7",

```

```

        "QueueType": "AGENT"
    },
    {
        "Id": "87654321-2222-3333-ac99-123456789102",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/87654321-2222-3333-
ac99-123456789102",
        "QueueType": "AGENT"
    },
    {
        "Id": "abcdefgh-3333-4444-8af3-201123456789",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/
abcdefgh-3333-4444-8af3-201123456789",
        "QueueType": "AGENT"
    },
    {
        "Id": "hgfedcba-4444-5555-a31f-123456789102",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/hgfedcba-4444-5555-a31f-123456789102",
        "Name": "BasicQueue",
        "QueueType": "STANDARD"
    },
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Antrian](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListQueues](#) di Referensi AWS CLI Perintah.

## list-routing-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-routing-profiles`.

### AWS CLI

Untuk membuat daftar profil perutean dalam sebuah instance

`list-routing-profiles` Contoh berikut mencantumkan profil perutean dalam instance Amazon Connect yang ditentukan.

```

aws connect list-routing-profiles \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

**Output:**

```
{
  "RoutingProfileSummaryList": [
    {
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/routing-profile/12345678-1111-2222-800e-
a2b3c4d5f6g7",
      "Name": "Basic Routing Profile"
    },
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Profil Perutean](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListRoutingProfiles](#) di Referensi AWS CLI Perintah.

**list-security-profiles**

Contoh kode berikut menunjukkan cara menggunakan `list-security-profiles`.

**AWS CLI**

Untuk membuat daftar profil keamanan dalam sebuah instance

`list-security-profiles` Contoh berikut mencantumkan profil keamanan dalam instans Amazon Connect yang ditentukan.

```
aws connect list-security-profiles \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

**Output:**

```
{
  "SecurityProfileSummaryList": [
    {
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/12345678-1111-2222-800e-
a2b3c4d5f6g7",
    },
  ]
}
```

```

        "Name": "CallCenterManager"
    },
    {
        "Id": "87654321-2222-3333-ac99-123456789102",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/87654321-2222-3333-
ac99-123456789102",
        "Name": "QualityAnalyst"
    },
    {
        "Id": "abcdefgh-3333-4444-8af3-201123456789",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/
abcdefgh-3333-4444-8af3-201123456789",
        "Name": "Agent"
    },
    {
        "Id": "12345678-1111-2222-800e-x2y3c4d5fzzzz",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/12345678-1111-2222-800e-
x2y3c4d5fzzzz",
        "Name": "Admin"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menetapkan Izin: Profil Keamanan](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListSecurityProfiles](#) di Referensi AWS CLI Perintah.

## list-user-hierarchy-groups

Contoh kode berikut menunjukkan cara menggunakan `list-user-hierarchy-groups`.

### AWS CLI

Untuk mencantumkan grup hierarki pengguna dalam sebuah instance

`list-user-hierarchy-groups` Contoh berikut mencantumkan grup hierarki pengguna dalam instance Amazon Connect yang ditentukan.

```
aws connect list-user-hierarchy-groups \
```

```
--instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
  "UserHierarchyGroupSummaryList": [
    {
      "Id": "0e2f6d1d-b3ca-494b-8dbc-ba81d9f8182a",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent-group/0e2f6d1d-b3ca-494b-8dbc-ba81d9f8182a",
      "Name": "Example Corporation"
    },
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengatur Hierarki Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListUserHierarchyGroups](#) di Referensi AWS CLI Perintah.

## list-users

Contoh kode berikut menunjukkan cara menggunakan `list-users`.

### AWS CLI

Untuk mencantumkan grup hierarki pengguna dalam sebuah instance

`list-users` Contoh berikut mencantumkan pengguna dalam instans Amazon Connect yang ditentukan.

```
aws connect list-users \
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
  "UserSummaryList": [
    {
      "Id": "0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
```

```

    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
    ea62-414c-97bb-d018e39e158e/agent/0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
    "Username": "Jane"
  },
  {
    "Id": "46f0c67c-3fc7-4806-ac99-403798788c14",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
    ea62-414c-97bb-d018e39e158e/agent/46f0c67c-3fc7-4806-ac99-403798788c14",
    "Username": "Paulo"
  },
  {
    "Id": "55a83578-95e1-4710-8af3-2b7afe310e48",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
    ea62-414c-97bb-d018e39e158e/agent/55a83578-95e1-4710-8af3-2b7afe310e48",
    "Username": "JohnD"
  },
  {
    "Id": "703e27b5-c9f0-4f1f-a239-64ccbb160125",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
    ea62-414c-97bb-d018e39e158e/agent/703e27b5-c9f0-4f1f-a239-64ccbb160125",
    "Username": "JohnS"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menambahkan Pengguna](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [ListUsers](#) di Referensi AWS CLI Perintah.

## update-contact-attributes

Contoh kode berikut menunjukkan cara menggunakan `update-contact-attributes`.

### AWS CLI

Untuk memperbarui atribut kontak

`update-contact-attributes` Contoh berikut memperbarui `greetingPlayed` atribut untuk pengguna Amazon Connect yang ditentukan.

```

aws connect update-contact-attributes \
  --initial-contact-id 11111111-2222-3333-4444-12345678910 \

```



```
--instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
--attributes greetingPlayed=false
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan Atribut Kontak Amazon Connect](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [UpdateContactAttributes](#) di Referensi AWS CLI Perintah.

## update-user-hierarchy

Contoh kode berikut menunjukkan cara menggunakan `update-user-hierarchy`.

### AWS CLI

Untuk memperbarui hierarki pengguna

`update-user-hierarchy` Contoh berikut memperbarui hierarki agen untuk pengguna Amazon Connect yang ditentukan.

```
aws connect update-user-hierarchy \  
  --hierarchy-group-id 12345678-a1b2-c3d4-e5f6-123456789abc \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengkonfigurasi Pengaturan Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [UpdateUserHierarchy](#) di Referensi AWS CLI Perintah.

## update-user-identity-info

Contoh kode berikut menunjukkan cara menggunakan `update-user-identity-info`.

### AWS CLI

Untuk memperbarui informasi identitas pengguna

`update-user-identity-info` Contoh berikut memperbarui informasi identitas untuk pengguna Amazon Connect yang ditentukan.

```
aws connect update-user-identity-info \  
  --identity-info FirstName=Mary,LastName=Major,Email=marym@example.com \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengkonfigurasi Pengaturan Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [UpdateUserIdentityInfo](#) di Referensi AWS CLI Perintah.

## update-user-phone-config

Contoh kode berikut menunjukkan cara menggunakan `update-user-phone-config`.

### AWS CLI

Untuk memperbarui konfigurasi telepon pengguna

`update-user-phone-config` Contoh berikut memperbarui konfigurasi telepon untuk pengguna yang ditentukan.

```
aws connect update-user-phone-config \  
  --phone-config  
  PhoneType=SOFT_PHONE,AutoAccept=false,AfterContactWorkTimeLimit=60,DeskPhoneNumber=  
+18005551212 \  
  --user-id 12345678-4444-3333-2222-111122223333 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengkonfigurasi Pengaturan Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [UpdateUserPhoneConfig](#) di Referensi AWS CLI Perintah.

## update-user-routing-profile

Contoh kode berikut menunjukkan cara menggunakan `update-user-routing-profile`.

## AWS CLI

Untuk memperbarui profil perutean pengguna

`update-user-routing-profile` Contoh berikut memperbarui profil perutean untuk pengguna Amazon Connect yang ditentukan.

```
aws connect update-user-routing-profile \  
  --routing-profile-id 12345678-1111-3333-2222-4444EXAMPLE \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengkonfigurasi Pengaturan Agen](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [UpdateUserRoutingProfile](#) di Referensi AWS CLI Perintah.

## **update-user-security-profiles**

Contoh kode berikut menunjukkan cara menggunakan `update-user-security-profiles`.

### AWS CLI

Untuk memperbarui profil keamanan pengguna

`update-user-security-profiles` Contoh berikut memperbarui profil keamanan untuk pengguna Amazon Connect yang ditentukan.

```
aws connect update-user-security-profiles \  
  --security-profile-ids 12345678-1234-1234-1234-1234567892111 \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menetapkan Izin: Profil Keamanan](#) di Panduan Administrator Amazon Connect.

- Untuk detail API, lihat [UpdateUserSecurityProfiles](#) di Referensi AWS CLI Perintah.

## AWS Cost and Usage Report contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Cost and Usage Report.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-report-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-report-definition`.

AWS CLI

Untuk menghapus Laporan AWS Biaya dan Penggunaan

Contoh ini menghapus Laporan AWS Biaya dan Penggunaan.

Perintah:

```
aws cur --region us-east-1 delete-report-definition --report-name "ExampleReport"
```

- Untuk detail API, lihat [DeleteReportDefinition](#) di Referensi AWS CLI Perintah.

### **describe-report-definitions**

Contoh kode berikut menunjukkan cara menggunakan `describe-report-definitions`.

## AWS CLI

Untuk mengambil daftar Laporan AWS Biaya dan Penggunaan

Contoh ini menjelaskan daftar Laporan AWS Biaya dan Penggunaan yang dimiliki oleh akun.

Perintah:

```
aws cur --region us-east-1 describe-report-definitions --max-items 5
```

Output:

```
{
  "ReportDefinitions": [
    {
      "ReportName": "ExampleReport",
      "Compression": "ZIP",
      "S3Region": "us-east-1",
      "Format": "textORcsv",
      "S3Prefix": "exampleprefix",
      "S3Bucket": "example-s3-bucket",
      "TimeUnit": "DAILY",
      "AdditionalArtifacts": [
        "REDSHIFT",
        "QUICKSIGHT"
      ],
      "AdditionalSchemaElements": [
        "RESOURCES"
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeReportDefinitions](#) di Referensi AWS CLI Perintah.

## put-report-definition

Contoh kode berikut menunjukkan cara menggunakan `put-report-definition`.

## AWS CLI

Untuk membuat Laporan AWS Biaya dan Penggunaan

`put-report-definition` Contoh berikut membuat Laporan AWS Biaya dan Penggunaan harian yang dapat Anda unggah ke Amazon Redshift atau Amazon. QuickSight

```
aws cur put-report-definition --report-definition file://report-definition.json
```

Isi dari `report-definition.json`:

```
{
  "ReportName": "ExampleReport",
  "TimeUnit": "DAILY",
  "Format": "textORcsv",
  "Compression": "ZIP",
  "AdditionalSchemaElements": [
    "RESOURCES"
  ],
  "S3Bucket": "example-s3-bucket",
  "S3Prefix": "exampleprefix",
  "S3Region": "us-east-1",
  "AdditionalArtifacts": [
    "REDSHIFT",
    "QUICKSIGHT"
  ]
}
```

- Untuk detail API, lihat [PutReportDefinition](#) di Referensi AWS CLI Perintah.

## Contoh Layanan Cost Explorer menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Layanan AWS Command Line Interface With Cost Explorer.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **get-cost-and-usage**

Contoh kode berikut menunjukkan cara menggunakan `get-cost-and-usage`.

## AWS CLI

Untuk mengambil penggunaan akun S3 untuk bulan September 2017

`get-cost-and-usage` Contoh berikut mengambil penggunaan akun S3 untuk bulan September 2017.

```
aws ce get-cost-and-usage \
  --time-period Start=2017-09-01,End=2017-10-01 \
  --granularity MONTHLY \
  --metrics "BlendedCost" "UnblendedCost" "UsageQuantity" \
  --group-by Type=DIMENSION,Key=SERVICE Type=TAG,Key=Environment \
  --filter file:///filters.json
```

Isi dari `filters.json`:

```
{
  "Dimensions": {
    "Key": "SERVICE",
    "Values": [
      "Amazon Simple Storage Service"
    ]
  }
}
```

Output:

```
{
  "GroupDefinitions": [
    {
      "Type": "DIMENSION",
      "Key": "SERVICE"
    },
  ],
```

```
{
  "Type": "TAG",
  "Key": "Environment"
},
"ResultsByTime": [
  {
    "Estimated": false,
    "TimePeriod": {
      "Start": "2017-09-01",
      "End": "2017-10-01"
    },
    "Total": {},
    "Groups": [
      {
        "Keys": [
          "Amazon Simple Storage Service",
          "Environment$"
        ],
        "Metrics": {
          "BlendedCost": {
            "Amount": "40.3527508453",
            "Unit": "USD"
          },
          "UnblendedCost": {
            "Amount": "40.3543773134",
            "Unit": "USD"
          },
          "UsageQuantity": {
            "Amount": "9312771.098461578",
            "Unit": "N/A"
          }
        }
      }
    ],
    {
      "Keys": [
        "Amazon Simple Storage Service",
        "Environment$Dev"
      ],
      "Metrics": {
        "BlendedCost": {
          "Amount": "0.2682364644",
          "Unit": "USD"
        }
      }
    }
  ]
}
```



```

    "UnblendedCost": {
      "Amount": "0.2682364644",
      "Unit": "USD"
    },
    "UsageQuantity": {
      "Amount": "22403.4395271182",
      "Unit": "N/A"
    }
  }
}
]
}
]
}
}

```

- Untuk detail API, lihat [GetCostAndUsage](#) di Referensi AWS CLI Perintah.

## get-dimension-values

Contoh kode berikut menunjukkan cara menggunakan `get-dimension-values`.

### AWS CLI

Untuk mengambil tag untuk dimensi SERVICE, dengan nilai “Elastis”

Contoh ini mengambil tag untuk dimensi SERVICE, dengan nilai “Elastis” untuk 01 Januari 2017 hingga 18 Mei 2017.

Perintah:

```
aws ce get-dimension-values --search-string Elastic --time-period
Start=2017-01-01,End=2017-05-18 --dimension SERVICE
```

Output:

```

{
  "TotalSize": 6,
  "DimensionValues": [
    {
      "Attributes": {},
      "Value": "Amazon ElastiCache"
    },
  ],
}

```

```

    {
      "Attributes": {},
      "Value": "EC2 - Other"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Compute Cloud - Compute"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Load Balancing"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic MapReduce"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elasticsearch Service"
    }
  ],
  "ReturnSize": 6
}

```

- Untuk detail API, lihat [GetDimensionValues](#) di Referensi AWS CLI Perintah.

## get-reservation-coverage

Contoh kode berikut menunjukkan cara menggunakan `get-reservation-coverage`.

### AWS CLI

Untuk mengambil cakupan reservasi untuk instans EC2 t2.nano di wilayah us-east-1

Contoh ini mengambil cakupan reservasi untuk instans EC2 t2.nano di wilayah us-east-1 untuk Juli-September 2017.

Perintah:

```
aws ce get-reservation-coverage --time-period Start=2017-07-01,End=2017-10-01 --
group-by Type=Dimension,Key=REGION --filter file://filters.json
```

filters.json:

```
{
  "And": [
    {
      "Dimensions": {
        "Key": "INSTANCE_TYPE",
        "Values": [
          "t2.nano"
        ]
      },
      "Dimensions": {
        "Key": "REGION",
        "Values": [
          "us-east-1"
        ]
      }
    }
  ]
}
```

#### Output:

```
{
  "TotalSize": 6,
  "DimensionValues": [
    {
      "Attributes": {},
      "Value": "Amazon ElastiCache"
    },
    {
      "Attributes": {},
      "Value": "EC2 - Other"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Compute Cloud - Compute"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Load Balancing"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic MapReduce"
    }
  ]
}
```

```

    },
    {
      "Attributes": {},
      "Value": "Amazon Elasticsearch Service"
    }
  ],
  "ReturnSize": 6
}

```

- Untuk detail API, lihat [GetReservationCoveragedi Referensi AWS CLI Perintah](#).

## get-reservation-purchase-recommendation

Contoh kode berikut menunjukkan cara menggunakan `get-reservation-purchase-recommendation`.

### AWS CLI

Untuk mengambil rekomendasi reservasi untuk Partial Upfront EC2 RI dengan jangka waktu tiga tahun

`get-reservation-purchase-recommendation` Contoh berikut mengambil rekomendasi untuk instans EC2 Partial Upfront dengan jangka waktu tiga tahun, berdasarkan 60 hari terakhir penggunaan EC2.

```

aws ce get-reservation-purchase-recommendation \
  --service "Amazon Redshift" \
  --lookback-period-in-days SIXTY_DAYS \
  --term-in-years THREE_YEARS \
  --payment-option PARTIAL_UPFRONT

```

### Output:

```

{
  "Recommendations": [],
  "Metadata": {
    "GenerationTimestamp": "2018-08-08T15:20:57Z",
    "RecommendationId": "00d59dde-a1ad-473f-8ff2-iexample3330b"
  }
}

```

- Untuk detail API, lihat [GetReservationPurchaseRecommendation](#) di Referensi AWS CLI Perintah.

## get-reservation-utilization

Contoh kode berikut menunjukkan cara menggunakan `get-reservation-utilization`.

### AWS CLI

Untuk mengambil pemanfaatan reservasi untuk akun Anda

`get-reservation-utilization` Contoh berikut mengambil pemanfaatan RI untuk semua jenis instans `t2.nano` dari 2018-03-01 hingga 2018-08-01 untuk akun tersebut.

```
aws ce get-reservation-utilization \
  --time-period Start=2018-03-01,End=2018-08-01 \
  --filter file://filters.json
```

Isi dari `filters.json`:

```
{
  "Dimensions": {
    "Key": "INSTANCE_TYPE",
    "Values": [
      "t2.nano"
    ]
  }
}
```

Output:

```
{
  "Total": {
    "TotalAmortizedFee": "0",
    "UtilizationPercentage": "0",
    "PurchasedHours": "0",
    "NetRISavings": "0",
    "TotalActualHours": "0",
    "AmortizedRecurringFee": "0",
    "UnusedHours": "0",
    "TotalPotentialRISavings": "0",
```

```
    "OnDemandCostOfRIHoursUsed": "0",
    "AmortizedUpfrontFee": "0"
  },
  "UtilizationsByTime": []
}
```

- Untuk detail API, lihat [GetReservationUtilization](#) di Referensi AWS CLI Perintah.

## get-tags

Contoh kode berikut menunjukkan cara menggunakan `get-tags`.

### AWS CLI

Untuk mengambil kunci dan nilai untuk tag alokasi biaya

Contoh ini mengambil semua tag alokasi biaya dengan kunci "Proyek" dan nilai yang berisi "SecretProject".

Perintah:

```
aws ce get-tags --search-string secretProject --time-period
Start=2017-01-01,End=2017-05-18 --tag-key Project
```

Output:

```
{
  "ReturnSize": 2,
  "Tags": [
    "secretProject1",
    "secretProject2"
  ],
  "TotalSize": 2
}
```

- Untuk detail API, lihat [GetTags](#) di Referensi AWS CLI Perintah.

## Contoh Firehose menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface Firehose with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **list-delivery-streams**

Contoh kode berikut menunjukkan cara menggunakan `list-delivery-streams`.

AWS CLI

Untuk membuat daftar aliran pengiriman yang tersedia

`list-delivery-streams` Contoh berikut mencantumkan aliran pengiriman yang tersedia di AWS akun Anda.

```
aws firehose list-delivery-streams
```

Output:

```
{
  "DeliveryStreamNames": [
    "my-stream"
  ],
  "HasMoreDeliveryStreams": false
}
```

Untuk informasi selengkapnya, lihat [Membuat Aliran Pengiriman Amazon Kinesis Data Firehose](#) di Panduan Developer Amazon Kinesis Data Firehose.

- Untuk detail API, lihat [ListDeliveryStreams](#) di Referensi AWS CLI Perintah.

## put-record-batch

Contoh kode berikut menunjukkan cara menggunakan `put-record-batch`.

### AWS CLI

Untuk menulis beberapa catatan ke aliran

`put-record-batch` Contoh berikut menulis tiga catatan ke aliran. Data dikodekan dalam format Base64.

```
aws firehose put-record-batch \
  --delivery-stream-name my-stream \
  --records file://records.json
```

Isi dari `myfile.json`:

```
[
  {"Data": "Rm1yc3QgdGhpbmc="},
  {"Data": "U2Vjb25kIHRoaW5n"},
  {"Data": "VGhpcmQgdGhpbmc="}
]
```

Output:

```
{
  "FailedPutCount": 0,
  "Encrypted": false,
  "RequestResponses": [
    {
      "RecordId": "9D20J6t2EqCTZTXwGzeSv/EVHxRoRCw89xd+o3+sXg8DhY0aWKPSmZy/
CGlRVEys1u1xbeKh6VofEYKkoeiDrcjrxhQp9iF7sUW7pujiMEQ5LzlrzCkGosxQn
+3boDnURDEaD42V7Giixp0yLJkYZcae1i7HzlCEoy9LJhMr8EjDSi40m/9Vc2uhwwuAtGE0XKpxJ2WD7ZRWtAnY1KAnv
"},
    {
      "RecordId": "jFirejqxCLlK5xjH/UNm1MVCjktEN76I7916X9PaZ
+PVa0SXDFu1WG0qEZhxq2js7xcZ552eoeDxsuTU1MSq9nZTbVfb6cQTIXnm/GsuF37Uhg67GkmR5z9016XKJ
+/+pDloFv7Hh9a3oUS6wYm3DcNRLTHHAimANp1PhkQvWpvLRfzbuCUkBphR2QVzhP90iHLbzGwy8/
DfH8sqWEUYASNJKS8GXP5s"
```



```

    },
    {
      "RecordId":
        "oy0amQ40o5Y2YV4vxzufdcM00w6n3EPr3tpPJGoYVVKH4APPVqNcbUgefo1stEFRg4hTLrf2k6eliHu/9+YJ5R3iie
DTBt3qBlmTj7Xq8SKVb01S7YvMTpWkMKA86f8JfmT8BMKoMb4XZS/s0kQLe+qh0sYKXW1"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Mengirim Data ke Aliran Pengiriman Amazon Kinesis Data Firehose](#) di Panduan Pengembang Amazon Kinesis Data Firehose.

- Untuk detail API, lihat [PutRecordBatch](#) di Referensi AWS CLI Perintah.

## put-record

Contoh kode berikut menunjukkan cara menggunakan `put-record`.

### AWS CLI

Untuk menulis catatan ke aliran

`put-record` Contoh berikut menulis data ke aliran. Data dikodekan dalam format Base64.

```

aws firehose put-record \
  --delivery-stream-name my-stream \
  --record '{"Data":"SGVsbG8gd29ybGQ="}'

```

Output:

```

{
  "RecordId": "RjB5K/nnoGFHqwTsZ1Nd/
TTqvjE8V5dsyXZTQn2JXrdpMT0wssyEb6nfC8fwf1whhwnItt4mvrn+gsqeK5jB7QjuLg283+Ps4Sz/
j1Xujv31iDhnPdaLw4B0yM9Amv7PcCuB2079RuM0NhoakbyUymIwY8yt20G8X2420wu1j1Fafhci4erAt7QhDEvpwuK8
  "Encrypted": false
}

```

Untuk informasi selengkapnya, lihat [Mengirim Data ke Aliran Pengiriman Amazon Kinesis Data Firehose](#) di Panduan Pengembang Amazon Kinesis Data Firehose.

- Untuk detail API, lihat [PutRecord](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Data Lifecycle Manager menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum menggunakan Amazon Data Lifecycle Manager AWS Command Line Interface with Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-default-role**

Contoh kode berikut menunjukkan cara menggunakan `create-default-role`.

AWS CLI

Untuk membuat peran IAM yang diperlukan untuk Amazon DLM

`d1m create-default-role` Contoh berikut membuat peran AWS DataLifecycleManagerDefaultRole default untuk mengelola snapshot.

```
aws dlm create-default-role \  
  --resource-type snapshot
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Peran layanan default untuk Amazon Data Lifecycle Manager di Panduan Pengguna Amazon Elastic Compute Cloud](#).

- Untuk detail API, lihat [CreateDefaultRole](#) di Referensi AWS CLI Perintah.

## create-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `create-lifecycle-policy`.

### AWS CLI

Untuk membuat kebijakan siklus hidup

`create-lifecycle-policy` Contoh berikut membuat kebijakan siklus hidup yang membuat snapshot harian volume pada waktu yang ditentukan. Tag yang ditentukan ditambahkan ke snapshot, dan tag juga disalin dari volume dan ditambahkan ke snapshot. Jika membuat snapshot baru melebihi jumlah maksimum yang ditentukan, snapshot tertua akan dihapus.

```
aws dlm create-lifecycle-policy \  
  --description "My first policy" \  
  --state ENABLED \  
  --execution-role-arn arn:aws:iam::12345678910:role/  
AWSDataLifecycleManagerDefaultRole \  
  --policy-details file://policyDetails.json
```

Isi dari `policyDetails.json`:

```
{  
  "ResourceTypes": [  
    "VOLUME"  
  ],  
  "TargetTags": [  
    {  
      "Key": "costCenter",  
      "Value": "115"  
    }  
  ],  
  "Schedules": [  
    {  
      "Name": "DailySnapshots",  
      "CopyTags": true,  
      "TagsToAdd": [  
        {  
          "Key": "type",  
          "Value": "myDailySnapshot"  
        }  
      ],  
      "CreateRule": {
```

```
        "Interval": 24,  
        "IntervalUnit": "HOURS",  
        "Times": [  
            "03:00"  
        ]  
    },  
    "RetainRule": {  
        "Count": 5  
    }  
}  
]  
}
```

Output:

```
{  
  "PolicyId": "policy-0123456789abcdef0"  
}
```

- Untuk detail API, lihat [CreateLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## **delete-lifecycle-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-lifecycle-policy`.

### AWS CLI

Untuk menghapus kebijakan siklus hidup

Contoh berikut menghapus kebijakan siklus hidup yang ditentukan. :

```
aws dlm delete-lifecycle-policy --policy-id policy-0123456789abcdef0
```

- Untuk detail API, lihat [DeleteLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## **get-lifecycle-policies**

Contoh kode berikut menunjukkan cara menggunakan `get-lifecycle-policies`.

### AWS CLI

Untuk mendapatkan ringkasan kebijakan siklus hidup Anda

`get-lifecycle-policies` Contoh berikut mencantumkan semua kebijakan siklus hidup Anda.

```
aws dlm get-lifecycle-policies
```

Output:

```
{
  "Policies": [
    {
      "PolicyId": "policy-0123456789abcdef0",
      "Description": "My first policy",
      "State": "ENABLED"
    }
  ]
}
```

- Untuk detail API, lihat [GetLifecyclePolicies](#) di Referensi AWS CLI Perintah.

## get-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `get-lifecycle-policy`.

AWS CLI

Untuk menjelaskan kebijakan siklus hidup

`get-lifecycle-policy` Contoh berikut menampilkan detail untuk kebijakan siklus hidup yang ditentukan.

```
aws dlm get-lifecycle-policy \
  --policy-id policy-0123456789abcdef0
```

Output:

```
{
  "Policy": {
    "PolicyId": "policy-0123456789abcdef0",
    "Description": "My policy",
    "State": "ENABLED",
    "ExecutionRoleArn": "arn:aws:iam::123456789012:role/
AWSDataLifecycleManagerDefaultRole",
    "DateCreated": "2019-08-08T17:45:42Z",
```

```

    "DateModified": "2019-08-08T17:45:42Z",
    "PolicyDetails": {
      "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
      "ResourceTypes": [
        "VOLUME"
      ],
      "TargetTags": [
        {
          "Key": "costCenter",
          "Value": "115"
        }
      ],
      "Schedules": [
        {
          "Name": "DailySnapshots",
          "CopyTags": true,
          "TagsToAdd": [
            {
              "Key": "type",
              "Value": "myDailySnapshot"
            }
          ],
          "CreateRule": {
            "Interval": 24,
            "IntervalUnit": "HOURS",
            "Times": [
              "03:00"
            ]
          },
          "RetainRule": {
            "Count": 5
          }
        }
      ]
    }
  }
}

```

- Untuk detail API, lihat [GetLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## update-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `update-lifecycle-policy`.

## AWS CLI

Contoh 1: Untuk mengaktifkan kebijakan siklus hidup

`update-lifecycle-policy` Contoh berikut mengaktifkan kebijakan siklus hidup yang ditentukan.

```
aws dlm update-lifecycle-policy \  
  --policy-id policy-0123456789abcdef0 \  
  --state ENABLED
```

Contoh 2: Untuk menonaktifkan kebijakan siklus hidup

`update-lifecycle-policy` Contoh berikut menonaktifkan kebijakan siklus hidup yang ditentukan.

```
aws dlm update-lifecycle-policy \  
  --policy-id policy-0123456789abcdef0 \  
  --state DISABLED
```

Contoh 3: Untuk memperbarui detail kebijakan siklus hidup

`update-lifecycle-policy` Contoh berikut memperbarui tag target untuk kebijakan siklus hidup yang ditentukan.

```
aws dlm update-lifecycle-policy \  
  --policy-id policy-0123456789abcdef0 \  
  --policy-details file://policyDetails.json
```

Isi dari `policyDetails.json`. Detail lain yang tidak direferensikan dalam file ini tidak diubah oleh perintah.

```
{  
  "TargetTags": [  
    {  
      "Key": "costCenter",  
      "Value": "120"  
    },  
    {  
      "Key": "project",
```

```
    "Value": "lima"  
  }  
]  
}
```

- Untuk detail API, lihat [UpdateLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## AWS Data Pipeline contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Data Pipeline.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **activate-pipeline**

Contoh kode berikut menunjukkan cara menggunakan `activate-pipeline`.

#### AWS CLI

Untuk mengaktifkan pipa

Contoh ini mengaktifkan pipeline yang ditentukan:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Untuk mengaktifkan pipeline pada tanggal dan waktu tertentu, gunakan perintah berikut:



```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE --start-timestamp 2015-04-07T00:00:00Z
```

- Untuk detail API, lihat [ActivatePipeline](#) di Referensi AWS CLI Perintah.

## add-tags

Contoh kode berikut menunjukkan cara menggunakan `add-tags`.

### AWS CLI

Untuk menambahkan tag ke pipeline

Contoh ini menambahkan tag yang ditentukan ke pipeline yang ditentukan:

```
aws datapipeline add-tags --pipeline-id df-00627471S0VYZEXAMPLE --tags key=environment,value=production key=owner,value=sales
```

Untuk melihat tag, gunakan perintah `describe-pipelines`. Misalnya, tag yang ditambahkan dalam perintah contoh muncul sebagai berikut dalam output untuk `describe-pipelines`:

```
{
  ...
  "tags": [
    {
      "value": "production",
      "key": "environment"
    },
    {
      "value": "sales",
      "key": "owner"
    }
  ]
  ...
}
```

- Untuk detail API, lihat [AddTags](#) di Referensi AWS CLI Perintah.

## create-pipeline

Contoh kode berikut menunjukkan cara menggunakan `create-pipeline`.

## AWS CLI

Untuk membuat pipa

Contoh ini membuat pipeline:

```
aws datapipeline create-pipeline --name my-pipeline --unique-id my-pipeline-token
```

Berikut ini adalah output contoh:

```
{
  "pipelineId": "df-00627471S0VYZEXAMPLE"
}
```

- Untuk detail API, lihat [CreatePipeline](#) di Referensi AWS CLI Perintah.

## **deactivate-pipeline**

Contoh kode berikut menunjukkan cara menggunakan `deactivate-pipeline`.

## AWS CLI

Untuk menonaktifkan pipa

Contoh ini menonaktifkan pipeline yang ditentukan:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Untuk menonaktifkan pipeline hanya setelah semua aktivitas yang berjalan selesai, gunakan perintah berikut:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE --no-cancel-active
```

- Untuk detail API, lihat [DeactivatePipeline](#) di Referensi AWS CLI Perintah.

## **delete-pipeline**

Contoh kode berikut menunjukkan cara menggunakan `delete-pipeline`.

## AWS CLI

Untuk menghapus pipa

Contoh ini menghapus pipeline yang ditentukan:

```
aws datapipeline delete-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

- Untuk detail API, lihat [DeletePipeline](#) di Referensi AWS CLI Perintah.

## describe-pipelines

Contoh kode berikut menunjukkan cara menggunakan `describe-pipelines`.

## AWS CLI

Untuk mendeskripsikan saluran pipa Anda

Contoh ini menjelaskan pipeline yang ditentukan:

```
aws datapipeline describe-pipelines --pipeline-ids df-00627471S0VYZEXAMPLE
```

Berikut ini adalah output contoh:

```
{
  "pipelineDescriptionList": [
    {
      "fields": [
        {
          "stringValue": "PENDING",
          "key": "@pipelineState"
        },
        {
          "stringValue": "my-pipeline",
          "key": "name"
        },
        {
          "stringValue": "2015-04-07T16:05:58",
          "key": "@creationTime"
        },
        {
          "stringValue": "df-00627471S0VYZEXAMPLE",
          "key": "@id"
        }
      ]
    }
  ]
}
```

```

    },
    {
      "stringValue": "123456789012",
      "key": "pipelineCreator"
    },
    {
      "stringValue": "PIPELINE",
      "key": "@sphere"
    },
    {
      "stringValue": "123456789012",
      "key": "@userId"
    },
    {
      "stringValue": "123456789012",
      "key": "@accountId"
    },
    {
      "stringValue": "my-pipeline-token",
      "key": "uniqueId"
    }
  ],
  "pipelineId": "df-00627471S0VYZEXAMPLE",
  "name": "my-pipeline",
  "tags": []
}
]
}

```

- Untuk detail API, lihat [DescribePipelines](#) di Referensi AWS CLI Perintah.

## get-pipeline-definition

Contoh kode berikut menunjukkan cara menggunakan `get-pipeline-definition`.

### AWS CLI

Untuk mendapatkan definisi pipeline

Contoh ini mendapatkan definisi pipeline untuk pipeline yang ditentukan:

```
aws datapipeline get-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE
```

Berikut ini adalah output contoh:

```
{
  "parameters": [
    {
      "type": "AWS::S3::ObjectKey",
      "id": "myS3OutputLoc",
      "description": "S3 output folder"
    },
    {
      "default": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/data",
      "type": "AWS::S3::ObjectKey",
      "id": "myS3InputLoc",
      "description": "S3 input folder"
    },
    {
      "default": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* >
${OUTPUT1_STAGING_DIR}/output.txt",
      "type": "String",
      "id": "myShellCmd",
      "description": "Shell command to run"
    }
  ],
  "objects": [
    {
      "type": "Ec2Resource",
      "terminateAfter": "20 Minutes",
      "instanceType": "t1.micro",
      "id": "EC2ResourceObj",
      "name": "EC2ResourceObj"
    },
    {
      "name": "Default",
      "failureAndRerunMode": "CASCADE",
      "resourceRole": "DataPipelineDefaultResourceRole",
      "schedule": {
        "ref": "DefaultSchedule"
      },
      "role": "DataPipelineDefaultRole",
      "scheduleType": "cron",
      "id": "Default"
    },
    {
```

```

        "directoryPath": "#{myS3OutputLoc}/#{format(@scheduledStartTime, 'YYYY-MM-dd-HH-mm-ss')}",
        "type": "S3DataNode",
        "id": "S3OutputLocation",
        "name": "S3OutputLocation"
    },
    {
        "directoryPath": "#{myS3InputLoc}",
        "type": "S3DataNode",
        "id": "S3InputLocation",
        "name": "S3InputLocation"
    },
    {
        "startAt": "FIRST_ACTIVATION_DATE_TIME",
        "name": "Every 15 minutes",
        "period": "15 minutes",
        "occurrences": "4",
        "type": "Schedule",
        "id": "DefaultSchedule"
    },
    {
        "name": "ShellCommandActivityObj",
        "command": "#{myShellCmd}",
        "output": {
            "ref": "S3OutputLocation"
        },
        "input": {
            "ref": "S3InputLocation"
        },
        "stage": "true",
        "type": "ShellCommandActivity",
        "id": "ShellCommandActivityObj",
        "runsOn": {
            "ref": "EC2ResourceObj"
        }
    }
],
"values": {
    "myS3OutputLoc": "s3://my-s3-bucket/",
    "myS3InputLoc": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/
data",
    "myShellCmd": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* >
${OUTPUT1_STAGING_DIR}/output.txt"
}

```

```
}
```

- Untuk detail API, lihat [GetPipelineDefinition](#) di Referensi AWS CLI Perintah.

## list-pipelines

Contoh kode berikut menunjukkan cara menggunakan `list-pipelines`.

### AWS CLI

Untuk membuat daftar saluran pipa Anda

Contoh ini mencantumkan pipeline Anda:

```
aws datapipeline list-pipelines
```

Berikut ini adalah output contoh:

```
{
  "pipelineIdList": [
    {
      "id": "df-00627471S0VYZEXAMPLE",
      "name": "my-pipeline"
    },
    {
      "id": "df-09028963KNVMREXAMPLE",
      "name": "ImportDDB"
    },
    {
      "id": "df-0870198233ZYVEXAMPLE",
      "name": "CrossRegionDDB"
    },
    {
      "id": "df-00189603TB4MZEXAMPLE",
      "name": "CopyRedshift"
    }
  ]
}
```

- Untuk detail API, lihat [ListPipelines](#) di Referensi AWS CLI Perintah.

## list-runs

Contoh kode berikut menunjukkan cara menggunakan `list-runs`.

### AWS CLI

Contoh 1: Untuk membuat daftar pipeline Anda berjalan

`list-runs` Contoh berikut mencantumkan run untuk pipeline yang ditentukan.

```
aws datapipeline list-runs --pipeline-id df-00627471S0VYZEXAMPLE
```

Output:

	Name	Scheduled Start	Status	Ended	ID
		Started			
1.	EC2ResourceObj	2015-04-12T17:33:02	CREATING		
	@EC2ResourceObj_2015-04-12T17:33:02		2015-04-12T17:33:10		
2.	S3InputLocation	2015-04-12T17:33:02	FINISHED		
	@S3InputLocation_2015-04-12T17:33:02		2015-04-12T17:33:09		
	2015-04-12T17:33:09				
3.	S3OutputLocation	2015-04-12T17:33:02	WAITING_ON_DEPENDENCIES		
	@S3OutputLocation_2015-04-12T17:33:02		2015-04-12T17:33:09		
4.	ShellCommandActivityObj	2015-04-12T17:33:02	WAITING_FOR_RUNNER		
	@ShellCommandActivityObj_2015-04-12T17:33:02		2015-04-12T17:33:09		

Contoh 2: Untuk membuat daftar pipeline berjalan di antara tanggal yang ditentukan

`list-runs` Contoh berikut menggunakan `--start-interval` untuk menentukan tanggal untuk memasukkan dalam output.

```
aws datapipeline list-runs --pipeline-id df-01434553B58A2SHZUK05 --start-interval 2017-10-07T00:00:00,2017-10-08T00:00:00
```

- Untuk detail API, lihat [ListRuns](#) di Referensi AWS CLI Perintah.

## put-pipeline-definition

Contoh kode berikut menunjukkan cara menggunakan `put-pipeline-definition`.



## AWS CLI

Untuk mengunggah definisi pipeline

Contoh ini mengunggah definisi pipeline yang ditentukan ke pipeline yang ditentukan:

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --
pipeline-definition file://my-pipeline-definition.json
```

Berikut ini adalah output contoh:

```
{
  "validationErrors": [],
  "errored": false,
  "validationWarnings": []
}
```

- Untuk detail API, lihat [PutPipelineDefinition](#) di Referensi AWS CLI Perintah.

## remove-tags

Contoh kode berikut menunjukkan cara menggunakan `remove-tags`.

## AWS CLI

Untuk menghapus tag dari pipeline

Contoh ini menghapus tag yang ditentukan dari pipeline yang ditentukan:

```
aws datapipeline remove-tags --pipeline-id df-00627471S0VYZEXAMPLE --tag-keys
environment
```

- Untuk detail API, lihat [RemoveTags](#) di Referensi AWS CLI Perintah.

## DataSync contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with DataSync.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **update-location-azure-blob**

Contoh kode berikut menunjukkan cara menggunakan `update-location-azure-blob`.

AWS CLI

Untuk memperbarui lokasi transfer Anda dengan agen baru

`update-location-object-storage` Contoh berikut memperbarui DataSync lokasi Anda untuk Microsoft Azure Blob Storage dengan agen baru.

```
aws datasync update-location-azure-blob \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0 \  
  --sas-configuration '{ \  
    "Token": "sas-token-for-azure-blob-storage-access" \  
  }'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti agen Anda](#) di Panduan AWS DataSync Pengguna.

- Untuk detail API, lihat [UpdateLocationAzureBlob](#) di Referensi AWS CLI Perintah.

## update-location-hdfs

Contoh kode berikut menunjukkan cara menggunakan `update-location-hdfs`.

### AWS CLI

Untuk memperbarui lokasi transfer Anda dengan agen baru

`update-location-hdfs` Contoh berikut memperbarui lokasi DataSync HDFS Anda dengan agen baru. Anda hanya perlu `--kerberos-krb5-conf` opsi `--kerberos-keytab` dan jika cluster HDFS Anda menggunakan otentikasi Kerberos.

```
aws datasync update-location-hdfs \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0 \  
  --kerberos-keytab file://hdfs.keytab  
  --kerberos-krb5-conf file://krb5.conf
```

Isi dari `hdfs.keytab`:

```
N/A. The content of this file is encrypted and not human readable.
```

Isi dari `krb5.conf`:

```
[libdefaults]  
  default_realm = EXAMPLE.COM  
  dns_lookup_realm = false  
  dns_lookup_kdc = false  
  rdns = true  
  ticket_lifetime = 24h  
  forwardable = true  
  udp_preference_limit = 1000000  
  default_tkt_enctypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-  
sha1  
  default_tgs_enctypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-  
sha1  
  permitted_enctypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-  
sha1
```

```
[realms]
EXAMPLE.COM = {
    kdc = kdc1.example.com
    admin_server = krbadmin.example.com
    default_domain = example.com
}

[domain_realm]
.example.com = EXAMPLE.COM
example.com = EXAMPLE.COM

[logging]
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kerberos/kadmin.log
default = FILE:/var/log/krb5libs.log
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti agen Anda](#) di Panduan AWS DataSync Pengguna.

- Untuk detail API, lihat [UpdateLocationHdfs](#) di Referensi AWS CLI Perintah.

## update-location-nfs

Contoh kode berikut menunjukkan cara menggunakan `update-location-nfs`.

### AWS CLI

Untuk memperbarui lokasi transfer Anda dengan agen baru

`update-location-nfs` Contoh berikut memperbarui lokasi DataSync NFS Anda dengan agen baru.

```
aws datasync update-location-nfs \
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
  abcdef01234567890 \
  --on-prem-config AgentArns=arn:aws:datasync:us-west-2:123456789012:agent/
  agent-1234567890abcdef0
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti agen Anda](#) di Panduan AWS DataSync Pengguna.

- Untuk detail API, lihat [UpdateLocationNfs](#) di Referensi AWS CLI Perintah.

## update-location-object-storage

Contoh kode berikut menunjukkan cara menggunakan `update-location-object-storage`.

### AWS CLI

Untuk memperbarui lokasi transfer Anda dengan agen baru

`update-location-object-storage` Contoh berikut memperbarui lokasi penyimpanan DataSync objek Anda dengan agen baru.

```
aws datasync update-location-object-storage \
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
  abcdef01234567890 \
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/
  agent-1234567890abcdef0 \
  --secret-key secret-key-for-object-storage
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti agen Anda](#) di Panduan AWS DataSync Pengguna.

- Untuk detail API, lihat [UpdateLocationObjectStorage](#) di Referensi AWS CLI Perintah.

## update-location-smb

Contoh kode berikut menunjukkan cara menggunakan `update-location-smb`.

### AWS CLI

Untuk memperbarui lokasi transfer Anda dengan agen baru

`update-location-smb` Contoh berikut memperbarui lokasi DataSync SMB Anda dengan agen baru.

```
aws datasync update-location-smb \
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
  abcdef01234567890 \
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/
  agent-1234567890abcdef0 \
```

```
--password smb-file-server-password
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti agen Anda](#) di Panduan AWS DataSync Pengguna.

- Untuk detail API, lihat [UpdateLocationSmb](#) di Referensi AWS CLI Perintah.

## Contoh DAX menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan DAX AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-cluster**

Contoh kode berikut menunjukkan cara menggunakan `create-cluster`.

AWS CLI

Untuk membuat cluster DAX

`create-cluster` Contoh berikut membuat cluster DAX dengan pengaturan yang ditentukan.

```
aws dax create-cluster \  
  --cluster-name daxcluster \  
  --password smb-file-server-password
```

```
--node-type dax.r4.large \  
--replication-factor 3 \  
--iam-role-arn roleARN \  
--sse-specification Enabled=true
```

### Output:

```
{  
  "Cluster": {  
    "ClusterName": "daxcluster",  
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",  
    "TotalNodes": 3,  
    "ActiveNodes": 0,  
    "NodeType": "dax.r4.large",  
    "Status": "creating",  
    "ClusterDiscoveryEndpoint": {  
      "Port": 8111  
    },  
    "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",  
    "SubnetGroup": "default",  
    "SecurityGroups": [  
      {  
        "SecurityGroupIdentifier": "sg-1af6e36e",  
        "Status": "active"  
      }  
    ],  
    "IamRoleArn": "arn:aws:iam::123456789012:role/  
DAXServiceRoleForDynamoDBAccess",  
    "ParameterGroup": {  
      "ParameterGroupName": "default.dax1.0",  
      "ParameterApplyStatus": "in-sync",  
      "NodeIdsToReboot": []  
    },  
    "SSEDescription": {  
      "Status": "ENABLED"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat Cluster DAX di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [CreateCluster](#) di Referensi AWS CLI Perintah.

## create-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `create-parameter-group`.

### AWS CLI

Untuk membuat grup parameter

Contoh `` `create-parameter-group` `` berikut membuat grup parameter dengan pengaturan yang ditentukan.

```
aws dax create-parameter-group \  
  --parameter-group-name daxparametergroup \  
  --description "A new parameter group"
```

Output:

```
{  
  "ParameterGroup": {  
    "ParameterGroupName": "daxparametergroup",  
    "Description": "A new parameter group"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [CreateParameterGroup](#) di Referensi AWS CLI Perintah.

## create-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `create-subnet-group`.

### AWS CLI

Untuk membuat grup subnet DAX

`create-subnet-group` Contoh berikut membuat grup subnet dengan pengaturan yang ditentukan.

```
aws dax create-subnet-group \  
  --parameter-group-name daxparametergroup \  
  --description "A new parameter group"
```



```
--subnet-group-name daxSubnetGroup \  
--subnet-ids subnet-11111111 subnet-22222222
```

Output:

```
{  
  "SubnetGroup": {  
    "SubnetGroupName": "daxSubnetGroup",  
    "VpcId": "vpc-05a1fa8e00c325226",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-11111111",  
        "SubnetAvailabilityZone": "us-west-2b"  
      },  
      {  
        "SubnetIdentifier": "subnet-22222222",  
        "SubnetAvailabilityZone": "us-west-2c"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat Grup Subnet](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [CreateSubnetGroup](#) di Referensi AWS CLI Perintah.

## decrease-replication-factor

Contoh kode berikut menunjukkan cara menggunakan `decrease-replication-factor`.

### AWS CLI

Untuk menghapus satu atau lebih node dari cluster

`decrease-replication-factor` Contoh berikut mengurangi jumlah node dalam cluster DAX yang ditentukan menjadi satu.

```
aws dax decrease-replication-factor \  
  --cluster-name daxcluster \  
  --new-replication-factor 1
```

## Output:

```
{
  "Cluster": {
    "ClusterName": "daxcluster",
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
    "TotalNodes": 3,
    "ActiveNodes": 3,
    "NodeType": "dax.r4.large",
    "Status": "modifying",
    "ClusterDiscoveryEndpoint": {
      "Address": "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
      "Port": 8111
    },
    "Nodes": [
      {
        "NodeId": "daxcluster-a",
        "Endpoint": {
          "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
          "Port": 8111
        },
        "NodeCreateTime": 1576625059.509,
        "AvailabilityZone": "us-west-2c",
        "NodeStatus": "available",
        "ParameterGroupStatus": "in-sync"
      },
      {
        "NodeId": "daxcluster-b",
        "Endpoint": {
          "Address": "daxcluster-
b.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
          "Port": 8111
        },
        "NodeCreateTime": 1576625059.509,
        "AvailabilityZone": "us-west-2a",
        "NodeStatus": "available",
        "ParameterGroupStatus": "in-sync"
      },
      {
        "NodeId": "daxcluster-c",
        "Endpoint": {
          "Address": "daxcluster-
c.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
```

```

        "Port": 8111
      },
      "NodeCreateTime": 1576625059.509,
      "AvailabilityZone": "us-west-2b",
      "NodeStatus": "available",
      "ParameterGroupStatus": "in-sync"
    }
  ],
  "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
  "SubnetGroup": "default",
  "SecurityGroups": [
    {
      "SecurityGroupIdentifier": "sg-1af6e36e",
      "Status": "active"
    }
  ],
  "IamRoleArn": "arn:aws:iam::123456789012:role/DAXServiceRoleForDynamoDBAccess",
  "ParameterGroup": {
    "ParameterGroupName": "default.dax1.0",
    "ParameterApplyStatus": "in-sync",
    "NodeIdsToReboot": []
  },
  "SSEDescription": {
    "Status": "ENABLED"
  }
}
}

```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [DecreaseReplicationFactor](#) di Referensi AWS CLI Perintah.

## delete-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster`.

### AWS CLI

Untuk menghapus kluster DAX

`delete-cluster` Contoh berikut menghapus cluster DAX tertentu.

```
aws dax delete-cluster \  
  --cluster-name daxcluster
```

### Output:

```
{  
  "Cluster": {  
    "ClusterName": "daxcluster",  
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",  
    "TotalNodes": 3,  
    "ActiveNodes": 0,  
    "NodeType": "dax.r4.large",  
    "Status": "deleting",  
    "ClusterDiscoveryEndpoint": {  
      "Address": "dd.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",  
      "Port": 8111  
    },  
    "PreferredMaintenanceWindow": "fri:06:00-fri:07:00",  
    "SubnetGroup": "default",  
    "SecurityGroups": [  
      {  
        "SecurityGroupIdentifier": "sg-1af6e36e",  
        "Status": "active"  
      }  
    ],  
    "IamRoleArn": "arn:aws:iam::123456789012:role/  
DAXServiceRoleForDynamoDBAccess",  
    "ParameterGroup": {  
      "ParameterGroupName": "default.dax1.0",  
      "ParameterApplyStatus": "in-sync",  
      "NodeIdsToReboot": []  
    },  
    "SSEDescription": {  
      "Status": "ENABLED"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DeleteCluster](#) di Referensi AWS CLI Perintah.

## delete-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `delete-parameter-group`.

### AWS CLI

Untuk menghapus grup parameter

`delete-parameter-group` Contoh berikut menghapus kelompok parameter DAX yang ditentukan.

```
aws dax delete-parameter-group \  
  --parameter-group-name daxparametergroup
```

Output:

```
{  
  "DeletionMessage": "Parameter group daxparametergroup has been deleted."  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DeleteParameterGroup](#) di Referensi AWS CLI Perintah.

## delete-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `delete-subnet-group`.

### AWS CLI

Untuk menghapus grup subnet

`delete-subnet-group` Contoh berikut menghapus kelompok subnet DAX yang ditentukan.

```
aws dax delete-subnet-group \  
  --subnet-group-name daxSubnetGroup
```

Output:

```
{  
  "DeletionMessage": "Subnet group daxSubnetGroup has been deleted."  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [DeleteSubnetGroup](#) di Referensi AWS CLI Perintah.

## describe-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-clusters`.

### AWS CLI

Untuk mengembalikan informasi tentang semua kluster DAX yang disediakan

`describe-clusters` Contoh berikut menampilkan detail tentang semua kluster DAX yang disediakan.

```
aws dax describe-clusters
```

Output:

```
{
  "Clusters": [
    {
      "ClusterName": "daxcluster",
      "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
      "TotalNodes": 1,
      "ActiveNodes": 1,
      "NodeType": "dax.r4.large",
      "Status": "available",
      "ClusterDiscoveryEndpoint": {
        "Address":
"daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
        "Port": 8111
      },
      "Nodes": [
        {
          "NodeId": "daxcluster-a",
          "Endpoint": {
            "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
```

```

        "Port": 8111
      },
      "NodeCreateTime": 1576625059.509,
      "AvailabilityZone": "us-west-2c",
      "NodeStatus": "available",
      "ParameterGroupStatus": "in-sync"
    }
  ],
  "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
  "SubnetGroup": "default",
  "SecurityGroups": [
    {
      "SecurityGroupIdentifier": "sg-1af6e36e",
      "Status": "active"
    }
  ],
  "IamRoleArn": "arn:aws:iam::123456789012:role/DAXServiceRoleForDynamoDBAccess",
  "ParameterGroup": {
    "ParameterGroupName": "default.dax1.0",
    "ParameterApplyStatus": "in-sync",
    "NodeIdsToReboot": []
  },
  "SSEDescription": {
    "Status": "ENABLED"
  }
}
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DescribeClusters](#) di Referensi AWS CLI Perintah.

## describe-default-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-default-parameters`.

### AWS CLI

Untuk mengembalikan informasi parameter sistem default untuk DAX

`describe-default-parameters` Contoh berikut menampilkan informasi parameter sistem default untuk DAX.

```
aws dax describe-default-parameters
```

Output:

```
{
  "Parameters": [
    {
      "ParameterName": "query-ttl-millis",
      "ParameterType": "DEFAULT",
      "ParameterValue": "300000",
      "NodeTypeSpecificValues": [],
      "Description": "Duration in milliseconds for queries to remain cached",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": "TRUE",
      "ChangeType": "IMMEDIATE"
    },
    {
      "ParameterName": "record-ttl-millis",
      "ParameterType": "DEFAULT",
      "ParameterValue": "300000",
      "NodeTypeSpecificValues": [],
      "Description": "Duration in milliseconds for records to remain valid in
cache (Default: 0 = infinite)",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": "TRUE",
      "ChangeType": "IMMEDIATE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DescribeDefaultParameters](#) di Referensi AWS CLI Perintah.



## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Untuk mengembalikan semua peristiwa yang terkait dengan cluster DAX dan grup parameter `describe-events` Contoh berikut menampilkan rincian peristiwa yang terkait dengan cluster DAX dan kelompok parameter.

```
aws dax describe-events
```

Output:

```
{
  "Events": [
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Cluster deleted.",
      "Date": 1576702736.706
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Removed node daxcluster-b.",
      "Date": 1576702691.738
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Removed node daxcluster-a.",
      "Date": 1576702633.498
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Removed node daxcluster-c.",
      "Date": 1576702631.329
    },
    {
      "SourceName": "daxcluster",
```

```
    "SourceType": "CLUSTER",
    "Message": "Cluster created.",
    "Date": 1576626560.057
  }
]
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## describe-parameter-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-parameter-groups`.

### AWS CLI

Untuk menggambarkan kelompok parameter yang didefinisikan dalam DAX

`describe-parameter-groups` Contoh berikut mengambil rincian tentang kelompok parameter yang didefinisikan dalam DAX.

```
aws dax describe-parameter-groups
```

Output:

```
{
  "ParameterGroups": [
    {
      "ParameterGroupName": "default.dax1.0",
      "Description": "Default parameter group for dax1.0"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [DescribeParameterGroups](#) di Referensi AWS CLI Perintah.

## describe-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-parameters`.

### AWS CLI

Untuk menggambarkan parameter yang didefinisikan dalam kelompok parameter DAX

`describe-parameters` Contoh berikut mengambil rincian tentang parameter yang didefinisikan dalam kelompok parameter DAX yang ditentukan.

```
aws dax describe-parameters \
  --parameter-group-name default.dax1.0
```

Output:

```
{
  "Parameters": [
    {
      "ParameterName": "query-ttl-millis",
      "ParameterType": "DEFAULT",
      "ParameterValue": "300000",
      "NodeTypeSpecificValues": [],
      "Description": "Duration in milliseconds for queries to remain cached",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": "TRUE",
      "ChangeType": "IMMEDIATE"
    },
    {
      "ParameterName": "record-ttl-millis",
      "ParameterType": "DEFAULT",
      "ParameterValue": "300000",
      "NodeTypeSpecificValues": [],
      "Description": "Duration in milliseconds for records to remain valid in
cache (Default: 0 = infinite)",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": "TRUE",
      "ChangeType": "IMMEDIATE"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [DescribeParameters](#) di Referensi AWS CLI Perintah.

## describe-subnet-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-subnet-groups`.

### AWS CLI

Untuk menggambarkan kelompok subnet yang didefinisikan dalam DAX

`describe-subnet-groups` Contoh berikut mengambil rincian untuk kelompok subnet yang didefinisikan dalam DAX.

```
aws dax describe-subnet-groups
```

### Output:

```
{
  "SubnetGroups": [
    {
      "SubnetGroupName": "default",
      "Description": "Default CacheSubnetGroup",
      "VpcId": "vpc-ee70a196",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-874953af",
          "SubnetAvailabilityZone": "us-west-2d"
        },
        {
          "SubnetIdentifier": "subnet-bd3d1fc4",
          "SubnetAvailabilityZone": "us-west-2a"
        },
        {
          "SubnetIdentifier": "subnet-72c2ff28",
          "SubnetAvailabilityZone": "us-west-2c"
        },
        {
```

```

        "SubnetIdentifier": "subnet-09e6aa42",
        "SubnetAvailabilityZone": "us-west-2b"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [DescribeSubnetGroups](#) di Referensi AWS CLI Perintah.

## increase-replication-factor

Contoh kode berikut menunjukkan cara menggunakan `increase-replication-factor`.

### AWS CLI

Untuk meningkatkan faktor replikasi untuk cluster DAX

`increase-replication-factor` Contoh berikut meningkatkan faktor replikasi cluster DAX yang ditentukan menjadi 3.

```

aws dax increase-replication-factor \
  --cluster-name daxcluster \
  --new-replication-factor 3

```

Output:

```

{
  "Cluster": {
    "ClusterName": "daxcluster",
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
    "TotalNodes": 3,
    "ActiveNodes": 1,
    "NodeType": "dax.r4.large",
    "Status": "modifying",
    "ClusterDiscoveryEndpoint": {
      "Address": "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
      "Port": 8111
    },
    "Nodes": [

```

```
{
  "NodeId": "daxcluster-a",
  "Endpoint": {
    "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
    "Port": 8111
  },
  "NodeCreateTime": 1576625059.509,
  "AvailabilityZone": "us-west-2c",
  "NodeStatus": "available",
  "ParameterGroupStatus": "in-sync"
},
{
  "NodeId": "daxcluster-b",
  "NodeStatus": "creating"
},
{
  "NodeId": "daxcluster-c",
  "NodeStatus": "creating"
}
],
"PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
"SubnetGroup": "default",
"SecurityGroups": [
  {
    "SecurityGroupIdentifier": "sg-1af6e36e",
    "Status": "active"
  }
],
"IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
"ParameterGroup": {
  "ParameterGroupName": "default.dax1.0",
  "ParameterApplyStatus": "in-sync",
  "NodeIdsToReboot": []
},
"SSEDescription": {
  "Status": "ENABLED"
}
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [IncreaseReplicationFactor](#) di Referensi AWS CLI Perintah.

## list-tags

Contoh kode berikut menunjukkan cara menggunakan `list-tags`.

### AWS CLI

Untuk mencantumkan tag pada sumber daya DAX

`list-tags` Contoh berikut mencantumkan kunci tag dan nilai yang dilampirkan ke cluster DAX yang ditentukan.

```
aws dax list-tags \  
  --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "ClusterUsage",  
      "Value": "prod"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [ListTags](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya DAX

`tag-resource` Contoh berikut melampirkan nama kunci tag yang ditentukan dan nilai terkait ke cluster DAX yang ditentukan untuk menggambarkan penggunaan klaster.

```
aws dax tag-resource \  
  --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster \  
  --tags="Key=ClusterUsage,Value=prod"
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "ClusterUsage",  
      "Value": "prod"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya DAX

`untag-resource` Contoh berikut menghapus tag dengan nama kunci yang ditentukan dari cluster DAX.

```
aws dax untag-resource \  
  --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster \  
  --tag-keys="ClusterUsage"
```

Output:

```
{
```



```
"Tags": []  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Kluster DAX di Panduan Pengembang](#) Amazon DynamoDB.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Contoh Detektif menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface Detective with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **accept-invitation**

Contoh kode berikut menunjukkan cara menggunakan `accept-invitation`.

### AWS CLI

Untuk menerima undangan untuk menjadi akun anggota dalam grafik perilaku

`accept-invitation` Contoh berikut menerima undangan untuk menjadi akun anggota dalam grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

```
aws detective accept-invitation \
```

```
--graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menanggapi undangan grafik perilaku](#) di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [AcceptInvitation](#) di Referensi AWS CLI Perintah.

## create-graph

Contoh kode berikut menunjukkan cara menggunakan `create-graph`.

### AWS CLI

Untuk mengaktifkan Amazon Detective dan membuat grafik perilaku baru

`create-graph` Contoh berikut memungkinkan Detektif untuk AWS akun yang menjalankan perintah di Wilayah tempat perintah dijalankan. Grafik perilaku baru dibuat yang memiliki akun itu sebagai akun administrasinya. Perintah tersebut juga memberikan nilai Keuangan ke tag Departemen.

```
aws detective create-graph \  
  --tags '{"Department": "Finance"}
```

Output:

```
{  
  "GraphArn": "arn:aws:detective:us-  
east-1:111122223333:graph:027c7c4610ea4aacaf0b883093cab899"  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Detektif Amazon](#) di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [CreateGraph](#) di Referensi AWS CLI Perintah.

## create-members

Contoh kode berikut menunjukkan cara menggunakan `create-members`.

## AWS CLI

Untuk mengundang akun anggota ke grafik perilaku

`create-members` Contoh berikut mengundang dua AWS akun untuk menjadi akun anggota dalam grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Untuk setiap akun, permintaan tersebut memberikan ID AWS akun dan alamat email pengguna root akun. Permintaan termasuk pesan khusus untuk dimasukkan ke dalam email undangan.

```
aws detective create-members \  
  --accounts AccountId=444455556666,EmailAddress=mmajor@example.com \  
  AccountId=123456789012,EmailAddress=jstiles@example.com \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \  
  --message "This is Paul Santos. I need to add your account to the data we use \  
  for security investigation in Amazon Detective. If you have any questions, contact \  
  me at psantos@example.com."
```

Output:

```
{  
  "Members": [  
    {  
      "AccountId": "444455556666",  
      "AdministratorId": "111122223333",  
      "EmailAddress": "mmajor@example.com",  
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",  
      "InvitedTime": 1579826107000,  
      "MasterId": "111122223333",  
      "Status": "INVITED",  
      "UpdatedTime": 1579826107000  
    },  
    {  
      "AccountId": "123456789012",  
      "AdministratorId": "111122223333",  
      "EmailAddress": "jstiles@example.com",  
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",  
      "InvitedTime": 1579826107000,  
      "MasterId": "111122223333",  
      "Status": "VERIFICATION_IN_PROGRESS",  
      "UpdatedTime": 1579826107000  
    }  
  ],  
}
```

```
"UnprocessedAccounts": [ ]
}
```

Untuk informasi selengkapnya, lihat Mengundang akun anggota ke grafik perilaku < <https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-add-member-accounts.html> > di Panduan Administrasi Detektif Amazon.

Untuk mengundang akun anggota tanpa mengirim email undangan

`create-members` Contoh berikut mengundang dua AWS akun untuk menjadi akun anggota dalam grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Untuk setiap akun, permintaan tersebut memberikan ID AWS akun dan alamat email pengguna root akun. Akun anggota tidak menerima email undangan.

```
aws detective create-members \
  --accounts AccountId=444455556666,EmailAddress=mmajor@example.com
  AccountId=123456789012,EmailAddress=jstiles@example.com \
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
  --disable-email-notification
```

Output:

```
{
  "Members": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    },
    {
      "AccountId": "123456789012",
      "AdministratorId": "111122223333",
      "EmailAddress": "jstiles@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",

```

```
    "Status": "VERIFICATION_IN_PROGRESS",
    "UpdateTime": 1579826107000
  },
  "UnprocessedAccounts": [ ]
}
```

Untuk informasi selengkapnya, lihat Mengundang akun anggota ke grafik perilaku < <https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-add-member-accounts.html> > di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [CreateMembers](#) di Referensi AWS CLI Perintah.

## delete-graph

Contoh kode berikut menunjukkan cara menggunakan `delete-graph`.

### AWS CLI

Untuk menonaktifkan Detektif dan menghapus grafik perilaku

`delete-graph` Contoh berikut menonaktifkan Detektif dan menghapus grafik perilaku yang ditentukan.

```
aws detective delete-graph \
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan Detektif Amazon di Panduan Administrasi Detektif Amazon](#).

- Untuk detail API, lihat [DeleteGraph](#) di Referensi AWS CLI Perintah.

## delete-members

Contoh kode berikut menunjukkan cara menggunakan `delete-members`.

### AWS CLI

Untuk menghapus akun anggota dari grafik perilaku

`delete-members` Contoh berikut menghapus dua akun anggota dari grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Untuk mengidentifikasi akun, permintaan memberikan ID AWS akun.

```
aws detective delete-members \  
  --account-ids 444455556666 123456789012 \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{  
  "AccountIds": [ "444455556666", "123456789012" ],  
  "UnprocessedAccounts": [ ]  
}
```

Untuk informasi selengkapnya, lihat Menghapus akun anggota dari grafik perilaku < <https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-remove-member-accounts.html> > di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [DeleteMembers](#) di Referensi AWS CLI Perintah.

## **disassociate-membership**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-membership`.

### AWS CLI

Untuk mengundurkan diri dari grafik perilaku

Contoh `disassociate-membership` berikut menghapus AWS akun yang menjalankan perintah dari grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

```
aws detective disassociate-membership \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Untuk informasi selengkapnya, lihat Menghapus akun Anda dari grafik perilaku < <https://docs.aws.amazon.com/detective/latest/adminguide/member-remove-self-from-graph.html> > di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [DisassociateMembership](#) di Referensi AWS CLI Perintah.

## get-members

Contoh kode berikut menunjukkan cara menggunakan `get-members`.

### AWS CLI

Untuk mengambil informasi tentang grafik perilaku yang dipilih akun anggota

`get-members` Contoh berikut mengambil informasi tentang dua akun anggota dalam grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Untuk kedua akun tersebut, permintaan tersebut memberikan ID AWS akun.

```
aws detective get-members \
  --account-ids 444455556666 123456789012 \
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{
  "MemberDetails": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    }
    {
      "AccountId": "123456789012",
      "AdministratorId": "111122223333",
      "EmailAddress": "jstiles@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    }
  ],
  "UnprocessedAccounts": [ ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat Melihat daftar akun dalam grafik perilaku < <https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-view-accounts.html>> di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [GetMembers](#) di Referensi AWS CLI Perintah.

## list-graphs

Contoh kode berikut menunjukkan cara menggunakan `list-graphs`.

### AWS CLI

Untuk melihat daftar grafik perilaku yang menjadi administrator akun Anda

`list-graphs` Contoh berikut mengambil grafik perilaku bahwa akun panggilan adalah administrator untuk dalam Wilayah saat ini.

```
aws detective list-graphs
```

Output:

```
{
  "GraphList": [
    {
      "Arn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "CreatedTime": 1579736111000
    }
  ]
}
```

- Untuk detail API, lihat [ListGraphs](#) di Referensi AWS CLI Perintah.

## list-invitations

Contoh kode berikut menunjukkan cara menggunakan `list-invitations`.

### AWS CLI

Untuk melihat daftar grafik perilaku yang merupakan anggota akun atau diundang



`list-invitations` Contoh berikut mengambil grafik perilaku yang telah diundang ke akun panggilan. Hasilnya hanya mencakup undangan terbuka dan diterima. Mereka tidak termasuk undangan yang ditolak atau keanggotaan yang dihapus.

```
aws detective list-invitations
```

Output:

```
{
  "Invitations": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat daftar undangan grafik perilaku Anda](https://docs.aws.amazon.com/detective/latest/adminguide/member-view-graph-invitations.html) < <https://docs.aws.amazon.com/detective/latest/adminguide/member-view-graph-invitations.html> > di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [ListInvitations](#) di Referensi AWS CLI Perintah.

## **list-members**

Contoh kode berikut menunjukkan cara menggunakan `list-members`.

### AWS CLI

Untuk mencantumkan akun anggota dalam grafik perilaku

`list-members` Contoh berikut mengambil akun anggota yang diundang dan diaktifkan untuk grafik `arn:aws:detective:us-east-1:111122223333:graph:123412341234` perilaku. Hasilnya tidak termasuk akun anggota yang dihapus.

```
aws detective list-members \
```

```
--graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{
  "MemberDetails": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-
east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    },
    {
      "AccountId": "123456789012",
      "AdministratorId": "111122223333",
      "EmailAddress": "jstiles@example.com",
      "GraphArn": "arn:aws:detective:us-
east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "PercentOfGraphUtilization": 2,
      "PercentOfGraphUtilizationUpdatedTime": 1586287843,
      "Status": "ENABLED",
      "UpdatedTime": 1579973711000,
      "VolumeUsageInBytes": 200,
      "VolumeUsageUpdatedTime": 1586287843
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat daftar akun dalam grafik perilaku](#) di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [ListMembers](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk mengambil tag yang ditetapkan ke grafik perilaku

`list-tags-for-resource` Contoh berikut mengembalikan tag yang ditetapkan untuk grafik perilaku tertentu.

```
aws detective list-tags-for-resource \  
  --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{  
  "Tags": {  
    "Department" : "Finance"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola tag untuk grafik perilaku](#) di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **reject-invitation**

Contoh kode berikut menunjukkan cara menggunakan `reject-invitation`.

## AWS CLI

Untuk menolak undangan untuk menjadi akun anggota dalam grafik perilaku

`reject-invitation` Contoh berikut menolak undangan untuk menjadi akun anggota dalam grafik perilaku `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

```
aws detective reject-invitation \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat Menanggapi undangan grafik perilaku < <https://docs.aws.amazon.com/detective/latest/adminguide/member-invitation-response.html> > di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [RejectInvitation](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menetapkan tag ke sumber daya

`tag-resource` Contoh berikut memberikan nilai untuk tag Department ke grafik perilaku yang ditentukan.

```
aws detective tag-resource \  
  --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \  
  --tags '{"Department":"Finance"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola tag untuk grafik perilaku](#) di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus nilai tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag Department dari grafik perilaku yang ditentukan.

```
aws detective untag-resource \  
  --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \  
  --tag-keys "Department"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola tag untuk grafik perilaku](#) di Panduan Administrasi Detektif Amazon.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Contoh Device Farm menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Device Farm.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-device-pool**

Contoh kode berikut menunjukkan cara menggunakan `create-device-pool`.

AWS CLI

Untuk membuat kumpulan perangkat

Perintah berikut membuat kumpulan perangkat Android untuk proyek:

```
aws devicefarm create-device-pool --name pool1 --rules file://  
device-pool-rules.json --project-arn "arn:aws:devicefarm:us-  
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506"
```

Anda bisa mendapatkan proyek ARN dari output atau. `create-project list-projects` File tersebut `device-pool-rules.json` adalah dokumen JSON di folder saat ini yang menentukan platform perangkat:

```
[
  {
    "attribute": "PLATFORM",
    "operator": "EQUALS",
    "value": "\"ANDROID\""
  }
]
```

Output:

```
{
  "devicePool": {
    "rules": [
      {
        "operator": "EQUALS",
        "attribute": "PLATFORM",
        "value": "\"ANDROID\""
      }
    ],
    "type": "PRIVATE",
    "name": "pool1",
    "arn": "arn:aws:devicefarm:us-west-2:123456789012:devicepool:070fc3ca-7ec1-4741-9c1f-d3e044efc506/2aa8d2a9-5e73-47ca-b929-659cb34b7dcd"
  }
}
```

- Untuk detail API, lihat [CreateDevicePool](#) di Referensi AWS CLI Perintah.

## create-project

Contoh kode berikut menunjukkan cara menggunakan `create-project`.

### AWS CLI

Untuk membuat proyek

Perintah berikut membuat proyek baru bernama `my-project`:

```
aws devicefarm create-project --name my-project
```

Output:

```
{
  "project": {
    "name": "myproject",
    "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506",
    "created": 1503612890.057
  }
}
```

- Untuk detail API, lihat [CreateProject](#) di Referensi AWS CLI Perintah.

## create-upload

Contoh kode berikut menunjukkan cara menggunakan `create-upload`.

### AWS CLI

Untuk membuat unggahan

Perintah berikut membuat unggahan untuk aplikasi Android:

```
aws devicefarm create-upload --project-arn "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506" --name app.apk --
type ANDROID_APP
```

Anda bisa mendapatkan proyek ARN dari output `create-project` atau `list-projects`.

Output:

```
{
  "upload": {
    "status": "INITIALIZED",
    "name": "app.apk",
    "created": 1503614408.769,
    "url": "https://prod-us-west-2-uploads.s3-us-west-2.amazonaws.com/
arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aproject%3A070fc3ca-
c7e1-4471-91cf-d3e4efc50604/uploads/arn%3Aaws%3Adevicefarm%3Aus-
```

```

west-2%3A123456789012%3Aupload%3A070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514/app.apk?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170824T224008Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-
Credential=AKIAEXAMPLEPBUMBC3GA%2F20170824%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=05050370c38894ef5bd09f5d009f36fc8f96fa4bb04e1bba9aca71b8dbe49a0f",
    "type": "ANDROID_APP",
    "arn": "arn:aws:devicefarm:us-
west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514"
  }
}

```

Gunakan URL yang ditandatangani dalam output untuk mengunggah file ke Device Farm:

```

curl -T app.apk "https://prod-us-west-2-uploads.s3-us-west-2.amazonaws.com/
arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aproject%3A070fc3ca-
c7e1-4471-91cf-d3e4efc50604/uploads/arn%3Aaws%3Adevicefarm%3Aus-
west-2%3A123456789012%3Aupload%3A070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514/app.apk?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170824T224008Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-
Credential=AKIAEXAMPLEPBUMBC3GA%2F20170824%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=05050370c38894ef5bd09f5d009f36fc8f96fa4bb04e1bba9aca71b8dbe49a0f"

```

- Untuk detail API, lihat [CreateUpload](#) di Referensi AWS CLI Perintah.

## get-upload

Contoh kode berikut menunjukkan cara menggunakan `get-upload`.

### AWS CLI

Untuk melihat unggahan

Perintah berikut mengambil informasi tentang unggahan:

```

aws devicefarm get-upload --arn "arn:aws:devicefarm:us-
west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514"

```

Anda bisa mendapatkan upload ARN dari output. `create-upload`

Output:



```
{
  "upload": {
    "status": "SUCCEEDED",
    "name": "app.apk",
    "created": 1505262773.186,
    "type": "ANDROID_APP",
    "arn": "arn:aws:devicefarm:us-
west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514",
    "metadata": "{\"device_admin\":false,\"activity_name\":
\\\"ccom.example.client.LauncherActivity\\\",\\\"version_name\\\":\\\"1.0.2.94\\\",\\\"screens
\\\":[\\\"small\\\",\\\"normal\\\",\\\"large\\\",\\\"xlarge\\\"],\\\"error_type\\\":null,\\\"sdk_version
\\\":\\\"16\\\",\\\"package_name\\\":\\\"com.example.client\\\",\\\"version_code\\\":\\\"20994\\\",
\\\"native_code\\\":[\\\"armeabi-v7a\\\"],\\\"target_sdk_version\\\":\\\"25\\\"}"
  }
}
```

- Untuk detail API, lihat [GetUpload](#) di Referensi AWS CLI Perintah.

## list-projects

Contoh kode berikut menunjukkan cara menggunakan `list-projects`.

### AWS CLI

Untuk daftar proyek

Berikut ini mengambil daftar proyek:

```
aws devicefarm list-projects
```

Output:

```
{
  "projects": [
    {
      "name": "myproject",
      "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506",
      "created": 1503612890.057
    },
    {
```

```
    "name": "otherproject",
    "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:a5f5b752-8098-49d1-86bf-5f7682c1c77e",
    "created": 1505257519.337
  }
]
```

- Untuk detail API, lihat [ListProjects](#) di Referensi AWS CLI Perintah.

## AWS Direct Connect contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Direct Connect.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **accept-direct-connect-gateway-association-proposal**

Contoh kode berikut menunjukkan cara menggunakan `accept-direct-connect-gateway-association-proposal`.

AWS CLI

Untuk menerima proposal asosiasi gateway

Berikut ini `accept-direct-connect-gateway-association-proposal` menerima proposal yang ditentukan.

```
aws directconnect accept-direct-connect-gateway-association-proposal \
  --direct-connect-gateway-id 11460968-4ac1-4fd3-bdb2-00599EXAMPLE \
  --proposal-id cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE \
  --associated-gateway-owner-account 111122223333

{
  "directConnectGatewayAssociation": {
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "111122223333",
    "associationState": "associating",
    "associatedGateway": {
      "id": "tgw-02f776b1a7EXAMPLE",
      "type": "transitGateway",
      "ownerAccount": "111122223333",
      "region": "us-east-1"
    },
    "associationId": "6441f8bf-5917-4279-ade1-9708bEXAMPLE",
    "allowedPrefixesToDirectConnectGateway": [
      {
        "cidr": "192.168.1.0/30"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Menerima atau Menolak Proposal Asosiasi Gateway Transit](#) di Panduan Pengguna Direct AWS Connect.

- Untuk detail API, lihat [AcceptDirectConnectGatewayAssociationProposal](#) di Referensi AWS CLI Perintah.

## allocate-connection-on-interconnect

Contoh kode berikut menunjukkan cara menggunakan `allocate-connection-on-interconnect`.

### AWS CLI

Untuk membuat koneksi yang di-host pada interkoneksi

`allocate-connection-on-interconnect` Perintah berikut membuat koneksi yang di-host pada interkoneksi:

```
aws directconnect allocate-connection-on-interconnect --bandwidth 500Mbps --
connection-name mydcinterconnect --owner-account 123456789012 --interconnect-id
dxcon-fgktov66 --vlan 101
```

### Output:

```
{
  "partnerName": "TIVIT",
  "vlan": 101,
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffzc51m1",
  "connectionState": "ordering",
  "bandwidth": "500Mbps",
  "location": "TIVIT",
  "connectionName": "mydcinterconnect",
  "region": "sa-east-1"
}
```

- Untuk detail API, lihat [AllocateConnectionOnInterconnect](#) di Referensi AWS CLI Perintah.

## allocate-hosted-connection

Contoh kode berikut menunjukkan cara menggunakan `allocate-hosted-connection`.

### AWS CLI

Untuk membuat koneksi yang di-host pada interkoneksi

`allocate-hosted-connection` Contoh berikut membuat koneksi host pada interkoneksi yang ditentukan.

```
aws directconnect allocate-hosted-connection \
  --bandwidth 500Mbps \
  --connection-name mydcinterconnect \
  --owner-account 123456789012
-connection-id dxcon-fgktov66
-vlan 101
```

### Output:

```
{
```

```

    "partnerName": "TIVIT",
    "vlan": 101,
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffzc51m1",
    "connectionState": "ordering",
    "bandwidth": "500Mbps",
    "location": "TIVIT",
    "connectionName": "mydcinterconnect",
    "region": "sa-east-1"
  }

```

- Untuk detail API, lihat [AllocateHostedConnection](#) di Referensi AWS CLI Perintah.

## allocate-private-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `allocate-private-virtual-interface`.

### AWS CLI

Untuk menyediakan antarmuka virtual pribadi

`allocate-private-virtual-interface` Perintah berikut menyediakan antarmuka virtual pribadi yang akan dimiliki oleh pelanggan yang berbeda:

```

aws directconnect allocate-private-virtual-interface --connection-id dxcon-
ffjrjrkx17 --owner-account 123456789012 --new-private-virtual-interface-allocation
virtualInterfaceName=PrivateVirtualInterface,vlan=1000,asn=65000,authKey=asdf34example,amaz

```

Output:

```

{
  "virtualInterfaceState": "confirming",
  "asn": 65000,
  "vlan": 1000,
  "customerAddress": "192.168.1.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrjrkx17",
  "virtualInterfaceId": "dxvif-fgy8orxu",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [],
  "location": "TIVIT",

```

```

    "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n <logical_connection id=\"dxvif-fgy8orxu\">\n <vlan>1000</
vlan>\n <customer_address>192.168.1.2/30</customer_address>\n
<amazon_address>192.168.1.1/30</amazon_address>\n <bgp_asn>65000</bgp_asn>\n
<bgp_auth_key>asdf34example</bgp_auth_key>\n <amazon_bgp_asn>7224</amazon_bgp_asn>
\n <connection_type>private</connection_type>\n</logical_connection>\n",
    "amazonAddress": "192.168.1.1/30",
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "PrivateVirtualInterface"
}

```

- Untuk detail API, lihat [AllocatePrivateVirtualInterface](#) di Referensi AWS CLI Perintah.

## allocate-public-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `allocate-public-virtual-interface`.

### AWS CLI

Untuk menyediakan antarmuka virtual publik

`allocate-public-virtual-interface` Perintah berikut menyediakan antarmuka virtual publik yang akan dimiliki oleh pelanggan yang berbeda:

```

aws directconnect allocate-public-virtual-interface --connection-id dxcon-
ffjrkk17 --owner-account 123456789012 --new-public-virtual-interface-allocation
virtualInterfaceName=PublicVirtualInterface,vlan=2000,asn=65000,authKey=asdf34example,amazo
{cidr=203.0.113.4/30}]

```

Output:

```

{
  "virtualInterfaceState": "confirming",
  "asn": 65000,
  "vlan": 2000,
  "customerAddress": "203.0.113.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrkk17",
  "virtualInterfaceId": "dxvif-fg9xo9vp",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [
    {

```

```

        "cidr": "203.0.113.0/30"
      },
      {
        "cidr": "203.0.113.4/30"
      }
    ],
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fg9xo9vp\">\n  <vlan>2000</
vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n
  <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>public</connection_type>\n</logical_connection>\n",
    "amazonAddress": "203.0.113.1/30",
    "virtualInterfaceType": "public",
    "virtualInterfaceName": "PublicVirtualInterface"
  }
}

```

- Untuk detail API, lihat [AllocatePublicVirtualInterface](#) di Referensi AWS CLI Perintah.

## allocate-transit-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `allocate-transit-virtual-interface`.

### AWS CLI

Untuk menyediakan antarmuka virtual transit yang akan dimiliki oleh AWS akun yang ditentukan

`allocate-transit-virtual-interface` Contoh berikut menyediakan antarmuka virtual transit untuk akun yang ditentukan.

```

aws directconnect allocate-transit-virtual-interface \
  --connection-id dxlag-fEXAMPLE \
  --owner-account 123456789012 \
  --new-transit-virtual-interface-allocation "virtualInterfaceName=Example Transit
Virtual
Interface,vlan=126,asn=65110,mtu=1500,authKey=0xzxcgA9YoW9h58u8SEXAMPLE,amazonAddress=192.16

```

Output:

```

{
  "virtualInterface": {

```

```

"ownerAccount": "123456789012",
"virtualInterfaceId": "dxvif-fEXAMPLE",
"location": "loc1",
"connectionId": "dxlag-fEXAMPLE",
"virtualInterfaceType": "transit",
"virtualInterfaceName": "Example Transit Virtual Interface",
"vlan": 126,
"asn": 65110,
"amazonSideAsn": 7224,
"authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
"amazonAddress": "192.168.1.1/30",
"customerAddress": "192.168.1.2/30",
"addressFamily": "ipv4",
"virtualInterfaceState": "confirming",
"customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fEXAMPLE\">\n  <vlan>126</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
  <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65110</bgp_asn>\n
  <bgp_auth_key>0xzxcgA9YoW9h58u8SEXAMPLE</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>transit</connection_type>\n</logical_connection>
\n",
"mtu": 1500,
"jumboFrameCapable": true,
"virtualGatewayId": "",
"directConnectGatewayId": "",
"routeFilterPrefixes": [],
"bgpPeers": [
  {
    "bgpPeerId": "dxpeer-fEXAMPLE",
    "asn": 65110,
    "authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
    "addressFamily": "ipv4",
    "amazonAddress": "192.168.1.1/30",
    "customerAddress": "192.168.1.2/30",
    "bgpPeerState": "pending",
    "bgpStatus": "down",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE"
  }
],
"region": "sa-east-1",
"awsDeviceV2": "loc1-26wz6vEXAMPLE",
"tags": [
  {
    "key": "Tag",

```



```
        "value": "Example"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Antarmuka Virtual Transit yang Dihosting](#) di Panduan Pengguna AWS Direct Connect.

- Untuk detail API, lihat [AllocateTransitVirtualInterface](#) di Referensi AWS CLI Perintah.

## associate-connection-with-lag

Contoh kode berikut menunjukkan cara menggunakan `associate-connection-with-lag`.

### AWS CLI

Untuk mengaitkan koneksi dengan LAG

Contoh berikut mengaitkan koneksi yang ditentukan dengan LAG yang ditentukan.

Perintah:

```
aws directconnect associate-connection-with-lag --lag-id dxlag-fhccu14t --
connection-id dxcon-fg9607vm
```

Output:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "lagId": "dxlag-fhccu14t",
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "EqDC2",
  "connectionName": "Con2ForLag",
  "region": "us-east-1"
}
```

- Untuk detail API, lihat [AssociateConnectionWithLag](#) di Referensi AWS CLI Perintah.

## associate-hosted-connection

Contoh kode berikut menunjukkan cara menggunakan `associate-hosted-connection`.

### AWS CLI

Untuk mengaitkan koneksi yang dihosting dengan LAG

Contoh berikut mengaitkan koneksi host yang ditentukan dengan LAG yang ditentukan.

Perintah:

```
aws directconnect associate-hosted-connection --parent-connection-id dxlag-fhccu14t
--connection-id dxcon-fg9607vm
```

Output:

```
{
  "partnerName": "TIVIT",
  "vlan": 101,
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "lagId": "dxlag-fhccu14t",
  "connectionState": "ordering",
  "bandwidth": "500Mbps",
  "location": "TIVIT",
  "connectionName": "mydcinterconnect",
  "region": "sa-east-1"
}
```

- Untuk detail API, lihat [AssociateHostedConnection](#) di Referensi AWS CLI Perintah.

## associate-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `associate-virtual-interface`.

### AWS CLI

Untuk mengaitkan antarmuka virtual dengan koneksi

Contoh berikut mengaitkan antarmuka virtual yang ditentukan dengan LAG yang ditentukan. Atau, untuk mengaitkan antarmuka virtual dengan koneksi, tentukan ID koneksi AWS Direct Connect untuk `--connection-id`; misalnya, `dxcon-ffnikghc`.

**Perintah:**

```
aws directconnect associate-virtual-interface --connection-id dxlag-ffjhj91x --
virtual-interface-id dxvif-fgputw0j
```

**Output:**

```
{
  "virtualInterfaceState": "pending",
  "asn": 65000,
  "vlan": 123,
  "customerAddress": "169.254.255.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxlag-ffjhj91x",
  "addressFamily": "ipv4",
  "virtualGatewayId": "vgw-38e90b51",
  "virtualInterfaceId": "dxvif-fgputw0j",
  "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
  "routeFilterPrefixes": [],
  "location": "CSVA1",
  "bgpPeers": [
    {
      "bgpStatus": "down",
      "customerAddress": "169.254.255.2/30",
      "addressFamily": "ipv4",
      "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
      "bgpPeerState": "deleting",
      "amazonAddress": "169.254.255.1/30",
      "asn": 65000
    },
    {
      "bgpStatus": "down",
      "customerAddress": "169.254.255.2/30",
      "addressFamily": "ipv4",
      "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
      "bgpPeerState": "pending",
      "amazonAddress": "169.254.255.1/30",
      "asn": 65000
    }
  ],
  "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<n<logical_connection id=\"dxvif-fgputw0j\">
  <vlan>123</vlan>
  <n
  <customer_address>169.254.255.2/30</customer_address>
  <n
```

```
<amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>0x123pK5_VBqv.UQ3kJ4123_</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>private</connection_type>\n</logical_connection>
\n",
  "amazonAddress": "169.254.255.1/30",
  "virtualInterfaceType": "private",
  "virtualInterfaceName": "VIF1A"
}
```

- Untuk detail API, lihat [AssociateVirtualInterface](#) di Referensi AWS CLI Perintah.

## confirm-connection

Contoh kode berikut menunjukkan cara menggunakan `confirm-connection`.

### AWS CLI

Untuk mengonfirmasi pembuatan koneksi yang dihosting pada interkoneksi

`confirm-connection` Perintah berikut mengonfirmasi pembuatan koneksi yang dihosting pada interkoneksi:

```
aws directconnect confirm-connection --connection-id dxcon-fg2wi7hy
```

Output:

```
{
  "connectionState": "pending"
}
```

- Untuk detail API, lihat [ConfirmConnection](#) di Referensi AWS CLI Perintah.

## confirm-private-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `confirm-private-virtual-interface`.

### AWS CLI

Untuk menerima kepemilikan antarmuka virtual pribadi

`confirm-private-virtual-interface` Perintah berikut menerima kepemilikan antarmuka virtual pribadi yang dibuat oleh pelanggan lain:

```
aws directconnect confirm-private-virtual-interface --virtual-interface-id dxvif-  
fgy8orxu --virtual-gateway-id vgw-e4a47df9
```

Output:

```
{  
  "virtualInterfaceState": "pending"  
}
```

- Untuk detail API, lihat [ConfirmPrivateVirtualInterface](#) di Referensi AWS CLI Perintah.

### **confirm-public-virtual-interface**

Contoh kode berikut menunjukkan cara menggunakan `confirm-public-virtual-interface`.

AWS CLI

Untuk menerima kepemilikan antarmuka virtual publik

`confirm-public-virtual-interface` Perintah berikut menerima kepemilikan antarmuka virtual publik yang dibuat oleh pelanggan lain:

```
aws directconnect confirm-public-virtual-interface --virtual-interface-id dxvif-  
fg9xo9vp
```

Output:

```
{  
  "virtualInterfaceState": "verifying"  
}
```

- Untuk detail API, lihat [ConfirmPublicVirtualInterface](#) di Referensi AWS CLI Perintah.

### **confirm-transit-virtual-interface**

Contoh kode berikut menunjukkan cara menggunakan `confirm-transit-virtual-interface`.

AWS CLI

Untuk menerima kepemilikan antarmuka virtual transit

Berikut ini `confirm-transit-virtual-interface` menerima kepemilikan antarmuka virtual transit yang dibuat oleh pelanggan lain.

```
aws directconnect confirm-transit-virtual-interface \  
  --virtual-interface-id dxvif-fEXAMPLE \  
  --direct-connect-gateway-id 4112ccf9-25e9-4111-8237-b6c5dEXAMPLE
```

Output:

```
{  
  "virtualInterfaceState": "pending"  
}
```

Untuk informasi selengkapnya, lihat [Menerima Antarmuka Virtual yang Dihosting](#) di Panduan Pengguna AWS Direct Connect.

- Untuk detail API, lihat [ConfirmTransitVirtualInterface](#) di Referensi AWS CLI Perintah.

## create-bgp-peer

Contoh kode berikut menunjukkan cara menggunakan `create-bgp-peer`.

### AWS CLI

Untuk membuat sesi peering IPv6 BGP

Contoh berikut membuat sesi peering IPv6 BGP pada antarmuka virtual pribadi. `dxvif-fg1vuj3d` Alamat IPv6 peer secara otomatis dialokasikan oleh Amazon.

Perintah:

```
aws directconnect create-bgp-peer --virtual-interface-id dxvif-fg1vuj3d --new-bgp-peer asn=64600,addressFamily=ipv6
```

Output:

```
{  
  "virtualInterface": {  
    "virtualInterfaceState": "available",  
    "asn": 65000,  
    "vlan": 125,  
    "customerAddress": "169.254.255.2/30",
```

```

"ownerAccount": "123456789012",
"connectionId": "dxcon-fguhmqlc",
"addressFamily": "ipv4",
"virtualGatewayId": "vgw-f9eb0c90",
"virtualInterfaceId": "dxvif-fg1vuj3d",
"authKey": "0xC_ukbCerl6EYA0example",
"routeFilterPrefixes": [],
"location": "EqDC2",
"bgpPeers": [
  {
    "bgpStatus": "down",
    "customerAddress": "169.254.255.2/30",
    "addressFamily": "ipv4",
    "authKey": "0xC_ukbCerl6EYA0uexample",
    "bgpPeerState": "available",
    "amazonAddress": "169.254.255.1/30",
    "asn": 65000
  },
  {
    "bgpStatus": "down",
    "customerAddress": "2001:db8:1100:2f0:0:1:9cb4:4216/125",
    "addressFamily": "ipv6",
    "authKey": "0xS27kAIU_VHPjjAexample",
    "bgpPeerState": "pending",
    "amazonAddress": "2001:db8:1100:2f0:0:1:9cb4:4211/125",
    "asn": 64600
  }
],
"customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fg1vuj3d\">\n  <vlan>125</
vlan>\n  <customer_address>169.254.255.2/30</customer_address>\n
  <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</
bgp_asn>\n  <bgp_auth_key>0xC_ukbCerl6EYA0uexample</bgp_auth_key>\n
  <ipv6_customer_address>2001:db8:1100:2f0:0:1:9cb4:4216/125</ipv6_customer_address>
\n  <ipv6_amazon_address>2001:db8:1100:2f0:0:1:9cb4:4211/125</ipv6_amazon_address>\n
  <ipv6_bgp_asn>64600</ipv6_bgp_asn>\n  <ipv6_bgp_auth_key>0xS27kAIU_VHPjjAexample</
ipv6_bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>\n
  <connection_type>private</connection_type>\n</logical_connection>\n",
  "amazonAddress": "169.254.255.1/30",
  "virtualInterfaceType": "private",
  "virtualInterfaceName": "Test"
}
}

```

- Untuk detail API, lihat [CreateBgpPeer](#) di Referensi AWS CLI Perintah.

## **create-connection**

Contoh kode berikut menunjukkan cara menggunakan `create-connection`.

### AWS CLI

Untuk membuat sambungan dari jaringan Anda ke lokasi AWS Direct Connect

`create-connection` Perintah berikut membuat koneksi dari jaringan Anda ke lokasi AWS Direct Connect:

```
aws directconnect create-connection --location TIVIT --bandwidth 1Gbps --connection-name "Connection to AWS"
```

Output:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg31dyv6",
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "TIVIT",
  "connectionName": "Connection to AWS",
  "region": "sa-east-1"
}
```

- Untuk detail API, lihat [CreateConnection](#) di Referensi AWS CLI Perintah.

## **create-direct-connect-gateway-association-proposal**

Contoh kode berikut menunjukkan cara menggunakan `create-direct-connect-gateway-association-proposal`.

### AWS CLI

Untuk membuat proposal untuk mengaitkan gateway transit yang ditentukan dengan gateway Direct Connect yang ditentukan



`create-direct-connect-gateway-association-proposal` Contoh berikut membuat proposal yang mengaitkan gateway transit yang ditentukan dengan gateway Direct Connect yang ditentukan.

```
aws directconnect create-direct-connect-gateway-association-proposal \
  --direct-connect-gateway-id 11460968-4ac1-4fd3-bdb2-00599EXAMPLE \
  --direct-connect-gateway-owner-account 111122223333 \
  --gateway-id tgw-02f776b1a7EXAMPLE \
  --add-allowed-prefixes-to-direct-connect-gateway cidr=192.168.1.0/30
```

Output:

```
{
  "directConnectGatewayAssociationProposal": {
    "proposalId": "cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE",
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "111122223333",
    "proposalState": "requested",
    "associatedGateway": {
      "id": "tgw-02f776b1a7EXAMPLE",
      "type": "transitGateway",
      "ownerAccount": "111122223333",
      "region": "us-east-1"
    },
    "requestedAllowedPrefixesToDirectConnectGateway": [
      {
        "cidr": "192.168.1.0/30"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Proposal Asosiasi Gateway Transit](#) di Panduan Pengguna AWS Direct Connect.

- Untuk detail API, lihat [CreateDirectConnectGatewayAssociationProposal](#) di Referensi AWS CLI Perintah.

## **create-direct-connect-gateway-association**

Contoh kode berikut menunjukkan cara menggunakan `create-direct-connect-gateway-association`.

## AWS CLI

Untuk mengaitkan gateway pribadi virtual dengan gateway Direct Connect

Contoh berikut mengaitkan gateway pribadi virtual vgw-6efe725e dengan gateway Direct Connect. 5f294f92-bafb-4011-916d-9b0bexample Anda harus menjalankan perintah di wilayah tempat gateway pribadi virtual berada.

Perintah:

```
aws directconnect create-direct-connect-gateway-association --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample --virtual-gateway-id vgw-6efe725e
```

Output:

```
{
  "directConnectGatewayAssociation": {
    "associationState": "associating",
    "virtualGatewayOwnerAccount": "123456789012",
    "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
    "virtualGatewayId": "vgw-6efe725e",
    "virtualGatewayRegion": "us-east-2"
  }
}
```

- Untuk detail API, lihat [CreateDirectConnectGatewayAssociation](#) di Referensi AWS CLI Perintah.

## **create-direct-connect-gateway**

Contoh kode berikut menunjukkan cara menggunakan `create-direct-connect-gateway`.

## AWS CLI

Untuk membuat gateway Direct Connect

Contoh berikut membuat gateway Direct Connect dengan nama `DxGateway1`.

Perintah:

```
aws directconnect create-direct-connect-gateway --direct-connect-gateway-name "DxGateway1"
```

**Output:**

```
{
  "directConnectGateway": {
    "amazonSideAsn": 64512,
    "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bdexample",
    "ownerAccount": "123456789012",
    "directConnectGatewayName": "DxGateway1",
    "directConnectGatewayState": "available"
  }
}
```

- Untuk detail API, lihat [CreateDirectConnectGateway](#) di Referensi AWS CLI Perintah.

**create-interconnect**

Contoh kode berikut menunjukkan cara menggunakan `create-interconnect`.

**AWS CLI**

Untuk membuat interkoneksi antara jaringan mitra dan AWS

`create-interconnect` Perintah berikut membuat interkoneksi antara jaringan mitra AWS Direct Connect dan lokasi AWS Direct Connect tertentu:

```
aws directconnect create-interconnect --interconnect-name "1G Interconnect to AWS"
--bandwidth 1Gbps --location TIVIT
```

**Output:**

```
{
  "region": "sa-east-1",
  "bandwidth": "1Gbps",
  "location": "TIVIT",
  "interconnectName": "1G Interconnect to AWS",
  "interconnectId": "dxcon-fgktov66",
  "interconnectState": "requested"
}
```

- Untuk detail API, lihat [CreateInterconnect](#) di Referensi AWS CLI Perintah.

## create-lag

Contoh kode berikut menunjukkan cara menggunakan `create-lag`.

### AWS CLI

Untuk membuat LAG dengan koneksi baru

Contoh berikut membuat LAG dan meminta dua koneksi AWS Direct Connect baru untuk LAG dengan bandwidth 1 Gbps.

Perintah:

```
aws directconnect create-lag --location CSVA1 --number-of-connections 2 --
connections-bandwidth 1Gbps --lag-name 1GBLag
```

Output:

```
{
  "awsDevice": "CSVA1-23u8t1paz8iks",
  "numberOfConnections": 2,
  "lagState": "pending",
  "ownerAccount": "123456789012",
  "lagName": "1GBLag",
  "connections": [
    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffqr6x5q",
      "lagId": "dxlag-ffjhj9lx",
      "connectionState": "requested",
      "bandwidth": "1Gbps",
      "location": "CSVA1",
      "connectionName": "Requested Connection 1 for Lag dxlag-ffjhj9lx",
      "region": "us-east-1"
    },
    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-fflqyj95",
      "lagId": "dxlag-ffjhj9lx",
      "connectionState": "requested",
      "bandwidth": "1Gbps",
      "location": "CSVA1",
      "connectionName": "Requested Connection 2 for Lag dxlag-ffjhj9lx",
      "region": "us-east-1"
    }
  ]
}
```

```
    }  
  ],  
  "lagId": "dxlag-ffjhj91x",  
  "minimumLinks": 0,  
  "connectionsBandwidth": "1Gbps",  
  "region": "us-east-1",  
  "location": "CSVA1"  
}
```

Untuk membuat LAG menggunakan koneksi yang ada

Contoh berikut membuat LAG dari koneksi yang ada di akun Anda, dan meminta koneksi baru kedua untuk LAG dengan bandwidth dan lokasi yang sama dengan koneksi yang ada.

Perintah:

```
aws directconnect create-lag --location EqDC2 --number-of-connections 2 --  
connections-bandwidth 1Gbps --lag-name 2ConnLAG --connection-id dxcon-fgk145dr
```

Output:

```
{  
  "awsDevice": "EqDC2-4h6ce2r1bes6",  
  "numberOfConnections": 2,  
  "lagState": "pending",  
  "ownerAccount": "123456789012",  
  "lagName": "2ConnLAG",  
  "connections": [  
    {  
      "ownerAccount": "123456789012",  
      "connectionId": "dxcon-fh61jcv0",  
      "lagId": "dxlag-fhccu14t",  
      "connectionState": "requested",  
      "bandwidth": "1Gbps",  
      "location": "EqDC2",  
      "connectionName": "Requested Connection 1 for Lag dxlag-fhccu14t",  
      "region": "us-east-1"  
    },  
    {  
      "ownerAccount": "123456789012",  
      "connectionId": "dxcon-fgk145dr",  
      "lagId": "dxlag-fhccu14t",  
      "connectionState": "down",  
    }  
  ]  
}
```

```

        "bandwidth": "1Gbps",
        "location": "EqDC2",
        "connectionName": "VAConn1",
        "region": "us-east-1"
    }
],
"lagId": "dxlag-fhccu14t",
"minimumLinks": 0,
"connectionsBandwidth": "1Gbps",
"region": "us-east-1",
"location": "EqDC2"
}

```

- Untuk detail API, lihat [CreateLag](#) di Referensi AWS CLI Perintah.

## create-private-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `create-private-virtual-interface`.

### AWS CLI

Untuk membuat antarmuka virtual pribadi

`create-private-virtual-interface` Perintah berikut membuat antarmuka virtual pribadi:

```

aws directconnect create-private-virtual-interface --
connection-id dxcon-ffjrkx17 --new-private-virtual-interface
virtualInterfaceName=PrivateVirtualInterface,vlan=101,asn=65000,authKey=asdf34example,amazon
aba37db6

```

Output:

```

{
  "virtualInterfaceState": "pending",
  "asn": 65000,
  "vlan": 101,
  "customerAddress": "192.168.1.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrkx17",
  "virtualGatewayId": "vgw-aba37db6",
  "virtualInterfaceId": "dxvif-ffhkh74f",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [],

```

```

    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\\\"UTF-8\\\"?>\\n<logical_connection id=\"dxvif-ffhkh74f\">\\n  <vlan>101</
vlan>\\n  <customer_address>192.168.1.2/30</customer_address>\\n
  <amazon_address>192.168.1.1/30</amazon_address>\\n  <bgp_asn>65000</bgp_asn>\\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\\n  <connection_type>private</connection_type>\\n</logical_connection>\\n",
    "amazonAddress": "192.168.1.1/30",
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "PrivateVirtualInterface"
  }

```

- Untuk detail API, lihat [CreatePrivateVirtualInterface](#) di Referensi AWS CLI Perintah.

## create-public-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `create-public-virtual-interface`.

### AWS CLI

Untuk membuat antarmuka virtual publik

`create-public-virtual-interface` Perintah berikut membuat antarmuka virtual publik:

```

aws directconnect create-public-virtual-interface --
connection-id dxcon-ffjrkrx17 --new-public-virtual-interface
virtualInterfaceName=PublicVirtualInterface,vlan=2000,asn=65000,authKey=asdf34example,amazonAddress=203.0.113.4/30

```

Output:

```

{
  "virtualInterfaceState": "verifying",
  "asn": 65000,
  "vlan": 2000,
  "customerAddress": "203.0.113.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrkrx17",
  "virtualInterfaceId": "dxvif-fgh0hcrk",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [
    {
      "cidr": "203.0.113.0/30"
    }
  ]
}

```

```

    },
    {
      "cidr": "203.0.113.4/30"
    }
  ],
  "location": "TIVIT",
  "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<n<logical_connection id=\"dxvif-fgh0hcrk\">
  <vlan>2000</vlan>
  <customer_address>203.0.113.2/30</customer_address>
  <amazon_address>203.0.113.1/30</amazon_address>
  <bgp_asn>65000</bgp_asn>
  <bgp_auth_key>asdf34example</bgp_auth_key>
  <amazon_bgp_asn>7224</amazon_bgp_asn>
</logical_connection>
",
  "amazonAddress": "203.0.113.1/30",
  "virtualInterfaceType": "public",
  "virtualInterfaceName": "PublicVirtualInterface"
}

```

- Untuk detail API, lihat [CreatePublicVirtualInterface](#) di Referensi AWS CLI Perintah.

## create-transit-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `create-transit-virtual-interface`.

### AWS CLI

Untuk membuat antarmuka virtual transit

`create-transit-virtual-interface` Contoh berikut membuat antarmuka virtual transit untuk koneksi yang ditentukan.

```

aws directconnect create-transit-virtual-interface \
  --connection-id dxlag-fEXAMPLE \
  --new-transit-virtual-interface "virtualInterfaceName=Example Transit Virtual
Interface,vlan=126,asn=65110,mtu=1500,authKey=0xxzxA9YoW9h58u8SvEXAMPLE,amazonAddress=192.1
aada-5a1baEXAMPLE,tags=[{key=Tag,value=Example}]"

```

Output:

```

{
  "virtualInterface": {
    "ownerAccount": "1111222233333",
    "virtualInterfaceId": "dxvif-fEXAMPLE",

```



```

"location": "loc1",
"connectionId": "dxlag-fEXAMPLE",
"virtualInterfaceType": "transit",
"virtualInterfaceName": "Example Transit Virtual Interface",
"vlan": 126,
"asn": 65110,
"amazonSideAsn": 4200000000,
"authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
"amazonAddress": "192.168.1.1/30",
"customerAddress": "192.168.1.2/30",
"addressFamily": "ipv4",
"virtualInterfaceState": "pending",
"customerRouterConfig": "<?xml version='1.0' encoding=
\"UTF-8'?'>\n<logical_connection id='dxvif-fEXAMPLE'>\n  <vlan>126</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
  <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65110</
bgp_asn>\n  <bgp_auth_key>0xzxcgA9YoW9h58u8Sv0mXRTw</bgp_auth_key>\n
  <amazon_bgp_asn>4200000000</amazon_bgp_asn>\n  <connection_type>transit</
connection_type>\n</logical_connection>\n",
"mtu": 1500,
"jumboFrameCapable": true,
"virtualGatewayId": "",
"directConnectGatewayId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
"routeFilterPrefixes": [],
"bgpPeers": [
  {
    "bgpPeerId": "dxpeer-EXAMPLE",
    "asn": 65110,
    "authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
    "addressFamily": "ipv4",
    "amazonAddress": "192.168.1.1/30",
    "customerAddress": "192.168.1.2/30",
    "bgpPeerState": "pending",
    "bgpStatus": "down",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE"
  }
],
"region": "sa-east-1",
"awsDeviceV2": "loc1-26wz6vEXAMPLE",
"tags": [
  {
    "key": "Tag",
    "value": "Example"
  }
]

```

```
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Antarmuka Virtual Transit ke Gateway Direct Connect](#) di Panduan Pengguna AWS Direct Connect.

- Untuk detail API, lihat [CreateTransitVirtualInterface](#) di Referensi AWS CLI Perintah.

## delete-bgp-peer

Contoh kode berikut menunjukkan cara menggunakan `delete-bgp-peer`.

### AWS CLI

Untuk menghapus peer BGP dari antarmuka virtual

Contoh berikut menghapus IPv6 BGP peer dari antarmuka virtual. `dxvif-fg1vuj3d`

Perintah:

```
aws directconnect delete-bgp-peer --virtual-interface-id dxvif-fg1vuj3d --asn 64600
--customer-address 2001:db8:1100:2f0:0:1:9cb4:4216/125
```

Output:

```
{
  "virtualInterface": {
    "virtualInterfaceState": "available",
    "asn": 65000,
    "vlan": 125,
    "customerAddress": "169.254.255.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fguhmq1c",
    "addressFamily": "ipv4",
    "virtualGatewayId": "vgw-f9eb0c90",
    "virtualInterfaceId": "dxvif-fg1vuj3d",
    "authKey": "0xC_ukbCer16EYA0example",
    "routeFilterPrefixes": [],
    "location": "EqDC2",
    "bgpPeers": [
      {
        "bgpStatus": "down",
```

```

        "customerAddress": "169.254.255.2/30",
        "addressFamily": "ipv4",
        "authKey": "0xC_ukbCerl6EYA0uexample",
        "bgpPeerState": "available",
        "amazonAddress": "169.254.255.1/30",
        "asn": 65000
    },
    {
        "bgpStatus": "down",
        "customerAddress": "2001:db8:1100:2f0:0:1:9cb4:4216/125",
        "addressFamily": "ipv6",
        "authKey": "0xS27kAIU_VHPjjAexample",
        "bgpPeerState": "deleting",
        "amazonAddress": "2001:db8:1100:2f0:0:1:9cb4:4211/125",
        "asn": 64600
    }
],
"customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fg1vuj3d\">\n  <vlan>125</
vlan>\n  <customer_address>169.254.255.2/30</customer_address>\n
  <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>0xC_ukbCerl6EYA0example</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>private</connection_type>\n</logical_connection>
\n",
    "amazonAddress": "169.254.255.1/30",
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "Test"
  }
}

```

- Untuk detail API, lihat [DeleteBgpPeer](#) di Referensi AWS CLI Perintah.

## delete-connection

Contoh kode berikut menunjukkan cara menggunakan `delete-connection`.

### AWS CLI

Untuk menghapus koneksi

`delete-connection` Perintah berikut menghapus koneksi yang ditentukan:

```
aws directconnect delete-connection --connection-id dxcon-fg31dyv6
```

**Output:**

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg31dyv6",
  "connectionState": "deleted",
  "bandwidth": "1Gbps",
  "location": "TIVIT",
  "connectionName": "Connection to AWS",
  "region": "sa-east-1"
}
```

- Untuk detail API, lihat [DeleteConnection](#) di Referensi AWS CLI Perintah.

**delete-direct-connect-gateway-association**

Contoh kode berikut menunjukkan cara menggunakan `delete-direct-connect-gateway-association`.

**AWS CLI**

Untuk menghapus asosiasi gateway Direct Connect

`delete-direct-connect-gateway-association` Contoh berikut menghapus asosiasi gateway Direct Connect dengan gateway transit yang memiliki ID asosiasi tertentu.

```
aws directconnect delete-direct-connect-gateway-association --association-id
be85116d-46eb-4b43-a27a-da0c2ad648de
```

**Output:**

```
{
  "directConnectGatewayAssociation": {
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "123456789012",
    "associationState": "disassociating",
    "associatedGateway": {
      "id": "tgw-095b3b0b54EXAMPLE",
      "type": "transitGateway",
      "ownerAccount": "123456789012",
      "region": "us-east-1"
    }
  },
}
```

```
    "associationId": " be85116d-46eb-4b43-a27a-da0c2ad648deEXAMPLE ",
    "allowedPrefixesToDirectConnectGateway": [
      {
        "cidr": "192.0.1.0/28"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Mengaitkan dan Memutuskan Gateway Transit di](#) Panduan Pengguna Direct AWS Connect.

- Untuk detail API, lihat [DeleteDirectConnectGatewayAssociation](#) di Referensi AWS CLI Perintah.

## **delete-direct-connect-gateway**

Contoh kode berikut menunjukkan cara menggunakan `delete-direct-connect-gateway`.

### AWS CLI

Untuk menghapus gateway Direct Connect

Contoh berikut menghapus gateway `5f294f92-bafb-4011-916d-9b0bexample` Direct Connect.

Perintah:

```
aws directconnect delete-direct-connect-gateway --direct-connect-gateway-id
5f294f92-bafb-4011-916d-9b0bexample
```

Output:

```
{
  "directConnectGateway": {
    "amazonSideAsn": 64512,
    "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
    "ownerAccount": "123456789012",
    "directConnectGatewayName": "DxGateway1",
    "directConnectGatewayState": "deleting"
  }
}
```

- Untuk detail API, lihat [DeleteDirectConnectGateway](#) di Referensi AWS CLI Perintah.

## delete-interconnect

Contoh kode berikut menunjukkan cara menggunakan `delete-interconnect`.

### AWS CLI

Untuk menghapus interkoneksi

`delete-interconnect` Perintah berikut menghapus interkoneksi yang ditentukan:

```
aws directconnect delete-interconnect --interconnect-id dxcon-fgktov66
```

Output:

```
{
  "interconnectState": "deleted"
}
```

- Untuk detail API, lihat [DeleteInterconnect](#) di Referensi AWS CLI Perintah.

## delete-lag

Contoh kode berikut menunjukkan cara menggunakan `delete-lag`.

### AWS CLI

Untuk menghapus LAG

Contoh berikut menghapus LAG yang ditentukan.

Perintah:

```
aws directconnect delete-lag --lag-id dxlag-ffrhowd9
```

Output:

```
{
  "awsDevice": "EqDC2-4h6ce2r1bes6",
  "numberOfConnections": 0,
  "lagState": "deleted",
  "ownerAccount": "123456789012",
  "lagName": "TestLAG",
  "connections": [],
}
```

```
"lagId": "dxlag-ffrhowd9",
"minimumLinks": 0,
"connectionsBandwidth": "1Gbps",
"region": "us-east-1",
"location": "EqDC2"
}
```

- Untuk detail API, lihat [DeleteLag](#) di Referensi AWS CLI Perintah.

## delete-virtual-interface

Contoh kode berikut menunjukkan cara menggunakan `delete-virtual-interface`.

### AWS CLI

Untuk menghapus antarmuka virtual

`delete-virtual-interface` Perintah berikut menghapus antarmuka virtual yang ditentukan:

```
aws directconnect delete-virtual-interface --virtual-interface-id dxvif-ffhkh74f
```

Output:

```
{
  "virtualInterfaceState": "deleting"
}
```

- Untuk detail API, lihat [DeleteVirtualInterface](#) di Referensi AWS CLI Perintah.

## describe-connection-loa

Contoh kode berikut menunjukkan cara menggunakan `describe-connection-loa`.

### AWS CLI

Untuk menggambarkan LOA-CFA Anda untuk koneksi menggunakan Linux atau Mac OS X

Contoh berikut menjelaskan LOA-CFA Anda untuk koneksi. `dxcon-fh6ayh1d` Isi LOA-CFA dikodekan base64. Perintah ini menggunakan `--query` parameter `--output` dan untuk mengontrol output dan mengekstrak isi `loaContent` struktur. Bagian akhir dari perintah menerjemahkan konten menggunakan base64 utilitas, dan mengirimkan output ke file PDF.

```
aws directconnect describe-connection-loa --connection-id dxcon-fh6ayh1d --output text --query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Untuk menggambarkan LOA-CFA Anda untuk koneksi menggunakan Windows

Contoh sebelumnya membutuhkan penggunaan base64 utilitas untuk memecahkan kode output. Di komputer Windows, Anda dapat menggunakannya `certutil` sebagai gantinya. Dalam contoh berikut, perintah pertama menjelaskan LOA-CFA Anda untuk koneksi `dxcon-fh6ayh1d` dan menggunakan `--query` parameter `--output` dan untuk mengontrol output dan mengekstrak isi `loaContent` struktur ke file yang disebut `myLoaCfa.base64`. Perintah kedua menggunakan utilitas `certutil` untuk memecahkan kode file dan mengirim output ke file PDF.

```
aws directconnect describe-connection-loa --connection-id dxcon-fh6ayh1d --output text --query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Untuk informasi selengkapnya tentang mengontrol output AWS CLI, lihat [Mengontrol Output Perintah dari Antarmuka Baris AWS Perintah di Panduan Pengguna Antarmuka Baris AWS Perintah](#).

- Untuk detail API, lihat [DescribeConnectionLoad](#) di Referensi AWS CLI Perintah.

## **describe-connections-on-interconnect**

Contoh kode berikut menunjukkan cara menggunakan `describe-connections-on-interconnect`.

### AWS CLI

Untuk membuat daftar koneksi pada interkoneksi

`describe-connections-on-interconnect` Perintah berikut mencantumkan koneksi yang telah disediakan pada interkoneksi yang diberikan:

```
aws directconnect describe-connections-on-interconnect --interconnect-id dxcon-fgktov66
```

Output:



```
{
  "connections": [
    {
      "partnerName": "TIVIT",
      "vlan": 101,
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffzc51m1",
      "connectionState": "ordering",
      "bandwidth": "500Mbps",
      "location": "TIVIT",
      "connectionName": "mydcinterconnect",
      "region": "sa-east-1"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeConnectionsOnInterconnect](#) di Referensi AWS CLI Perintah.

## describe-connections

Contoh kode berikut menunjukkan cara menggunakan `describe-connections`.

### AWS CLI

Untuk membuat daftar semua koneksi di wilayah saat ini

`describe-connections` Perintah berikut mencantumkan semua koneksi di wilayah saat ini:

```
aws directconnect describe-connections
```

Output:

```
{
  "connections": [
    {
      "awsDevice": "EqDC2-123h49s71dabc",
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-fguhmq1c",
      "lagId": "dxlag-ffrz71kw",
      "connectionState": "down",
      "bandwidth": "1Gbps",
```

```

        "location": "EqDC2",
        "connectionName": "My_Connection",
        "loaIssueTime": 1491568964.0,
        "region": "us-east-1"
    }
]
}

```

- Untuk detail API, lihat [DescribeConnections](#) di Referensi AWS CLI Perintah.

## describe-direct-connect-gateway-association-proposals

Contoh kode berikut menunjukkan cara menggunakan `describe-direct-connect-gateway-association-proposals`.

### AWS CLI

Untuk menjelaskan proposal asosiasi gateway Direct Connect Anda

`describe-direct-connect-gateway-association-proposals` Contoh berikut menampilkan detail tentang proposal asosiasi gateway Direct Connect Anda.

```
aws directconnect describe-direct-connect-gateway-association-proposals
```

### Output:

```

{
  "directConnectGatewayAssociationProposals": [
    {
      "proposalId": "c2ede9b4-bbc6-4d33-923c-bc4feEXAMPLE",
      "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
      "directConnectGatewayOwnerAccount": "111122223333",
      "proposalState": "requested",
      "associatedGateway": {
        "id": "tgw-02f776b1a7EXAMPLE",
        "type": "transitGateway",
        "ownerAccount": "111122223333",
        "region": "us-east-1"
      },
      "existingAllowedPrefixesToDirectConnectGateway": [
        {
          "cidr": "192.168.2.0/30"
        }
      ]
    }
  ]
}

```

```

        },
        {
            "cidr": "192.168.1.0/30"
        }
    ],
    "requestedAllowedPrefixesToDirectConnectGateway": [
        {
            "cidr": "192.168.1.0/30"
        }
    ]
},
{
    "proposalId": "cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE",
    "directConnectGatewayId": "11560968-4ac1-4fd3-bcb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "111122223333",
    "proposalState": "accepted",
    "associatedGateway": {
        "id": "tgw-045776b1a7EXAMPLE",
        "type": "transitGateway",
        "ownerAccount": "111122223333",
        "region": "us-east-1"
    },
    "existingAllowedPrefixesToDirectConnectGateway": [
        {
            "cidr": "192.168.4.0/30"
        },
        {
            "cidr": "192.168.5.0/30"
        }
    ],
    "requestedAllowedPrefixesToDirectConnectGateway": [
        {
            "cidr": "192.168.5.0/30"
        }
    ]
}
]
}
}

```

Untuk informasi selengkapnya, lihat [Mengaitkan dan Memutuskan Gateway Transit di](#) Panduan Pengguna Direct AWS Connect.

- Untuk detail API, lihat [DescribeDirectConnectGatewayAssociationProposals](#) di Referensi AWS CLI Perintah.



## describe-direct-connect-gateway-attachments

Contoh kode berikut menunjukkan cara menggunakan `describe-direct-connect-gateway-attachments`.

### AWS CLI

Untuk menjelaskan lampiran gateway Direct Connect

Contoh berikut menjelaskan antarmuka virtual yang dilampirkan ke gateway `5f294f92-bafb-4011-916d-9b0bexample` Direct Connect.

Perintah:

```
aws directconnect describe-direct-connect-gateway-attachments --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample
```

Output:

```
{
  "directConnectGatewayAttachments": [
    {
      "virtualInterfaceOwnerAccount": "123456789012",
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
      "virtualInterfaceRegion": "us-east-2",
      "attachmentState": "attaching",
      "virtualInterfaceId": "dxvif-fg9zyabc"
    }
  ],
  "nextToken":
  "eyJ2IjoxLCJzIjoxLCJpIjoibEhXd1NpUXF5RzhoL1JyUW52S1V2QT09IiwieyI6Im5wQjFHQ0RyQUdRS3puNnNXcU"
}
```

- Untuk detail API, lihat [DescribeDirectConnectGatewayAttachments](#) di Referensi AWS CLI Perintah.

## describe-direct-connect-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-direct-connect-gateways`.

## AWS CLI

Untuk mendeskripsikan gateway Direct Connect Anda

Contoh berikut menjelaskan semua gateway Direct Connect Anda.

Perintah:

```
aws directconnect describe-direct-connect-gateways
```

Output:

```
{
  "directConnectGateways": [
    {
      "amazonSideAsn": 64512,
      "directConnectGatewayId": "cf68415c-f4ae-48f2-87a7-3b52cexample",
      "ownerAccount": "123456789012",
      "directConnectGatewayName": "DxGateway2",
      "directConnectGatewayState": "available"
    },
    {
      "amazonSideAsn": 64512,
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bdexample",
      "ownerAccount": "123456789012",
      "directConnectGatewayName": "DxGateway1",
      "directConnectGatewayState": "available"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeDirectConnectGateways](#) di Referensi AWS CLI Perintah.

## describe-hosted-connections

Contoh kode berikut menunjukkan cara menggunakan `describe-hosted-connections`.

## AWS CLI

Untuk membuat daftar koneksi pada interkoneksi

Contoh berikut mencantumkan koneksi yang telah disediakan pada interkoneksi yang diberikan.

**Perintah:**

```
aws directconnect describe-hosted-connections --connection-id dxcon-fgktov66
```

**Output:**

```
{
  "connections": [
    {
      "partnerName": "TIVIT",
      "vlan": 101,
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffzc51m1",
      "connectionState": "ordering",
      "bandwidth": "500Mbps",
      "location": "TIVIT",
      "connectionName": "mydcinterconnect",
      "region": "sa-east-1"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeHostedConnections](#) di Referensi AWS CLI Perintah.

**describe-interconnect-loa**

Contoh kode berikut menunjukkan cara menggunakan `describe-interconnect-loa`.

**AWS CLI**

Untuk menggambarkan LOA-CFA Anda untuk interkoneksi menggunakan Linux atau Mac OS X

Contoh berikut menjelaskan LOA-CFA Anda untuk interkoneksi. `dxcon-fh6ayh1d` Isi LOA-CFA dikodekan base64. Perintah ini menggunakan `--query` parameter `--output` dan untuk mengontrol output dan mengekstrak isi `loaContent` struktur. Bagian akhir dari perintah menerjemahkan konten menggunakan base64 utilitas, dan mengirimkan output ke file PDF.

```
aws directconnect describe-interconnect-loa --interconnect-id dxcon-fh6ayh1d --output text --query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Untuk menggambarkan LOA-CFA Anda untuk interkoneksi menggunakan Windows

Contoh sebelumnya membutuhkan penggunaan base64 utilitas untuk memecahkan kode output. Di komputer Windows, Anda dapat menggunakannya `certutil` sebagai gantinya. Dalam contoh berikut, perintah pertama menjelaskan LOA-CFA Anda untuk interkoneksi `dxcon-fh6ayh1d` dan menggunakan `--query` parameter `--output` dan untuk mengontrol output dan mengekstrak isi `loaContent` struktur ke file yang disebut `myLoaCfa.base64`. Perintah kedua menggunakan utilitas `certutil` untuk memecahkan kode file dan mengirim output ke file PDF.

```
aws directconnect describe-interconnect-loa --interconnect-id dxcon-fh6ayh1d --output text --query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Untuk informasi selengkapnya tentang mengontrol output AWS CLI, lihat [Mengontrol Output Perintah dari Antarmuka Baris AWS Perintah di Panduan Pengguna Antarmuka Baris AWS Perintah](#).

- Untuk detail API, lihat [DescribeInterconnectLoad](#) di Referensi AWS CLI Perintah.

## describe-interconnects

Contoh kode berikut menunjukkan cara menggunakan `describe-interconnects`.

### AWS CLI

Untuk daftar interkoneksi

`describe-interconnects` Perintah berikut mencantumkan interkoneksi yang dimiliki oleh AWS akun Anda:

```
aws directconnect describe-interconnects
```

Output:

```
{
  "interconnects": [
    {
      "region": "sa-east-1",
      "bandwidth": "1Gbps",
      "location": "TIVIT",
      "interconnectName": "1G Interconnect to AWS",
    }
  ]
}
```



```
        "interconnectId": "dxcon-fgktov66",
        "interconnectState": "down"
      }
    ]
  }
```

- Untuk detail API, lihat [DescribeInterconnects](#) di Referensi AWS CLI Perintah.

## describe-lags

Contoh kode berikut menunjukkan cara menggunakan `describe-lags`.

### AWS CLI

Untuk menggambarkan LAG Anda

Perintah berikut menjelaskan semua LAG Anda untuk wilayah saat ini.

Perintah:

```
aws directconnect describe-lags
```

Output:

```
{
  "lags": [
    {
      "awsDevice": "EqDC2-19y7z3m17xpuz",
      "numberOfConnections": 2,
      "lagState": "down",
      "ownerAccount": "123456789012",
      "lagName": "DA-LAG",
      "connections": [
        {
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-ffnikghc",
          "lagId": "dxlag-fgsu9erb",
          "connectionState": "requested",
          "bandwidth": "10Gbps",
          "location": "EqDC2",
          "connectionName": "Requested Connection 1 for Lag dxlag-fgsu9erb",
          "region": "us-east-1"
        }
      ],
    }
  ],
}
```

```

        {
            "ownerAccount": "123456789012",
            "connectionId": "dxcon-fglgbdea",
            "lagId": "dxlag-fgsu9erb",
            "connectionState": "requested",
            "bandwidth": "10Gbps",
            "location": "EqDC2",
            "connectionName": "Requested Connection 2 for Lag dxlag-fgsu9erb",
            "region": "us-east-1"
        }
    ],
    "lagId": "dxlag-fgsu9erb",
    "minimumLinks": 0,
    "connectionsBandwidth": "10Gbps",
    "region": "us-east-1",
    "location": "EqDC2"
}
]
}

```

- Untuk detail API, lihat [DescribeLags](#) di Referensi AWS CLI Perintah.

## describe-loa

Contoh kode berikut menunjukkan cara menggunakan `describe-loa`.

### AWS CLI

Untuk menggambarkan LOA-CFA Anda untuk koneksi menggunakan Linux atau Mac OS X

Contoh berikut menjelaskan LOA-CFA Anda untuk koneksi. `dxcon-fh6ayh1d` isi LOA-CFA dikodekan base64. Perintah ini menggunakan `--query` parameter `--output` dan untuk mengontrol output dan mengekstrak isi `loaContent` struktur. Bagian akhir dari perintah menerjemahkan konten menggunakan base64 utilitas, dan mengirimkan output ke file PDF.

```
aws directconnect describe-loa --connection-id dxcon-fh6ayh1d --output text --query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Untuk menggambarkan LOA-CFA Anda untuk koneksi menggunakan Windows

Contoh sebelumnya membutuhkan penggunaan base64 utilitas untuk memecahkan kode output. Di komputer Windows, Anda dapat menggunakannya `certutil` sebagai gantinya. Dalam contoh

berikut, perintah pertama menjelaskan LOA-CFA Anda untuk koneksi dxcon-fh6ayh1d dan menggunakan `--query` parameter `--output` dan untuk mengontrol output dan mengekstrak isi `loaContent` struktur ke file yang disebut `myLoaCfa.base64`. Perintah kedua menggunakan utilitas `certutil` untuk memecahkan kode file dan mengirim output ke file PDF.

```
aws directconnect describe-loa --connection-id dxcon-fh6ayh1d --output text --query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Untuk informasi selengkapnya tentang mengontrol output AWS CLI, lihat [Mengontrol Output Perintah dari Antarmuka Baris AWS Perintah di Panduan Pengguna Antarmuka Baris AWS Perintah](#).

- Untuk detail API, lihat [DescribeLoad](#) di Referensi AWS CLI Perintah.

## describe-locations

Contoh kode berikut menunjukkan cara menggunakan `describe-locations`.

### AWS CLI

Untuk mencantumkan mitra dan lokasi AWS Direct Connect

`describe-locations` Perintah berikut mencantumkan mitra dan lokasi AWS Direct Connect di wilayah saat ini:

```
aws directconnect describe-locations
```

Output:

```
{
  "locations": [
    {
      "locationName": "NAP do Brasil, Barueri, Sao Paulo",
      "locationCode": "TNDB"
    },
    {
      "locationName": "Tivit - Site Transamerica (Sao Paulo)",
      "locationCode": "TIVIT"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [DescribeLocations](#) di Referensi AWS CLI Perintah.

## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Untuk menjelaskan tag untuk sumber daya AWS Direct Connect

Perintah berikut menjelaskan tag untuk koneksi `dxcon-abcabc12`.

Perintah:

```
aws directconnect describe-tags --resource-arns arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12
```

Output:

```
{
  "resourceTags": [
    {
      "resourceArn": "arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12",
      "tags": [
        {
          "value": "VAConnection",
          "key": "Name"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## describe-virtual-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-virtual-gateways`.

## AWS CLI

Untuk membuat daftar gateway pribadi virtual

`describe-virtual-gateways` Perintah berikut mencantumkan gateway pribadi virtual yang dimiliki oleh akun Anda AWS :

```
aws directconnect describe-virtual-gateways
```

Output:

```
{
  "virtualGateways": [
    {
      "virtualGatewayId": "vgw-aba37db6",
      "virtualGatewayState": "available"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeVirtualGateways](#) di Referensi AWS CLI Perintah.

## `describe-virtual-interfaces`

Contoh kode berikut menunjukkan cara menggunakan `describe-virtual-interfaces`.

## AWS CLI

Untuk membuat daftar semua antarmuka virtual

`describe-virtual-interfaces` Perintah berikut mencantumkan informasi tentang semua antarmuka virtual yang terkait dengan AWS akun Anda:

```
aws directconnect describe-virtual-interfaces --connection-id dxcon-ffjrjx17
```

Output:

```
{
  "virtualInterfaces": [
    {
      "virtualInterfaceState": "down",
      "asn": 65000,

```

```

    "vlan": 101,
    "customerAddress": "192.168.1.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffjrnx17",
    "virtualGatewayId": "vgw-aba37db6",
    "virtualInterfaceId": "dxvif-ffhkh74f",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [],
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\\\"UTF-8\\\"?>\\n<logical_connection id=\\\"dxvif-ffhkh74f\\\">\\n  <vlan>101</
vlan>\\n  <customer_address>192.168.1.2/30</customer_address>\\n
  <amazon_address>192.168.1.1/30</amazon_address>\\n  <bgp_asn>65000</bgp_asn>\\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\\n  <connection_type>private</connection_type>\\n</logical_connection>\\n",
    "amazonAddress": "192.168.1.1/30",
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "PrivateVirtualInterface"
  },
  {
    "virtualInterfaceState": "verifying",
    "asn": 65000,
    "vlan": 2000,
    "customerAddress": "203.0.113.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffjrnx17",
    "virtualGatewayId": "",
    "virtualInterfaceId": "dxvif-fgh0hcrk",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [
      {
        "cidr": "203.0.113.4/30"
      },
      {
        "cidr": "203.0.113.0/30"
      }
    ],
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\\\"UTF-8\\\"?>\\n<logical_connection id=\\\"dxvif-fgh0hcrk\\\">\\n  <vlan>2000</
vlan>\\n  <customer_address>203.0.113.2/30</customer_address>\\n
  <amazon_address>203.0.113.1/30</amazon_address>\\n  <bgp_asn>65000</bgp_asn>\\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\\n  <connection_type>public</connection_type>\\n</logical_connection>\\n",

```

```
        "amazonAddress": "203.0.113.1/30",
        "virtualInterfaceType": "public",
        "virtualInterfaceName": "PublicVirtualInterface"
    }
]
}
```

- Untuk detail API, lihat [DescribeVirtualInterfaces](#) di Referensi AWS CLI Perintah.

## disassociate-connection-from-lag

Contoh kode berikut menunjukkan cara menggunakan `disassociate-connection-from-lag`.

### AWS CLI

Untuk memisahkan koneksi dari LAG

Contoh berikut memisahkan koneksi yang ditentukan dari LAG yang ditentukan.

Perintah:

```
aws directconnect disassociate-connection-from-lag --lag-id dxlag-fhccu14t --
connection-id dxcon-fg9607vm
```

Output:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "EqDC2",
  "connectionName": "Con2ForLag",
  "region": "us-east-1"
}
```

- Untuk detail API, lihat [DisassociateConnectionFromLag](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menambahkan tag ke sumber daya AWS Direct Connect

Perintah berikut menambahkan tag dengan kunci dari Name dan nilai VAConnection untuk koneksi dxcon-abcabc12. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws directconnect tag-resource --resource-arn arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12 --tags "key=Name,value=VAConnection"
```

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan untag-resource.

## AWS CLI

Untuk menghapus tag dari sumber daya AWS Direct Connect

Perintah berikut menghapus tag dengan kunci Name dari koneksi dxcon-abcabc12. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws directconnect untag-resource --resource-arn arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12 --tag-keys Name
```

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-direct-connect-gateway-association

Contoh kode berikut menunjukkan cara menggunakan update-direct-connect-gateway-association.

## AWS CLI

Untuk memperbarui atribut yang ditentukan dari asosiasi gateway Direct Connect



`update-direct-connect-gateway-association` Contoh berikut menambahkan blok CIDR yang ditentukan ke asosiasi gateway Direct Connect.

```
aws directconnect update-direct-connect-gateway-association \
  --association-id 820a6e4f-5374-4004-8317-3f64bEXAMPLE \
  --add-allowed-prefixes-to-direct-connect-gateway cidr=192.168.2.0/30
```

Output:

```
{
  "directConnectGatewayAssociation": {
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "111122223333",
    "associationState": "updating",
    "associatedGateway": {
      "id": "tgw-02f776b1a7EXAMPLE",
      "type": "transitGateway",
      "ownerAccount": "111122223333",
      "region": "us-east-1"
    },
    "associationId": "820a6e4f-5374-4004-8317-3f64bEXAMPLE",
    "allowedPrefixesToDirectConnectGateway": [
      {
        "cidr": "192.168.2.0/30"
      },
      {
        "cidr": "192.168.1.0/30"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Direct Connect Gateway](#) di Panduan Pengguna Direct AWS Connect.

- Untuk detail API, lihat [UpdateDirectConnectGatewayAssociation](#) di Referensi AWS CLI Perintah.

## update-lag

Contoh kode berikut menunjukkan cara menggunakan `update-lag`.

## AWS CLI

Untuk memperbarui LAG

Contoh berikut mengubah nama LAG yang ditentukan.

Perintah:

```
aws directconnect update-lag --lag-id dxlag-ffjhj91x --lag-name 2ConnLag
```

Output:

```
{
  "awsDevice": "CSVA1-23u8t1paz8iks",
  "numberOfConnections": 2,
  "lagState": "down",
  "ownerAccount": "123456789012",
  "lagName": "2ConnLag",
  "connections": [
    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-fflqyj95",
      "lagId": "dxlag-ffjhj91x",
      "connectionState": "requested",
      "bandwidth": "1Gbps",
      "location": "CSVA1",
      "connectionName": "Requested Connection 2 for Lag dxlag-ffjhj91x",
      "region": "us-east-1"
    },
    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffqr6x5q",
      "lagId": "dxlag-ffjhj91x",
      "connectionState": "requested",
      "bandwidth": "1Gbps",
      "location": "CSVA1",
      "connectionName": "Requested Connection 1 for Lag dxlag-ffjhj91x",
      "region": "us-east-1"
    }
  ],
  "lagId": "dxlag-ffjhj91x",
  "minimumLinks": 0,
  "connectionsBandwidth": "1Gbps",
  "region": "us-east-1",
```

```
"location": "CSVA1"
}
```

- Untuk detail API, lihat [UpdateLag](#) di Referensi AWS CLI Perintah.

## update-virtual-interface-attributes

Contoh kode berikut menunjukkan cara menggunakan `update-virtual-interface-attributes`.

### AWS CLI

Untuk memperbarui MTU antarmuka virtual

`update-virtual-interface-attributes` Contoh berikut memperbarui MTU dari antarmuka virtual yang ditentukan.

```
aws directconnect update-virtual-interface-attributes \
  --virtual-interface-id dxvif-fEXAMPLE \
  --mtu 1500
```

Output:

```
{
  "ownerAccount": "1111222233333",
  "virtualInterfaceId": "dxvif-fEXAMPLE",
  "location": "loc1",
  "connectionId": "dxlag-fEXAMPLE",
  "virtualInterfaceType": "transit",
  "virtualInterfaceName": "example transit virtual interface",
  "vlan": 125,
  "asn": 650001,
  "amazonSideAsn": 64512,
  "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
  "amazonAddress": "169.254.248.1/30",
  "customerAddress": "169.254.248.2/30",
  "addressFamily": "ipv4",
  "virtualInterfaceState": "down",
  "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fEXAMPLE\">\n  <vlan>125</vlan>
\n  <customer_address>169.254.248.2/30</customer_address>\n
  <amazon_address>169.254.248.1/30</amazon_address>\n  <bgp_asn>650001</bgp_asn>\n
```

```

    <bgp_auth_key>0xzxcgA9YoW9h58u8SEXAMPLE</bgp_auth_key>\n  <amazon_bgp_asn>64512</amazon_bgp_asn>\n  <connection_type>transit</connection_type>\n</logical_connection>\n",
    "mtu": 1500,
    "jumboFrameCapable": true,
    "virtualGatewayId": "",
    "directConnectGatewayId": "879b76a1-403d-4700-8b53-4a56ed85436e",
    "routeFilterPrefixes": [],
    "bgpPeers": [
      {
        "bgpPeerId": "dxpeer-fEXAMPLE",
        "asn": 650001,
        "authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
        "addressFamily": "ipv4",
        "amazonAddress": "169.254.248.1/30",
        "customerAddress": "169.254.248.2/30",
        "bgpPeerState": "available",
        "bgpStatus": "down",
        "awsDeviceV2": "loc1-26wz6vEXAMPLE"
      }
    ],
    "region": "sa-east-1",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE",
    "tags": []
  }
}

```

Untuk informasi selengkapnya, lihat [Mengatur MTU Jaringan untuk Antarmuka Virtual Pribadi atau Antarmuka Virtual Transit di Panduan Pengguna Direct AWS Connect](#).

- Untuk detail API, lihat [UpdateVirtualInterfaceAttributes](#) di Referensi AWS CLI Perintah.

## AWS Directory Service contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Directory Service.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **describe-directories**

Contoh kode berikut menunjukkan cara menggunakan `describe-directories`.

AWS CLI

Untuk mendapatkan detail tentang direktori Anda

`describe-directories` Contoh berikut menampilkan rincian tentang direktori yang ditentukan.

```
aws ds describe-directories \
  --directory-id d-a1b2c3d4e5
```

Output:

```
{
  "DirectoryDescriptions": [
    {
      "DirectoryId": "d-a1b2c3d4e5",
      "Name": "mydirectory.example.com",
      "ShortName": "mydirectory",
      "Size": "Small",
      "Edition": "Standard",
      "Alias": "d-a1b2c3d4e5",
      "AccessUrl": "d-a1b2c3d4e5.awsapps.com",
      "Stage": "Active",
      "ShareStatus": "Shared",
      "ShareMethod": "HANDSHAKE",
      "ShareNotes": "These are my share notes",
      "LaunchTime": "2019-07-08T15:33:46.327000-07:00",
      "StageLastUpdatedDateTime": "2019-07-08T15:59:12.307000-07:00",
      "Type": "SharedMicrosoftAD",
      "SsoEnabled": false,
      "DesiredNumberOfDomainControllers": 0,
```

```

    "OwnerDirectoryDescription": {
      "DirectoryId": "d-b2c3d4e5f6",
      "AccountId": "123456789111",
      "DnsIpAddrs": [
        "203.113.0.248",
        "203.113.0.253"
      ],
      "VpcSettings": {
        "VpcId": "vpc-a1b2c3d4",
        "SubnetIds": [
          "subnet-a1b2c3d4",
          "subnet-d4c3b2a1"
        ],
        "AvailabilityZones": [
          "us-west-2a",
          "us-west-2c"
        ]
      }
    }
  ]
}

```

- Untuk detail API, lihat [DescribeDirectories](#) di Referensi AWS CLI Perintah.

## describe-trusts

Contoh kode berikut menunjukkan cara menggunakan `describe-trusts`.

### AWS CLI

Untuk mendapatkan detail tentang hubungan kepercayaan Anda

`describe-trusts` Contoh berikut menampilkan rincian tentang hubungan kepercayaan untuk direktori tertentu.

```
aws ds describe-trusts \
  --directory-id d-a1b2c3d4e5
```

Output:

```
{
```

```
"Trusts": [
  {
    "DirectoryId": "d-a1b2c3d4e5",
    "TrustId": "t-9a8b7c6d5e",
    "RemoteDomainName": "other.example.com",
    "TrustType": "Forest",
    "TrustDirection": "Two-Way",
    "TrustState": "Verified",
    "CreatedDateTime": "2017-06-20T18:08:45.614000-07:00",
    "LastUpdatedDateTime": "2019-06-04T10:52:12.410000-07:00",
    "StateLastUpdatedDateTime": "2019-06-04T10:52:12.410000-07:00",
    "SelectiveAuth": "Disabled"
  }
]
```

- Untuk detail API, lihat [DescribeTrusts](#) di Referensi AWS CLI Perintah.

## AWS DMS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS DMS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags-to-resource**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-resource`.

## AWS CLI

Untuk menambahkan tag ke sumber daya

`add-tags-to-resource` Contoh berikut menambahkan tag ke contoh replikasi.

```
aws dms add-tags-to-resource \  
  --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \  
  \  
  --tags Key=Environment,Value=PROD Key=Project,Value=dbMigration
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [AddTagsToResource](#) di Referensi AWS CLI Perintah.

## `create-endpoint`

Contoh kode berikut menunjukkan cara menggunakan `create-endpoint`.

## AWS CLI

Untuk membuat titik akhir

`create-endpoint` Contoh berikut membuat titik akhir untuk sumber Amazon S3.

```
aws dms create-endpoint \  
  --endpoint-type source \  
  --engine-name s3 \  
  --endpoint-identifier src-endpoint \  
  --s3-settings file://s3-settings.json
```

Isi dari `s3-settings.json`:

```
{  
  "BucketName": "my-corp-data",  
  "BucketFolder": "sourcedata",  
  "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role"  
}
```

Output:



```
{
  "Endpoint": {
    "EndpointIdentifier": "src-endpoint",
    "EndpointType": "SOURCE",
    "EngineName": "s3",
    "EngineDisplayName": "Amazon S3",
    "ExtraConnectionAttributes": "bucketFolder=sourcedata;bucketName=my-corp-
data;compressionType=NONE;csvDelimiter=,;csvRowDelimiter=\n;",
    "Status": "active",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECU0J6QVZ56DAHT3U",
    "SslMode": "none",
    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
    "S3Settings": {
      "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
      "CsvRowDelimiter": "\n",
      "CsvDelimiter": ",",
      "BucketFolder": "sourcedata",
      "BucketName": "my-corp-data",
      "CompressionType": "NONE",
      "EnableStatistics": true
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Titik Akhir AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [CreateEndpoint](#) di Referensi AWS CLI Perintah.

## create-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `create-event-subscription`.

### AWS CLI

Untuk membuat daftar langganan acara

`create-event-subscription` Contoh berikut membuat langganan acara ke topik Amazon SNS (`my-sns-topic`).

```
aws dms create-event-subscription \
```

```
--subscription-name my-dms-events \  
--sns-topic-arn arn:aws:sns:us-east-1:123456789012:my-sns-topic
```

Output:

```
{  
  "EventSubscription": {  
    "CustomerAwsId": "123456789012",  
    "CustSubscriptionId": "my-dms-events",  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",  
    "Status": "creating",  
    "SubscriptionCreationTime": "2020-05-21 21:58:38.598",  
    "Enabled": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Acara dan Pemberitahuan](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [CreateEventSubscription](#) di Referensi AWS CLI Perintah.

## create-replication-instance

Contoh kode berikut menunjukkan cara menggunakan `create-replication-instance`.

AWS CLI

Untuk membuat contoh replikasi

`create-replication-instance` Contoh berikut menciptakan contoh replikasi.

```
aws dms create-replication-instance \  
  --replication-instance-identifier my-repl-instance \  
  --replication-instance-class dms.t2.micro \  
  --allocated-storage 5
```

Output:

```
{  
  "ReplicationInstance": {  
    "ReplicationInstanceIdentifier": "my-repl-instance",  
    "ReplicationInstanceClass": "dms.t2.micro",
```

```
"ReplicationInstanceStatus": "creating",
"AllocatedStorage": 5,
"VpcSecurityGroups": [
  {
    "VpcSecurityGroupId": "sg-f839b688",
    "Status": "active"
  }
],
"ReplicationSubnetGroup": {
  "ReplicationSubnetGroupIdentifier": "default",
  "ReplicationSubnetGroupDescription": "default",
  "VpcId": "vpc-136a4c6a",
  "SubnetGroupStatus": "Complete",
  "Subnets": [
    {
      "SubnetIdentifier": "subnet-da327bf6",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1a"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-42599426",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1d"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-bac383e0",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1c"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-6746046b",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1f"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-d7c825e8",
```

```

        "SubnetAvailabilityZone": {
            "Name": "us-east-1e"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-cbfff283",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1b"
        },
        "SubnetStatus": "Active"
    }
]
},
"PreferredMaintenanceWindow": "sat:12:35-sat:13:05",
"PendingModifiedValues": {},
"MultiAZ": false,
"EngineVersion": "3.3.2",
"AutoMinorVersionUpgrade": true,
"KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",
"ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:ZK2VQBUWFDBAWHIXHAYG5G2PKY",
"PubliclyAccessible": true
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instance Replikasi AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [CreateReplicationInstance](#) di Referensi AWS CLI Perintah.

## create-replication-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `create-replication-subnet-group`.

### AWS CLI

Untuk membuat grup subnet

`create-replication-subnet-group` Contoh berikut membuat grup yang terdiri dari 3 subnet.

```
aws dms create-replication-subnet-group \
```

```
--replication-subnet-group-identifier my-subnet-group \  
--replication-subnet-group-description "my subnet group" \  
--subnet-ids subnet-da327bf6 subnet-bac383e0 subnet-d7c825e8
```

### Output:

```
{  
  "ReplicationSubnetGroup": {  
    "ReplicationSubnetGroupIdentifier": "my-subnet-group",  
    "ReplicationSubnetGroupDescription": "my subnet group",  
    "VpcId": "vpc-136a4c6a",  
    "SubnetGroupStatus": "Complete",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-da327bf6",  
        "SubnetAvailabilityZone": {  
          "Name": "us-east-1a"  
        },  
        "SubnetStatus": "Active"  
      },  
      {  
        "SubnetIdentifier": "subnet-bac383e0",  
        "SubnetAvailabilityZone": {  
          "Name": "us-east-1c"  
        },  
        "SubnetStatus": "Active"  
      },  
      {  
        "SubnetIdentifier": "subnet-d7c825e8",  
        "SubnetAvailabilityZone": {  
          "Name": "us-east-1e"  
        },  
        "SubnetStatus": "Active"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Jaringan untuk Instance Replikasi](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [CreateReplicationSubnetGroup](#) di Referensi AWS CLI Perintah.

## create-replication-task

Contoh kode berikut menunjukkan cara menggunakan `create-replication-task`.

### AWS CLI

Untuk membuat tugas replikasi

`create-replication-task` Contoh berikut menciptakan tugas replikasi.

```
aws dms create-replication-task \  
  --replication-task-identifier movedata \  
  --source-endpoint-arn arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA \  
  --target-endpoint-arn arn:aws:dms:us-  
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U \  
  --replication-instance-arn $RI_ARN \  
  --migration-type full-load \  
  --table-mappings file://table-mappings.json
```

Isi dari `table-mappings.json`:

```
{  
  "rules": [  
    {  
      "rule-type": "selection",  
      "rule-id": "1",  
      "rule-name": "1",  
      "object-locator": {  
        "schema-name": "prodrep",  
        "table-name": "%"  
      },  
      "rule-action": "include",  
      "filters": []  
    }  
  ]  
}
```

Output:

```
{  
  "ReplicationTask": {  
    "ReplicationTaskIdentifier": "moveit2",
```

```

    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "MigrationType": "full-load",
    "TableMappings": ...output omitted... ,
    "ReplicationTaskSettings": ...output omitted... ,
    "Status": "creating",
    "ReplicationTaskCreationDate": 1590524772.505,
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tugas AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [CreateReplicationTask](#) di Referensi AWS CLI Perintah.

## delete-connection

Contoh kode berikut menunjukkan cara menggunakan `delete-connection`.

### AWS CLI

Untuk menghapus koneksi

`delete-connection` Contoh berikut memisahkan titik akhir dari contoh replikasi.

```

aws dms delete-connection \
  --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA \
  --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE

```

Output:

```

{
  "Connection": {
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",

```

```

    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",
    "Status": "deleting",
    "EndpointIdentifier": "src-database-1",
    "ReplicationInstanceIdentifier": "my-repl-instance"
  }
}

```

Untuk informasi selengkapnya, lihat [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Endpoints.Creating.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Endpoints.Creating.html) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DeleteConnection](#) di Referensi AWS CLI Perintah.

## delete-endpoint

Contoh kode berikut menunjukkan cara menggunakan `delete-endpoint`.

### AWS CLI

Untuk menghapus titik akhir

`delete-endpoint` Contoh berikut menghapus titik akhir.

```

aws dms delete-endpoint \
  --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:0UJJVX04XZ4CYTSEG5XGMN2R3Y

```

Output:

```

{
  "Endpoint": {
    "EndpointIdentifier": "src-endpoint",
    "EndpointType": "SOURCE",
    "EngineName": "s3",
    "EngineDisplayName": "Amazon S3",
    "ExtraConnectionAttributes": "bucketFolder=sourcedata;bucketName=my-corp-
data;compressionType=NONE;csvDelimiter=,;csvRowDelimiter=\n;",
    "Status": "deleting",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:0UJJVX04XZ4CYTSEG5XGMN2R3Y",
    "SslMode": "none",
    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
    "S3Settings": {

```



```

    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
    "CsvRowDelimiter": "\\n",
    "CsvDelimiter": ",",
    "BucketFolder": "sourcedata",
    "BucketName": "my-corp-data",
    "CompressionType": "NONE",
    "EnableStatistics": true
  }
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Titik Akhir AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DeleteEndpoint](#) di Referensi AWS CLI Perintah.

## delete-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `delete-event-subscription`.

### AWS CLI

Untuk menghapus langganan acara

`delete-event-subscription` Contoh berikut menghapus langganan ke topik Amazon SNS.

```

aws dms delete-event-subscription \
  --subscription-name "my-dms-events"

```

Output:

```

{
  "EventSubscription": {
    "CustomerAwsId": "123456789012",
    "CustSubscriptionId": "my-dms-events",
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
    "Status": "deleting",
    "SubscriptionCreationTime": "2020-05-21 21:58:38.598",
    "Enabled": true
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Acara dan Pemberitahuan](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DeleteEventSubscription](#) di Referensi AWS CLI Perintah.

## delete-replication-instance

Contoh kode berikut menunjukkan cara menggunakan `delete-replication-instance`.

### AWS CLI

Untuk menghapus contoh replikasi

Contoh `delete-replication-instance` berikut menghapus instans replikasi.

```
aws dms delete-replication-instance \
  --replication-instance-arn arn:aws:dms:us-
  east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE
```

### Output:

```
{
  "ReplicationInstance": {
    "ReplicationInstanceIdentifier": "my-repl-instance",
    "ReplicationInstanceClass": "dms.t2.micro",
    "ReplicationInstanceStatus": "deleting",
    "AllocatedStorage": 5,
    "InstanceCreateTime": 1590011235.952,
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-f839b688",
        "Status": "active"
      }
    ],
    "AvailabilityZone": "us-east-1e",
    "ReplicationSubnetGroup": {
      "ReplicationSubnetGroupIdentifier": "default",
      "ReplicationSubnetGroupDescription": "default",
      "VpcId": "vpc-136a4c6a",
      "SubnetGroupStatus": "Complete",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-da327bf6",
```

```
        "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-42599426",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1d"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-bac383e0",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1c"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-6746046b",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1f"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-d7c825e8",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1e"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-cbfff283",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1b"
        },
        "SubnetStatus": "Active"
    }
]
},
"PreferredMaintenanceWindow": "wed:11:42-wed:12:12",
"PendingModifiedValues": {},
```

```

    "MultiAZ": true,
    "EngineVersion": "3.3.2",
    "AutoMinorVersionUpgrade": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
    "ReplicationInstancePublicIpAddress": "54.225.120.92",
    "ReplicationInstancePrivateIpAddress": "172.31.30.121",
    "ReplicationInstancePublicIpAddresses": [
        "54.225.120.92",
        "3.230.18.248"
    ],
    "ReplicationInstancePrivateIpAddresses": [
        "172.31.30.121",
        "172.31.75.90"
    ],
    "PubliclyAccessible": true,
    "SecondaryAvailabilityZone": "us-east-1b"
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instance Replikasi AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DeleteReplicationInstance](#) di Referensi AWS CLI Perintah.

## delete-replication-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `delete-replication-subnet-group`.

### AWS CLI

Untuk menghapus grup subnet

`delete-replication-subnet-group` Contoh berikut menghapus grup subnet.

```
aws dms delete-replication-subnet-group \
--replication-subnet-group-identifier my-subnet-group
```

Output:

(none)

Untuk informasi selengkapnya, lihat [Menyiapkan Jaringan untuk Instance Replikasi](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DeleteReplicationSubnetGroup](#) di Referensi AWS CLI Perintah.

## delete-replication-task

Contoh kode berikut menunjukkan cara menggunakan `delete-replication-task`.

### AWS CLI

Untuk menghapus tugas replikasi

`delete-replication-task` Contoh berikut menghapus tugas replikasi.

```
aws dms delete-replication-task \  
  --replication-task-arn arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Output:

```
{  
  "ReplicationTask": {  
    "ReplicationTaskIdentifier": "moveit2",  
    "SourceEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",  
    "TargetEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "MigrationType": "full-load",  
    "TableMappings": "...output omitted...",  
    "ReplicationTaskSettings": "...output omitted...",  
    "Status": "deleting",  
    "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",  
    "ReplicationTaskCreationDate": 1590524772.505,  
    "ReplicationTaskStartDate": 1590789988.677,  
    "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tugas AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DeleteReplicationTask](#) di Referensi AWS CLI Perintah.

## describe-account-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-account-attributes`.

### AWS CLI

Untuk menggambarkan atribut akun

`describe-account-attributes` Contoh berikut mencantumkan atribut untuk AWS akun Anda.

```
aws dms describe-account-attributes
```

Output:

```
{  
  "AccountQuotas": [  
    {  
      "AccountQuotaName": "ReplicationInstances",  
      "Used": 1,  
      "Max": 20  
    },  
    {  
      "AccountQuotaName": "AllocatedStorage",  
      "Used": 5,  
      "Max": 10000  
    },  
    ...remaining output omitted...  
  ],  
  "UniqueAccountIdentifier": "cqahfbfy5xee"  
}
```

- Untuk detail API, lihat [DescribeAccountAttributes](#) di Referensi AWS CLI Perintah.

## describe-certificates

Contoh kode berikut menunjukkan cara menggunakan `describe-certificates`.

### AWS CLI

Untuk daftar sertifikat yang tersedia

`describe-certificates` Contoh berikut mencantumkan sertifikat yang tersedia di AWS akun Anda.

```
aws dms describe-certificates
```

Output:

```
{
  "Certificates": [
    {
      "CertificateIdentifier": "my-cert",
      "CertificateCreationDate": 1543259542.506,
      "CertificatePem": "-----BEGIN CERTIFICATE-----
\nMIID9DCCAtygAwIBAgIBQjANBgkqhkiG9w0BAQ ...U"
      ... remaining output omitted ...
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan SSL](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeCertificates](#) di Referensi AWS CLI Perintah.

## describe-connections

Contoh kode berikut menunjukkan cara menggunakan `describe-connections`.

### AWS CLI

Untuk menggambarkan koneksi

`describe-connections` Contoh berikut mencantumkan koneksi yang telah Anda uji antara instance replikasi dan titik akhir.

```
aws dms describe-connections
```

Output:

```
{
  "Connections": [
    {
      "Status": "successful",
      "ReplicationInstanceIdentifier": "test",
      "EndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:ZW5UAN6P4E77EC7YWHK4RZZ3BE",
      "EndpointIdentifier": "testsrc1",
      "ReplicationInstanceArn": "arn:aws:dms:us-east-1:123456789012:rep:6UTDJGB0US3VI3SUWA66XFJCJQ"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Titik Akhir Sumber dan Target](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeConnections](#) di Referensi AWS CLI Perintah.

## `describe-endpoint-types`

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoint-types`.

AWS CLI

Untuk mencantumkan jenis titik akhir yang tersedia

`describe-endpoint-types` Contoh berikut mencantumkan jenis endpoint MySQL yang tersedia.

```
aws dms describe-endpoint-types \
  --filters "Name=engine-name,Values=mysql"
```

Output:



```
{
  "SupportedEndpointTypes": [
    {
      "EngineName": "mysql",
      "SupportsCDC": true,
      "EndpointType": "source",
      "EngineDisplayName": "MySQL"
    },
    {
      "EngineName": "mysql",
      "SupportsCDC": true,
      "EndpointType": "target",
      "EngineDisplayName": "MySQL"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat *Bekerja dengan AWS DMS Endpoints`\_\_* di *Panduan Pengguna Layanan Migrasi Database AWS* . < [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Endpoints.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Endpoints.html)>

- Untuk detail API, lihat [DescribeEndpointTypes](#) di Referensi AWS CLI Perintah.

## describe-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoints`.

### AWS CLI

Untuk menggambarkan titik akhir

`describe-endpoints` Contoh berikut mencantumkan titik akhir di AWS akun Anda.

```
aws dms describe-endpoints
```

Output:

```
{
  "Endpoints": [
    {
      "Username": "dms",
```

```

    "Status": "active",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:SF2W0FLWYWKVE0HID2EKLP3SJI",
    "ServerName": "ec2-52-32-48-61.us-west-2.compute.amazonaws.com",
    "EndpointType": "SOURCE",
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/94d5c4e7-4e4c-44be-
b58a-c8da7adf57cd",
    "DatabaseName": "test",
    "EngineName": "mysql",
    "EndpointIdentifier": "pri100",
    "Port": 8193
  },
  {
    "Username": "admin",
    "Status": "active",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:TJJZCIH3CJ24TJRU4VC32WEWFR",
    "ServerName": "test.example.com",
    "EndpointType": "SOURCE",
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/2431021b-1cf2-
a2d4-77b2-59a9e4bce323",
    "DatabaseName": "EMPL",
    "EngineName": "oracle",
    "EndpointIdentifier": "test",
    "Port": 1521
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Titik Akhir AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeEndpoints](#) di Referensi AWS CLI Perintah.

## describe-event-categories

Contoh kode berikut menunjukkan cara menggunakan `describe-event-categories`.

### AWS CLI

Untuk menggambarkan kategori acara

`describe-event-categories` Contoh berikut mencantumkan kategori acara yang tersedia.

```
aws dms describe-event-categories
```

Output:

```
{
  "EventCategoryGroupList": [
    {
      "SourceType": "replication-instance",
      "EventCategories": [
        "low storage",
        "configuration change",
        "maintenance",
        "deletion",
        "creation",
        "failover",
        "failure"
      ]
    },
    {
      "SourceType": "replication-task",
      "EventCategories": [
        "configuration change",
        "state change",
        "deletion",
        "creation",
        "failure"
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Acara dan Pemberitahuan](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeEventCategories](#) di Referensi AWS CLI Perintah.

## **describe-event-subscriptions**

Contoh kode berikut menunjukkan cara menggunakan `describe-event-subscriptions`.

## AWS CLI

Untuk menggambarkan langganan acara

`describe-event-subscriptions` Contoh berikut mencantumkan langganan acara ke topik Amazon SNS.

```
aws dms describe-event-subscriptions
```

Output:

```
{
  "EventSubscriptionsList": [
    {
      "CustomerAwsId": "123456789012",
      "CustSubscriptionId": "my-dms-events",
      "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
      "Status": "deleting",
      "SubscriptionCreationTime": "2020-05-21 22:28:51.924",
      "Enabled": true
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Acara dan Pemberitahuan](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeEventSubscriptions](#) di Referensi AWS CLI Perintah.

## `describe-events`

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

## AWS CLI

Untuk daftar acara DMS

`describe-events` Contoh berikut mencantumkan peristiwa yang berasal dari contoh replikasi.

```
aws dms describe-events \
  --source-type "replication-instance"
```

**Output:**

```
{
  "Events": [
    {
      "SourceIdentifier": "my-repl-instance",
      "SourceType": "replication-instance",
      "Message": "Replication application shutdown",
      "EventCategories": [],
      "Date": 1590771645.776
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Acara dan Pemberitahuan](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

**describe-orderable-replication-instances**

Contoh kode berikut menunjukkan cara menggunakan `describe-orderable-replication-instances`.

**AWS CLI**

Untuk menggambarkan contoh replikasi yang dapat dipesan

`describe-orderable-replication-instances` Contoh berikut mencantumkan jenis contoh replikasi yang dapat Anda pesan.

```
aws dms describe-orderable-replication-instances
```

**Output:**

```
{
  "OrderableReplicationInstances": [
    {
      "EngineVersion": "3.3.2",
      "ReplicationInstanceClass": "dms.c4.2xlarge",
      "StorageType": "gp2",

```

```
    "MinAllocatedStorage": 5,  
    "MaxAllocatedStorage": 6144,  
    "DefaultAllocatedStorage": 100,  
    "IncludedAllocatedStorage": 100,  
    "AvailabilityZones": [  
        "us-east-1a",  
        "us-east-1b",  
        "us-east-1c",  
        "us-east-1d",  
        "us-east-1e",  
        "us-east-1f"  
    ]  
  },  
  {  
    "EngineVersion": "3.3.2",  
    "ReplicationInstanceClass": "dms.c4.4xlarge",  
    "StorageType": "gp2",  
    "MinAllocatedStorage": 5,  
    "MaxAllocatedStorage": 6144,  
    "DefaultAllocatedStorage": 100,  
    "IncludedAllocatedStorage": 100,  
    "AvailabilityZones": [  
        "us-east-1a",  
        "us-east-1b",  
        "us-east-1c",  
        "us-east-1d",  
        "us-east-1e",  
        "us-east-1f"  
    ]  
  },  
  ...remaining output omitted...  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instance Replikasi AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeOrderableReplicationInstances](#) di Referensi AWS CLI Perintah.

## **describe-refresh-schemas-status**

Contoh kode berikut menunjukkan cara menggunakan `describe-refresh-schemas-status`.

## AWS CLI

Untuk mencantumkan status penyegaran untuk titik akhir

`describe-refresh-schemas-status` Contoh berikut mengembalikan status permintaan penyegaran sebelumnya.

```
aws dms describe-refresh-schemas-status \
  --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA
```

Output:

```
{
  "RefreshSchemasStatus": {
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "Status": "successful",
    "LastRefreshDate": 1590786544.605
  }
}
```

- Untuk detail API, lihat [DescribeRefreshSchemasStatus](#) di Referensi AWS CLI Perintah.

## `describe-replication-instances`

Contoh kode berikut menunjukkan cara menggunakan `describe-replication-instances`.

### AWS CLI

Untuk menggambarkan contoh replikasi

`describe-replication-instances` Contoh berikut mencantumkan contoh replikasi di akun Anda AWS .

```
aws dms describe-replication-instances
```

Output:

```
{
```

```
"ReplicationInstances": [
  {
    "ReplicationInstanceIdentifier": "my-repl-instance",
    "ReplicationInstanceClass": "dms.t2.micro",
    "ReplicationInstanceStatus": "available",
    "AllocatedStorage": 5,
    "InstanceCreateTime": 1590011235.952,
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-f839b688",
        "Status": "active"
      }
    ],
    "AvailabilityZone": "us-east-1e",
    "ReplicationSubnetGroup": {
      "ReplicationSubnetGroupIdentifier": "default",
      "ReplicationSubnetGroupDescription": "default",
      "VpcId": "vpc-136a4c6a",
      "SubnetGroupStatus": "Complete",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-da327bf6",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-42599426",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1d"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-bac383e0",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1c"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-6746046b",
          "SubnetAvailabilityZone": {
```



```

        "Name": "us-east-1f"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-d7c825e8",
    "SubnetAvailabilityZone": {
      "Name": "us-east-1e"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-cbfff283",
    "SubnetAvailabilityZone": {
      "Name": "us-east-1b"
    },
    "SubnetStatus": "Active"
  }
]
},
"PreferredMaintenanceWindow": "wed:11:42-wed:12:12",
"PendingModifiedValues": {
  "MultiAZ": true
},
"MultiAZ": false,
"EngineVersion": "3.3.2",
"AutoMinorVersionUpgrade": true,
"KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
f7bc0f8e-1a3a-4ace-9faa-e8494fa3921a",
"ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
"ReplicationInstancePublicIpAddress": "3.230.18.248",
"ReplicationInstancePrivateIpAddress": "172.31.75.90",
"ReplicationInstancePublicIpAddresses": [
  "3.230.18.248"
],
"ReplicationInstancePrivateIpAddresses": [
  "172.31.75.90"
],
"PubliclyAccessible": true,
"FreeUntil": 1590194829.267
}
]

```

```
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instance Replikasi AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeReplicationInstances](#) di Referensi AWS CLI Perintah.

## describe-replication-subnet-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-replication-subnet-groups`.

### AWS CLI

Untuk menampilkan grup subnet yang tersedia

`describe-replication-subnet-groups` Contoh berikut mencantumkan grup subnet yang tersedia.

```
aws dms describe-replication-subnet-groups \
  --filter "Name=replication-subnet-group-id,Values=my-subnet-group"
```

Output:

```
{
  "ReplicationSubnetGroups": [
    {
      "ReplicationSubnetGroupIdentifier": "my-subnet-group",
      "ReplicationSubnetGroupDescription": "my subnet group",
      "VpcId": "vpc-136a4c6a",
      "SubnetGroupStatus": "Complete",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-da327bf6",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-bac383e0",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1c"
          }
        }
      ]
    }
  ]
}
```

```

    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-d7c825e8",
    "SubnetAvailabilityZone": {
      "Name": "us-east-1e"
    },
    "SubnetStatus": "Active"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan Jaringan untuk Instance Replikasi](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeReplicationSubnetGroups](#) di Referensi AWS CLI Perintah.

## describe-replication-task-assessment-results

Contoh kode berikut menunjukkan cara menggunakan `describe-replication-task-assessment-results`.

### AWS CLI

Untuk membuat daftar hasil penilaian tugas replikasi

`describe-replication-task-assessment-results` Contoh berikut mencantumkan hasil penilaian tugas sebelumnya.

```
aws dms describe-replication-task-assessment-results
```

Output:

```

{
  "ReplicationTaskAssessmentResults": [
    {
      "ReplicationTaskIdentifier": "moveit2",
      "ReplicationTaskArn": "arn:aws:dms:us-east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII",

```

```

        "ReplicationTaskLastAssessmentDate": 1590790230.0,
        "AssessmentStatus": "No issues found",
        "AssessmentResultsFile": "moveit2/2020-05-29-22-10"
    }
]
}

```

Untuk selengkapnya, lihat [Membuat Laporan Penilaian Tugas](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeReplicationTaskAssessmentResults](#) di Referensi AWS CLI Perintah.

## describe-replication-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-replication-tasks`.

### AWS CLI

Untuk menggambarkan tugas replikasi

`describe-replication-tasks` Contoh berikut menjelaskan tugas replikasi saat ini.

```
aws dms describe-replication-tasks
```

Output:

```

{
  "ReplicationTasks": [
    {
      "ReplicationTaskIdentifier": "moveit2",
      "SourceEndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:6GGI6YPWYGAYUVLKIB732KEVWA",
      "TargetEndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
      "ReplicationInstanceArn": "arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
      "MigrationType": "full-load",
      "TableMappings": "...output omitted... ",
      "ReplicationTaskSettings": "...output omitted... ",
      "Status": "stopped",
      "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
    }
  ]
}

```

```

    "ReplicationTaskCreationDate": 1590524772.505,
    "ReplicationTaskStartDate": 1590619805.212,
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII",
    "ReplicationTaskStats": {
      "FullLoadProgressPercent": 100,
      "ElapsedTimeMillis": 0,
      "TablesLoaded": 0,
      "TablesLoading": 0,
      "TablesQueued": 0,
      "TablesErrored": 0,
      "FreshStartDate": 1590619811.528,
      "StartDate": 1590619811.528,
      "StopDate": 1590619842.068
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tugas AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeReplicationTasks](#) di Referensi AWS CLI Perintah.

## describe-schemas

Contoh kode berikut menunjukkan cara menggunakan `describe-schemas`.

### AWS CLI

Untuk membuat skema basis data

`describe-schemas` Contoh berikut mencantumkan tabel yang tersedia di titik akhir.

```

aws dms describe-schemas \
  --endpoint-arn "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA"

```

Output:

```

{
  "Schemas": [

```

```
    "prodrep"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Ini adalah judul topik](#) dalam Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [DescribeSchemas](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk contoh replikasi.

```
aws dms list-tags-for-resource \  
  --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE
```

Output:

```
{  
  "TagList": [  
    {  
      "Key": "Project",  
      "Value": "dbMigration"  
    },  
    {  
      "Key": "Environment",  
      "Value": "PROD"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## modify-endpoint

Contoh kode berikut menunjukkan cara menggunakan `modify-endpoint`.

### AWS CLI

Untuk memodifikasi titik akhir

`modify-endpoint` Contoh berikut menambahkan atribut koneksi tambahan ke titik akhir.

```
aws dms modify-endpoint \
  --endpoint-arn "arn:aws:dms:us-
  east-1:123456789012:endpoint:GUVAFG34EECU0J6QVZ56DAHT3U" \
  --extra-connection-attributes "compressionType=GZIP"
```

Output:

```
{
  "Endpoint": {
    "EndpointIdentifier": "src-endpoint",
    "EndpointType": "SOURCE",
    "EngineName": "s3",
    "EngineDisplayName": "Amazon S3",
    "ExtraConnectionAttributes":
"compressionType=GZIP;csvDelimiter=,;csvRowDelimiter=\n;",
    "Status": "active",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECU0J6QVZ56DAHT3U",
    "SslMode": "none",
    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
    "S3Settings": {
      "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
      "CsvRowDelimiter": "\n",
      "CsvDelimiter": ",",
      "BucketFolder": "",
      "BucketName": "",
      "CompressionType": "GZIP",
      "EnableStatistics": true
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS DMS Endpoints`\\_\\_](#) di Panduan Pengguna Layanan Migrasi Database AWS . < [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Endpoints.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Endpoints.html)>

- Untuk detail API, lihat [ModifyEndpoint](#) di Referensi AWS CLI Perintah.

## **modify-event-subscription**

Contoh kode berikut menunjukkan cara menggunakan `modify-event-subscription`.

### AWS CLI

Untuk memodifikasi langganan acara

`modify-event-subscription` Contoh berikut mengubah jenis sumber langganan acara.

```
aws dms modify-event-subscription \  
  --subscription-name "my-dms-events" \  
  --source-type replication-task
```

Output:

```
{  
  "EventSubscription": {  
    "CustomerAwsId": "123456789012",  
    "CustSubscriptionId": "my-dms-events",  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",  
    "Status": "modifying",  
    "SubscriptionCreationTime": "2020-05-29 17:04:40.262",  
    "SourceType": "replication-task",  
    "Enabled": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Acara dan Pemberitahuan](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [ModifyEventSubscription](#) di Referensi AWS CLI Perintah.

## **modify-replication-instance**

Contoh kode berikut menunjukkan cara menggunakan `modify-replication-instance`.



## AWS CLI

Untuk memodifikasi contoh replikasi

`modify-replication-instance` Contoh berikut memodifikasi instance replikasi sehingga menggunakan penerapan Multi-AZ.

```
aws dms modify-replication-instance \  
  --replication-instance-arn arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \  
  --multi-az
```

Output:

```
{  
  "ReplicationInstance": {  
    "ReplicationInstanceIdentifier": "my-repl-instance",  
    "ReplicationInstanceClass": "dms.t2.micro",  
    "ReplicationInstanceStatus": "available",  
    "AllocatedStorage": 5,  
    "InstanceCreateTime": 1590011235.952,  
  
    ...output omitted...  
  
    "PendingModifiedValues": {  
      "MultiAZ": true  
    },  
    "MultiAZ": false,  
    "EngineVersion": "3.3.2",  
    "AutoMinorVersionUpgrade": true,  
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-  
e8494fa3921a",  
  
    ...output omitted...  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instance Replikasi AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [ModifyReplicationInstance](#) di Referensi AWS CLI Perintah.

## modify-replication-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `modify-replication-subnet-group`.

### AWS CLI

Untuk memodifikasi grup subnet

`modify-replication-subnet-group` Contoh berikut mengubah daftar subnet yang terkait dengan grup subnet.

```
aws dms modify-replication-subnet-group \  
  --replication-subnet-group-identifier my-subnet-group \  
  --subnet-id subnet-da327bf6 subnet-bac383e0
```

Output:

```
{  
  "ReplicationSubnetGroup": {  
    "ReplicationSubnetGroupIdentifier": "my-subnet-group",  
    "ReplicationSubnetGroupDescription": "my subnet group",  
    "VpcId": "vpc-136a4c6a",  
    "SubnetGroupStatus": "Complete",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-da327bf6",  
        "SubnetAvailabilityZone": {  
          "Name": "us-east-1a"  
        },  
        "SubnetStatus": "Active"  
      },  
      {  
        "SubnetIdentifier": "subnet-bac383e0",  
        "SubnetAvailabilityZone": {  
          "Name": "us-east-1c"  
        },  
        "SubnetStatus": "Active"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Jaringan untuk Instance Replikasi](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [ModifyReplicationSubnetGroup](#) di Referensi AWS CLI Perintah.

## modify-replication-task

Contoh kode berikut menunjukkan cara menggunakan `modify-replication-task`.

### AWS CLI

Untuk memodifikasi tugas replikasi

`modify-replication-task` Contoh berikut mengubah pemetaan tabel untuk tugas.

```
aws dms modify-replication-task \  
  --replication-task-arn "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII" \  
  --table-mappings file://table-mappings.json
```

Isi dari `table-mappings.json`:

```
{  
  "rules": [  
    {  
      "rule-type": "selection",  
      "rule-id": "1",  
      "rule-name": "1",  
      "object-locator": {  
        "schema-name": "prodrep",  
        "table-name": "ACCT_%"  
      },  
      "rule-action": "include",  
      "filters": []  
    }  
  ]  
}
```

Output:

```
{
```

```

    "ReplicationTask": {
      "ReplicationTaskIdentifier": "moveit2",
      "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",
      "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
      "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
      "MigrationType": "full-load",
      "TableMappings": ...output omitted...,
      "ReplicationTaskSettings": ...output omitted...,
      "Status": "modifying",
      "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
      "ReplicationTaskCreationDate": 1590524772.505,
      "ReplicationTaskStartDate": 1590789424.653,
      "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tugas AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [ModifyReplicationTask](#) di Referensi AWS CLI Perintah.

## reboot-replication-instance

Contoh kode berikut menunjukkan cara menggunakan `reboot-replication-instance`.

### AWS CLI

Untuk me-reboot instance replikasi

Contoh `reboot-replication-instance` berikut mem-boot ulang instans replikasi.

```

aws dms reboot-replication-instance \
  --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE

```

Output:

```
{
```

```

"ReplicationInstance": {
  "ReplicationInstanceIdentifier": "my-repl-instance",
  "ReplicationInstanceClass": "dms.t2.micro",
  "ReplicationInstanceStatus": "rebooting",
  "AllocatedStorage": 5,
  "InstanceCreateTime": 1590011235.952,
  ... output omitted ...
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instance Replikasi AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [RebootReplicationInstance](#) di Referensi AWS CLI Perintah.

## refresh-schemas

Contoh kode berikut menunjukkan cara menggunakan `refresh-schemas`.

### AWS CLI

Untuk me-refresh skema database

`refresh-schemas` Contoh berikut meminta agar AWS DMS menyegarkan daftar skema pada titik akhir.

```

aws dms refresh-schemas \
  --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \
  --endpoint-arn "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA"

```

Output:

```

{
  "RefreshSchemasStatus": {
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "Status": "refreshing",
    "LastRefreshDate": 1590019949.103
  }
}

```

```
}  
}
```

- Untuk detail API, lihat [RefreshSchemas](#) di Referensi AWS CLI Perintah.

## reload-tables

Contoh kode berikut menunjukkan cara menggunakan `reload-tables`.

### AWS CLI

Untuk menyegarkan daftar tabel yang tersedia di titik akhir

`reload-tables` Contoh berikut memuat ulang daftar tabel yang tersedia di titik akhir.

```
aws dms reload-tables \  
  --replication-task-arn "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII" \  
  --tables-to-reload "SchemaName=prodrep,TableName=ACCT_BAL"
```

Output:

```
{  
  "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"  
}
```

- Untuk detail API, lihat [ReloadTables](#) di Referensi AWS CLI Perintah.

## remove-tags-from-resource

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-resource`.

### AWS CLI

Untuk menghapus tag dari contoh replikasi

`remove-tags-from-resource` Contoh berikut menghapus tag dari contoh replikasi.

```
aws dms remove-tags-from-resource \  

```

```
--resource-arn arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE  
\n--tag-keys Environment Project
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [RemoveTagsFromResource](#) di Referensi AWS CLI Perintah.

## start-replication-task-assessment

Contoh kode berikut menunjukkan cara menggunakan `start-replication-task-assessment`.

### AWS CLI

Untuk memulai penilaian tugas

`start-replication-task-assessment` Contoh berikut memulai penilaian tugas replikasi.

```
aws dms start-replication-task-assessment \  
  --replication-task-arn arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Output:

```
{  
  "ReplicationTask": {  
    "ReplicationTaskIdentifier": "moveit2",  
    "SourceEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",  
    "TargetEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "MigrationType": "full-load",  
    "TableMappings": "...output omitted...",  
    "ReplicationTaskSettings": "...output omitted...",  
    "Status": "testing",  
    "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",  
    "ReplicationTaskCreationDate": 1590524772.505,
```

```

    "ReplicationTaskStartDate": 1590789988.677,
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
  }
}

```

Untuk selengkapnya, lihat [Membuat Laporan Penilaian Tugas](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [StartReplicationTaskAssessment](#) di Referensi AWS CLI Perintah.

## start-replication-task

Contoh kode berikut menunjukkan cara menggunakan `start-replication-task`.

### AWS CLI

Untuk memulai tugas replikasi

command-name Contoh berikut mencantumkan widget yang tersedia di AWS akun Anda.

```

aws dms start-replication-task \
  --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII \
  --start-replication-task-type reload-target

```

Output:

```

{
  "ReplicationTask": {
    "ReplicationTaskIdentifier": "moveit2",
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWYGAYUVLKIB732KEVWA",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "MigrationType": "full-load",
    "TableMappings": "...output omitted... ",
    "ReplicationTaskSettings": "...output omitted... ",
    "Status": "starting",
    "ReplicationTaskCreationDate": 1590524772.505,
  }
}

```



```

    "ReplicationTaskStartDate": 1590619805.212,
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tugas AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [StartReplicationTask](#) di Referensi AWS CLI Perintah.

## stop-replication-task

Contoh kode berikut menunjukkan cara menggunakan `stop-replication-task`.

### AWS CLI

Untuk menghentikan tugas

`stop-replication-task` Contoh berikut menghentikan tugas.

```

aws dms stop-replication-task \
  --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII

```

Output:

```

{
  "ReplicationTask": {
    "ReplicationTaskIdentifier": "moveit2",
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "MigrationType": "full-load",
    "TableMappings": "...output omitted...",
    "ReplicationTaskSettings": "...output omitted...",
    "Status": "stopping",
    "ReplicationTaskCreationDate": 1590524772.505,
    "ReplicationTaskStartDate": 1590789424.653,

```

```

    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tugas AWS DMS](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [StopReplicationTask](#) di Referensi AWS CLI Perintah.

## test-connection

Contoh kode berikut menunjukkan cara menggunakan `test-connection`.

### AWS CLI

Untuk menguji koneksi ke titik akhir

`test-connection` Contoh berikut menguji apakah titik akhir dapat diakses dari contoh replikasi.

```

aws dms test-connection \
  --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUe \
  --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA

```

Output:

```

{
  "Connection": {
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUe",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",
    "Status": "testing",
    "EndpointIdentifier": "src-database-1",
    "ReplicationInstanceIdentifier": "my-repl-instance"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat titik akhir sumber dan target](#) di Panduan Pengguna Layanan Migrasi AWS Database.

- Untuk detail API, lihat [TestConnection](#) di Referensi AWS CLI Perintah.

## Amazon DocumentDB contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon DocumentDB. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags-to-resource**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-resource`.

AWS CLI

Untuk menambahkan satu atau beberapa tag ke sumber daya tertentu

`add-tags-to-resource` Contoh berikut menambahkan tiga tag ke `sample-cluster`. Satu tag (`CropB`) memiliki nama kunci tetapi tidak memiliki nilai.

```
aws docdb add-tags-to-resource \  
  --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \  
  --tags Key="CropA",Value="Apple" Key="CropB" Key="CropC",Value="Corn"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [AddTagsToResource](#) di Referensi AWS CLI Perintah.

## **apply-pending-maintenance-action**

Contoh kode berikut menunjukkan cara menggunakan `apply-pending-maintenance-action`.

### AWS CLI

Untuk melakukan tindakan pemeliharaan yang tertunda selama jendela pemeliharaan berikutnya

`apply-pending-maintenance-action` Contoh berikut menyebabkan semua tindakan pembaruan sistem dilakukan selama jendela pemeliharaan terjadwal berikutnya.

```
aws docdb apply-pending-maintenance-action \  
--resource-identifier arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \  
--apply-action system-update \  
--opt-in-type next-maintenance
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menerapkan Pembaruan Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [ApplyPendingMaintenanceAction](#) di Referensi AWS CLI Perintah.

## **copy-db-cluster-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `copy-db-cluster-parameter-group`.

### AWS CLI

Untuk menduplikasi grup parameter cluster DB yang ada

`copy-db-cluster-parameter-group` Contoh berikut membuat salinan dari kelompok parameter `custom-docdb3-6` bernama `custom-docdb3-6-copy`. Saat membuat salinan itu menambahkan tag ke grup parameter baru.

```
aws docdb copy-db-cluster-parameter-group \  
--source-db-cluster-parameter-group-identifier custom-docdb3-6 \  
--target-db-cluster-parameter-group-identifier custom-docdb3-6-copy \  
--target-db-cluster-parameter-group-description "Copy of custom-docdb3-6" \  
--tags Key="CopyNumber",Value="1" Key="Modifiable",Value="Yes"
```

**Output:**

```
{
  "DBClusterParameterGroup": {
    "DBParameterGroupFamily": "docdb3.6",
    "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:12345678901:cluster-pg:custom-docdb3-6-copy",
    "DBClusterParameterGroupName": "custom-docdb3-6-copy",
    "Description": "Copy of custom-docdb3-6"
  }
}
```

Untuk informasi selengkapnya, lihat [Menyalin Grup Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CopyDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

**copy-db-cluster-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `copy-db-cluster-snapshot`.

**AWS CLI**

Untuk membuat salinan snapshot

`copy-db-cluster-snapshot` Contoh berikut membuat salinan `sample-cluster-snapshot` bernama `sample-cluster-snapshot-copy`. Salinan memiliki semua tag asli ditambah tag baru dengan nama kunci `CopyNumber`.

```
aws docdb copy-db-cluster-snapshot \
  --source-db-cluster-snapshot-identifier sample-cluster-snapshot \
  --target-db-cluster-snapshot-identifier sample-cluster-snapshot-copy \
  --copy-tags \
  --tags Key="CopyNumber",Value="1"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyalin Snapshot Cluster](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CopyDbClusterSnapshot](#) di Referensi AWS CLI Perintah.

## create-db-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster-parameter-group`.

### AWS CLI

Untuk membuat grup parameter cluster Amazon DocumentDB

`create-db-cluster-parameter-group` Contoh berikut membuat kelompok parameter cluster DB `sample-parameter-group` menggunakan `docdb3.6` keluarga.

```
aws docdb create-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name sample-parameter-group \  
  --db-parameter-group-family docdb3.6 \  
  --description "Sample parameter group based on docdb3.6"
```

Output:

```
{  
  "DBClusterParameterGroup": {  
    "Description": "Sample parameter group based on docdb3.6",  
    "DBParameterGroupFamily": "docdb3.6",  
    "DBClusterParameterGroupArn": "arn:aws:rds:us-west-2:123456789012:cluster-  
pg:sample-parameter-group",  
    "DBClusterParameterGroupName": "sample-parameter-group"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CreateDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## create-db-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster-snapshot`.

### AWS CLI

Untuk membuat snapshot cluster Amazon DocumentDB manual

`create-db-cluster-snapshot` Contoh berikut membuat snapshot cluster Amazon DB bernama `sample-cluster-snapshot`.

```
aws docdb create-db-cluster-snapshot \  
  --db-cluster-identifier sample-cluster \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{  
  "DBClusterSnapshot": {  
    "MasterUsername": "master-user",  
    "SnapshotCreateTime": "2019-03-18T18:27:14.794Z",  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2b",  
      "us-west-2c",  
      "us-west-2d",  
      "us-west-2e",  
      "us-west-2f"  
    ],  
    "SnapshotType": "manual",  
    "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-  
snapshot:sample-cluster-snapshot",  
    "EngineVersion": "3.6.0",  
    "PercentProgress": 0,  
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",  
    "Engine": "docdb",  
    "DBClusterIdentifier": "sample-cluster",  
    "Status": "creating",  
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z",  
    "Port": 0,  
    "StorageEncrypted": false,  
    "VpcId": "vpc-91280df6"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Snapshot Kluster Manual](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CreateDbClusterSnapshot](#) di Referensi AWS CLI Perintah.

## create-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster`.

## AWS CLI

Untuk membuat cluster Amazon DocumentDB

`create-db-cluster` Contoh berikut membuat cluster Amazon DocumentDB `sample-cluster` bernama dengan jendela pemeliharaan pilihan pada hari Minggu antara 20:30 dan 11:00.

```
aws docdb create-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --engine docdb \  
  --master-username master-user \  
  --master-user-password password \  
  --preferred-maintenance-window Sun:20:30-Sun:21:00
```

Output:

```
{  
  "DBCluster": {  
    "DBClusterParameterGroup": "default.docdb3.6",  
    "AssociatedRoles": [],  
    "DBSubnetGroup": "default",  
    "ClusterCreateTime": "2019-03-18T18:06:34.616Z",  
    "Status": "creating",  
    "Port": 27017,  
    "PreferredMaintenanceWindow": "sun:20:30-sun:21:00",  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "DBClusterMembers": [],  
    "Engine": "docdb",  
    "DBClusterIdentifier": "sample-cluster",  
    "PreferredBackupWindow": "10:12-10:42",  
    "AvailabilityZones": [  
      "us-west-2d",  
      "us-west-2f",  
      "us-west-2e"  
    ],  
    "MasterUsername": "master-user",  
    "BackupRetentionPeriod": 1,  
    "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-77186e0d",  
        "Status": "active"  
      }  
    ]  
  }  
}
```



```

    }
  ],
  "StorageEncrypted": false,
  "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
  "DbClusterResourceId": "cluster-L3R4YRSBUYDP4GLMTJ2WF5GH5Q",
  "MultiAZ": false,
  "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "EngineVersion": "3.6.0"
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CreateDbCluster](#) di Referensi AWS CLI Perintah.

## create-db-instance

Contoh kode berikut menunjukkan cara menggunakan `create-db-instance`.

### AWS CLI

Untuk membuat instance cluster Amazon DocumentDB

Kode `create-db-instance` contoh berikut membuat instance `sample-cluster-instance-2` di cluster Amazon DocumentDB. `sample-cluster`

```

aws docdb create-db-instance \
  --db-cluster-identifier sample-cluster \
  --db-instance-class db.r4.xlarge \
  --db-instance-identifier sample-cluster-instance-2 \
  --engine docdb

```

Output:

```

{
  "DBInstance": {
    "DBInstanceStatus": "creating",
    "PendingModifiedValues": {
      "PendingCloudwatchLogsExports": {
        "LogTypesToEnable": [
          "audit"
        ]
      }
    }
  }
}

```

```
    ]
  }
},
"PubliclyAccessible": false,
"PreferredBackupWindow": "00:00-00:30",
"PromotionTier": 1,
"EngineVersion": "3.6.0",
"BackupRetentionPeriod": 3,
"DBInstanceIdentifier": "sample-cluster-instance-2",
"PreferredMaintenanceWindow": "tue:10:28-tue:10:58",
"StorageEncrypted": false,
"Engine": "docdb",
"DBClusterIdentifier": "sample-cluster",
"DBSubnetGroup": {
  "Subnets": [
    {
      "SubnetAvailabilityZone": {
        "Name": "us-west-2a"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-4e26d263"
    },
    {
      "SubnetAvailabilityZone": {
        "Name": "us-west-2c"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-afc329f4"
    },
    {
      "SubnetAvailabilityZone": {
        "Name": "us-west-2d"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-53ab3636"
    },
    {
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-991cb8d0"
    }
  ]
},
```

```

        "DBSubnetGroupDescription": "default",
        "SubnetGroupStatus": "Complete",
        "VpcId": "vpc-91280df6",
        "DBSubnetGroupName": "default"
    },
    "DBInstanceClass": "db.r4.xlarge",
    "VpcSecurityGroups": [
        {
            "Status": "active",
            "VpcSecurityGroupId": "sg-77186e0d"
        }
    ],
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance-2",
    "DbiResourceId": "db-XEKJLEMGRV5ZKCARUVA4H03ITE"
}
}

```

Untuk informasi selengkapnya, lihat [Menambahkan Instans Amazon DocumentDB ke Cluster](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CreateDbInstance](#) di Referensi AWS CLI Perintah.

## create-db-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `create-db-subnet-group`.

### AWS CLI

Untuk membuat grup subnet Amazon DocumentDB

`create-db-subnet-group` Contoh berikut membuat grup subnet Amazon DocumentDB bernama `sample-subnet-group`

```

aws docdb create-db-subnet-group \
  --db-subnet-group-description "a sample subnet group" \
  --db-subnet-group-name sample-subnet-group \
  --subnet-ids "subnet-29ab1025" "subnet-991cb8d0" "subnet-53ab3636"

```

Output:

```
{
```

```
"DBSubnetGroup": {
  "SubnetGroupStatus": "Complete",
  "DBSubnetGroupName": "sample-subnet-group",
  "DBSubnetGroupDescription": "a sample subnet group",
  "VpcId": "vpc-91280df6",
  "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:sample-
subnet-group",
  "Subnets": [
    {
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-53ab3636",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2d"
      }
    },
    {
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-991cb8d0",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      }
    },
    {
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-29ab1025",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2c"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Subnet Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [CreateDbSubnetGroup](#) di Referensi AWS CLI Perintah.

## **delete-db-cluster-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster-parameter-group`.

## AWS CLI

Untuk menghapus grup parameter cluster Amazon DocumentDB

`delete-db-cluster-parameter-group` Contoh berikut menghapus grup parameter Amazon `sample-parameter-group` DocumentDB.

```
aws docdb delete-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name sample-parameter-group
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Grup Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DeleteDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## **delete-db-cluster-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster-snapshot`.

## AWS CLI

Untuk menghapus snapshot cluster Amazon DocumentDB

`delete-db-cluster-snapshot` Contoh berikut menghapus snapshot cluster Amazon `sample-cluster-snapshot` DocumentDB.

```
aws docdb delete-db-cluster-snapshot \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{  
  "DBClusterSnapshot": {  
    "DBClusterIdentifier": "sample-cluster",  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2b",  
      "us-west-2c",  
      "us-west-2d"  
    ],  
  },  
}
```

```
"DBClusterSnapshotIdentifier": "sample-cluster-snapshot",
  "VpcId": "vpc-91280df6",
  "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-
snapshot:sample-cluster-snapshot",
  "EngineVersion": "3.6.0",
  "Engine": "docdb",
  "SnapshotCreateTime": "2019-03-18T18:27:14.794Z",
  "Status": "available",
  "MasterUsername": "master-user",
  "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
  "PercentProgress": 100,
  "StorageEncrypted": false,
  "SnapshotType": "manual",
  "Port": 0
}
}
```

Untuk informasi selengkapnya, lihat [Menghapus Snapshot Cluster](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DeleteDbClusterSnapshot](#) di Referensi AWS CLI Perintah.

## delete-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster`.

### AWS CLI

Untuk menghapus cluster Amazon DocumentDB

`delete-db-cluster` Contoh berikut menghapus cluster Amazon DocumentDB. `sample-cluster` Tidak ada cadangan cluster yang dibuat sebelum menghapusnya. CATATAN: Anda harus menghapus semua instance yang terkait dengan cluster sebelum Anda dapat menghapusnya.

```
aws docdb delete-db-cluster \
  --db-cluster-identifier sample-cluster \
  --skip-final-snapshot
```

Output:

```
{
```

```
"DBCluster": {
  "DBClusterIdentifier": "sample-cluster",
  "DBSubnetGroup": "default",
  "EngineVersion": "3.6.0",
  "Engine": "docdb",
  "LatestRestorableTime": "2019-03-18T18:07:24.610Z",
  "PreferredMaintenanceWindow": "sun:20:30-sun:21:00",
  "StorageEncrypted": false,
  "EarliestRestorableTime": "2019-03-18T18:07:24.610Z",
  "Port": 27017,
  "VpcSecurityGroups": [
    {
      "Status": "active",
      "VpcSecurityGroupId": "sg-77186e0d"
    }
  ],
  "MultiAZ": false,
  "MasterUsername": "master-user",
  "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
  "Status": "available",
  "PreferredBackupWindow": "10:12-10:42",
  "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "AvailabilityZones": [
    "us-west-2c",
    "us-west-2b",
    "us-west-2a"
  ],
  "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "DbClusterResourceId": "cluster-L3R4YRSBUYDP4GLMTJ2WF5GH5Q",
  "ClusterCreateTime": "2019-03-18T18:06:34.616Z",
  "AssociatedRoles": [],
  "DBClusterParameterGroup": "default.docdb3.6",
  "HostedZoneId": "ZNKXH85TT8WW",
  "BackupRetentionPeriod": 1,
  "DBClusterMembers": []
}
}
```

Untuk informasi selengkapnya, lihat [Menghapus Cluster Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DeleteDbCluster](#) di Referensi AWS CLI Perintah.

## delete-db-instance

Contoh kode berikut menunjukkan cara menggunakan `delete-db-instance`.

### AWS CLI

Untuk menghapus instance Amazon DocumentDB

`delete-db-instance` Contoh berikut menghapus instance Amazon DocumentDB. `sample-cluster-instance-2`

```
aws docdb delete-db-instance \
  --db-instance-identifier sample-cluster-instance-2
```

Output:

```
{
  "DBInstance": {
    "DBSubnetGroup": {
      "Subnets": [
        {
          "SubnetAvailabilityZone": {
            "Name": "us-west-2a"
          },
          "SubnetStatus": "Active",
          "SubnetIdentifier": "subnet-4e26d263"
        },
        {
          "SubnetAvailabilityZone": {
            "Name": "us-west-2c"
          },
          "SubnetStatus": "Active",
          "SubnetIdentifier": "subnet-afc329f4"
        },
        {
          "SubnetAvailabilityZone": {
            "Name": "us-west-2d"
          },
          "SubnetStatus": "Active",
          "SubnetIdentifier": "subnet-53ab3636"
        },
        {
          "SubnetAvailabilityZone": {
```



```

        "Name": "us-west-2b"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-991cb8d0"
    }
  ],
  "DBSubnetGroupName": "default",
  "DBSubnetGroupDescription": "default",
  "VpcId": "vpc-91280df6",
  "SubnetGroupStatus": "Complete"
},
"PreferredBackupWindow": "00:00-00:30",
"InstanceCreateTime": "2019-03-18T18:37:33.709Z",
"DBInstanceClass": "db.r4.xlarge",
"DbiResourceId": "db-XEKJLEMGRV5ZKCARUVA4H03ITE",
"BackupRetentionPeriod": 3,
"Engine": "docdb",
"VpcSecurityGroups": [
  {
    "Status": "active",
    "VpcSecurityGroupId": "sg-77186e0d"
  }
],
"AutoMinorVersionUpgrade": true,
"PromotionTier": 1,
"EngineVersion": "3.6.0",
"Endpoint": {
  "Address": "sample-cluster-instance-2.corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "HostedZoneId": "ZNKXH85TT8WVW",
  "Port": 27017
},
"DBInstanceIdentifier": "sample-cluster-instance-2",
"PreferredMaintenanceWindow": "tue:10:28-tue:10:58",
"EnabledCloudwatchLogsExports": [
  "audit"
],
"PendingModifiedValues": {},
"DBInstanceStatus": "deleting",
"PubliclyAccessible": false,
"DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance-2",
"DBClusterIdentifier": "sample-cluster",
"AvailabilityZone": "us-west-2c",

```

```
    "StorageEncrypted": false
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus Instans Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DeleteDbInstance](#) di Referensi AWS CLI Perintah.

## **delete-db-subnet-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-subnet-group`.

### AWS CLI

Untuk menghapus grup subnet Amazon DocumentDB

`delete-db-subnet-group` Contoh berikut menghapus grup subnet Amazon `sample-subnet-group` DocumentDB.

```
aws docdb delete-db-subnet-group \  
  --db-subnet-group-name sample-subnet-group
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Grup Subnet Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DeleteDbSubnetGroup](#) di Referensi AWS CLI Perintah.

## **describe-db-cluster-parameter-groups**

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-parameter-groups`.

### AWS CLI

Untuk melihat detail satu atau beberapa grup parameter cluster Amazon DocumentDB

`describe-db-cluster-parameter-groups` Contoh berikut menampilkan detail untuk grup parameter cluster Amazon DocumentDB. `custom3-6-param-grp`

```
aws docdb describe-db-cluster-parameter-groups \  
  --db-cluster-parameter-group-name custom3-6-param-grp
```

Output:

```
{  
  "DBClusterParameterGroups": [  
    {  
      "DBParameterGroupFamily": "docdb3.6",  
      "DBClusterParameterGroupArn": "arn:aws:rds:us-  
east-1:123456789012:cluster-pg:custom3-6-param-grp",  
      "Description": "Custom docdb3.6 parameter group",  
      "DBClusterParameterGroupName": "custom3-6-param-grp"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat Grup Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribeDbClusterParameterGroups](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-parameters`.

AWS CLI

Untuk melihat daftar parameter terperinci untuk grup parameter cluster Amazon DocumentDB.

`describe-db-cluster-parameters` Contoh berikut mencantumkan parameter untuk grup parameter Amazon DocumentDB `custom3-6-param-grp`.

```
aws docdb describe-db-cluster-parameters \  
  --db-cluster-parameter-group-name custom3-6-param-grp
```

Output:

```
{  
  "Parameters": [  
    {  
      "DataType": "string",
```

```

    "ParameterName": "audit_logs",
    "IsModifiable": true,
    "ApplyMethod": "pending-reboot",
    "Source": "system",
    "ApplyType": "dynamic",
    "AllowedValues": "enabled,disabled",
    "Description": "Enables auditing on cluster.",
    "ParameterValue": "disabled"
  },
  {
    "DataType": "string",
    "ParameterName": "tls",
    "IsModifiable": true,
    "ApplyMethod": "pending-reboot",
    "Source": "system",
    "ApplyType": "static",
    "AllowedValues": "disabled,enabled",
    "Description": "Config to enable/disable TLS",
    "ParameterValue": "enabled"
  },
  {
    "DataType": "string",
    "ParameterName": "ttl_monitor",
    "IsModifiable": true,
    "ApplyMethod": "pending-reboot",
    "Source": "user",
    "ApplyType": "dynamic",
    "AllowedValues": "disabled,enabled",
    "Description": "Enables TTL Monitoring",
    "ParameterValue": "enabled"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribeDbClusterParameters](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-snapshot-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-snapshot-attributes`.

## AWS CLI

Untuk mencantumkan nama dan nilai atribut snapshot Amazon DocumentDB

`describe-db-cluster-snapshot-attributes` Contoh berikut mencantumkan nama atribut dan nilai untuk snapshot Amazon `sample-cluster-snapshot` DocumentDB.

```
aws docdb describe-db-cluster-snapshot-attributes \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{  
  "DBClusterSnapshotAttributesResult": {  
    "DBClusterSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": []  
      }  
    ],  
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"  
  }  
}
```

Untuk informasi selengkapnya, lihat [DescribeDB ClusterSnapshotAttributes di Panduan Pengembang](#) Amazon DocumentDB.

- Untuk detail API, lihat [DescribeDbClusterSnapshotAttributes](#) di Referensi AWS CLI Perintah.

## `describe-db-cluster-snapshots`

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-snapshots`.

## AWS CLI

Untuk menggambarkan snapshot Amazon DocumentDB

`describe-db-cluster-snapshots` Contoh berikut menampilkan detail untuk snapshot Amazon DocumentDB. `sample-cluster-snapshot`

```
aws docdb describe-db-cluster-snapshots \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot
```

**Output:**

```
{
  "DBClusterSnapshots": [
    {
      "AvailabilityZones": [
        "us-west-2a",
        "us-west-2b",
        "us-west-2c",
        "us-west-2d"
      ],
      "Status": "available",
      "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-snapshot:sample-cluster-snapshot",
      "SnapshotCreateTime": "2019-03-15T20:41:26.515Z",
      "SnapshotType": "manual",
      "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",
      "DBClusterIdentifier": "sample-cluster",
      "MasterUsername": "master-user",
      "StorageEncrypted": false,
      "VpcId": "vpc-91280df6",
      "EngineVersion": "3.6.0",
      "PercentProgress": 100,
      "Port": 0,
      "Engine": "docdb",
      "ClusterCreateTime": "2019-03-15T20:29:58.836Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [DescribeDB ClusterSnapshots di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DescribeDbClusterSnapshots](#) di Referensi AWS CLI Perintah.

**describe-db-clusters**

Contoh kode berikut menunjukkan cara menggunakan `describe-db-clusters`.

**AWS CLI**

Untuk mendapatkan informasi rinci tentang satu atau beberapa cluster Amazon DocumentDB.

`describe-db-clusters` Contoh berikut menampilkan detail untuk cluster Amazon DocumentDB. `sample-cluster` Dengan menghilangkan `--db-cluster-identifier` parameter Anda bisa mendapatkan informasi hingga 100 cluster.

```
aws docdb describe-db-clusters
  --db-cluster-identifier sample-cluster
```

Output:

```
{
  "DBClusters": [
    {
      "DBClusterParameterGroup": "default.docdb3.6",
      "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
      "PreferredBackupWindow": "00:00-00:30",
      "DBClusterIdentifier": "sample-cluster",
      "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
      "LatestRestorableTime": "2019-03-18T20:28:03.239Z",
      "MasterUsername": "master-user",
      "DBClusterMembers": [
        {
          "PromotionTier": 1,
          "DBClusterParameterGroupStatus": "in-sync",
          "IsClusterWriter": false,
          "DBInstanceIdentifier": "sample-cluster"
        },
        {
          "PromotionTier": 1,
          "DBClusterParameterGroupStatus": "in-sync",
          "IsClusterWriter": true,
          "DBInstanceIdentifier": "sample-cluster2"
        }
      ],
      "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
      "VpcSecurityGroups": [
        {
          "VpcSecurityGroupId": "sg-77186e0d",
          "Status": "active"
        }
      ],
      "Engine": "docdb",
```

```

        "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "DBSubnetGroup": "default",
        "MultiAZ": true,
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2c",
            "us-west-2b"
        ],
        "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",
        "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
        "BackupRetentionPeriod": 3,
        "HostedZoneId": "ZNKXH85TT8WWW",
        "StorageEncrypted": false,
        "EnabledCloudwatchLogsExports": [
            "audit"
        ],
        "AssociatedRoles": [],
        "EngineVersion": "3.6.0",
        "Port": 27017,
        "Status": "available"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menjelaskan Cluster Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DescribeDbClusters](#) di Referensi AWS CLI Perintah.

## describe-db-engine-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-db-engine-versions`.

### AWS CLI

Untuk daftar versi mesin Amazon DocumentDB yang tersedia

`describe-db-engine-versions` Contoh berikut mencantumkan semua versi mesin Amazon DocumentDB yang tersedia.



```
aws docdb describe-db-engine-versions \  
  --engine docdb
```

Output:

```
{  
  "DBEngineVersions": [  
    {  
      "DBEngineVersionDescription": "DocDB version 1.0.200837",  
      "DBParameterGroupFamily": "docdb3.6",  
      "EngineVersion": "3.6.0",  
      "ValidUpgradeTarget": [],  
      "DBEngineDescription": "Amazon DocumentDB (with MongoDB compatibility)",  
      "SupportsLogExportsToCloudwatchLogs": true,  
      "Engine": "docdb",  
      "ExportableLogTypes": [  
        "audit"  
      ]  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [DescribeDB EngineVersions di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DescribeDbEngineVersions](#) di Referensi AWS CLI Perintah.

## describe-db-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-db-instances`.

### AWS CLI

Untuk menemukan informasi tentang instans Amazon DocumentDB yang disediakan

`describe-db-instances` Contoh berikut menampilkan detail tentang instance Amazon DocumentDB. `sample-cluster-instance` Dengan menghilangkan `--db-instance-identifier` parameter, Anda mendapatkan informasi hingga 100 instance.

```
aws docdb describe-db-instances \  
  --db-instance-identifier sample-cluster-instance
```

## Output:

```
{
  "DBInstances": [
    {
      "Endpoint": {
        "HostedZoneId": "ZNKXH85TT8WVW",
        "Address": "sample-cluster-instance.corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "Port": 27017
      },
      "PreferredBackupWindow": "00:00-00:30",
      "DBInstanceStatus": "available",
      "DBInstanceClass": "db.r4.large",
      "EnabledCloudwatchLogsExports": [
        "audit"
      ],
      "DBInstanceIdentifier": "sample-cluster-instance",
      "DBSubnetGroup": {
        "Subnets": [
          {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-4e26d263",
            "SubnetAvailabilityZone": {
              "Name": "us-west-2a"
            }
          },
          {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-afc329f4",
            "SubnetAvailabilityZone": {
              "Name": "us-west-2c"
            }
          },
          {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-53ab3636",
            "SubnetAvailabilityZone": {
              "Name": "us-west-2d"
            }
          },
          {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-991cb8d0",
```

```

        "SubnetAvailabilityZone": {
            "Name": "us-west-2b"
        }
    ],
    "DBSubnetGroupName": "default",
    "SubnetGroupStatus": "Complete",
    "DBSubnetGroupDescription": "default",
    "VpcId": "vpc-91280df6"
},
"InstanceCreateTime": "2019-03-15T20:36:06.338Z",
"Engine": "docdb",
"StorageEncrypted": false,
"AutoMinorVersionUpgrade": true,
"DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance",
"PreferredMaintenanceWindow": "tue:08:39-tue:09:09",
"VpcSecurityGroups": [
    {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
    }
],
"DBClusterIdentifier": "sample-cluster",
"PendingModifiedValues": {},
"BackupRetentionPeriod": 3,
"PubliclyAccessible": false,
"EngineVersion": "3.6.0",
"PromotionTier": 1,
"AvailabilityZone": "us-west-2c",
"DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA"
}
]
}

```

Untuk informasi selengkapnya, lihat [Menjelaskan Instans Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DescribeDbInstances](#) di Referensi AWS CLI Perintah.

## describe-db-subnet-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-subnet-groups`.

## AWS CLI

Untuk mengambil daftar deskripsi subnet Amazon DocumentDB

`describe-db-subnet-groups` Contoh berikut menjelaskan rincian untuk subnet Amazon DocumentDB bernama `default`

```
aws docdb describe-db-subnet-groups \  
--db-subnet-group-name default
```

Output:

```
{  
  "DBSubnetGroups": [  
    {  
      "VpcId": "vpc-91280df6",  
      "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:default",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-4e26d263",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2a"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-afc329f4",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2c"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-53ab3636",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2d"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-991cb8d0",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {
```

```

        "Name": "us-west-2b"
      }
    }
  ],
  "DBSubnetGroupName": "default",
  "SubnetGroupStatus": "Complete",
  "DBSubnetGroupDescription": "default"
}
]
}

```

Untuk informasi selengkapnya, lihat [Menjelaskan Grup Subnet](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribeDbSubnetGroups](#) di Referensi AWS CLI Perintah.

## describe-engine-default-cluster-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-engine-default-cluster-parameters`.

### AWS CLI

Untuk menjelaskan informasi parameter mesin dan sistem default untuk Amazon DocumentDB

`describe-engine-default-cluster-parameters` Contoh berikut menampilkan detail untuk mesin default dan informasi parameter sistem untuk grup parameter Amazon DocumentDB. `docdb3.6`

```
aws docdb describe-engine-default-cluster-parameters \
  --db-parameter-group-family docdb3.6
```

### Output:

```
{
  "EngineDefaults": {
    "DBParameterGroupFamily": "docdb3.6",
    "Parameters": [
      {
        "ApplyType": "dynamic",
        "ParameterValue": "disabled",

```

```

    "Description": "Enables auditing on cluster.",
    "Source": "system",
    "DataType": "string",
    "MinimumEngineVersion": "3.6.0",
    "AllowedValues": "enabled,disabled",
    "ParameterName": "audit_logs",
    "IsModifiable": true
  },
  {
    "ApplyType": "static",
    "ParameterValue": "enabled",
    "Description": "Config to enable/disable TLS",
    "Source": "system",
    "DataType": "string",
    "MinimumEngineVersion": "3.6.0",
    "AllowedValues": "disabled,enabled",
    "ParameterName": "tls",
    "IsModifiable": true
  },
  {
    "ApplyType": "dynamic",
    "ParameterValue": "enabled",
    "Description": "Enables TTL Monitoring",
    "Source": "system",
    "DataType": "string",
    "MinimumEngineVersion": "3.6.0",
    "AllowedValues": "disabled,enabled",
    "ParameterName": "ttl_monitor",
    "IsModifiable": true
  }
]
}

```

Untuk informasi selengkapnya, lihat [DescribeEngineDefaultClusterParameters](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribeEngineDefaultClusterParameters](#) di Referensi AWS CLI Perintah.

## describe-event-categories

Contoh kode berikut menunjukkan cara menggunakan `describe-event-categories`.

## AWS CLI

Untuk menggambarkan semua kategori acara Amazon DocumentDB

`describe-event-categories` Contoh berikut mencantumkan semua kategori untuk jenis sumber peristiwa Amazon DocumentDB. `db-instance`

```
aws docdb describe-event-categories \
  --source-type db-cluster
```

Output:

```
{
  "EventCategoriesMapList": [
    {
      "SourceType": "db-cluster",
      "EventCategories": [
        "failover",
        "maintenance",
        "notification",
        "failure"
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Kategori Acara](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribeEventCategories](#) di Referensi AWS CLI Perintah.

## `describe-events`

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

## AWS CLI

Untuk daftar acara Amazon DocumentDB

`describe-events` Contoh berikut mencantumkan semua acara Amazon DocumentDB selama 24 jam terakhir (1440 menit).

```
aws docdb describe-events \  
--duration 1440
```

Perintah ini tidak menghasilkan output. Output:

```
{  
  "Events": [  
    {  
      "EventCategories": [  
        "failover"  
      ],  
      "Message": "Started cross AZ failover to DB instance: sample-cluster",  
      "Date": "2019-03-18T21:36:29.807Z",  
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-  
cluster",  
      "SourceIdentifier": "sample-cluster",  
      "SourceType": "db-cluster"  
    },  
    {  
      "EventCategories": [  
        "availability"  
      ],  
      "Message": "DB instance restarted",  
      "Date": "2019-03-18T21:36:40.793Z",  
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster",  
      "SourceIdentifier": "sample-cluster",  
      "SourceType": "db-instance"  
    },  
    {  
      "EventCategories": [],  
      "Message": "A new writer was promoted. Restarting database as a  
reader.",  
      "Date": "2019-03-18T21:36:43.873Z",  
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",  
      "SourceIdentifier": "sample-cluster2",  
      "SourceType": "db-instance"  
    },  
    {  
      "EventCategories": [  
        "availability"  
      ],  
      "Message": "DB instance restarted",  
      "Date": "2019-03-18T21:36:51.257Z",
```



```

    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "failover"
    ],
    "Message": "Completed failover to DB instance: sample-cluster",
    "Date": "2019-03-18T21:36:53.462Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
    "SourceIdentifier": "sample-cluster",
    "SourceType": "db-cluster"
  },
  {
    "Date": "2019-03-19T16:51:48.847Z",
    "EventCategories": [
      "configuration change"
    ],
    "Message": "Updated parameter audit_logs to enabled with apply method
pending-reboot",
    "SourceIdentifier": "custom3-6-param-grp",
    "SourceType": "db-parameter-group"
  },
  {
    "EventCategories": [
      "configuration change"
    ],
    "Message": "Applying modification to database instance class",
    "Date": "2019-03-19T17:55:20.095Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "availability"
    ],
    "Message": "DB instance shutdown",
    "Date": "2019-03-19T17:56:31.127Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  }

```

```
},
{
  "EventCategories": [
    "configuration change"
  ],
  "Message": "Finished applying modification to DB instance class",
  "Date": "2019-03-19T18:00:45.822Z",
  "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
  "SourceIdentifier": "sample-cluster2",
  "SourceType": "db-instance"
},
{
  "EventCategories": [
    "availability"
  ],
  "Message": "DB instance restarted",
  "Date": "2019-03-19T18:00:53.397Z",
  "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
  "SourceIdentifier": "sample-cluster2",
  "SourceType": "db-instance"
},
{
  "EventCategories": [
    "availability"
  ],
  "Message": "DB instance shutdown",
  "Date": "2019-03-19T18:23:36.045Z",
  "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
  "SourceIdentifier": "sample-cluster2",
  "SourceType": "db-instance"
},
{
  "EventCategories": [
    "availability"
  ],
  "Message": "DB instance restarted",
  "Date": "2019-03-19T18:23:46.209Z",
  "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
  "SourceIdentifier": "sample-cluster2",
  "SourceType": "db-instance"
},
{
  "Date": "2019-03-19T18:39:05.822Z",
  "EventCategories": [
```

```

        "configuration change"
      ],
      "Message": "Updated parameter ttl_monitor to enabled with apply method
immediate",
      "SourceIdentifier": "custom3-6-param-grp",
      "SourceType": "db-parameter-group"
    },
    {
      "Date": "2019-03-19T18:39:48.067Z",
      "EventCategories": [
        "configuration change"
      ],
      "Message": "Updated parameter audit_logs to disabled with apply method
immediate",
      "SourceIdentifier": "custom3-6-param-grp",
      "SourceType": "db-parameter-group"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat Acara Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## describe-orderable-db-instance-options

Contoh kode berikut menunjukkan cara menggunakan `describe-orderable-db-instance-options`.

### AWS CLI

Untuk menemukan opsi instans Amazon DocumentDB, Anda dapat memesan

`describe-orderable-db-instance-options` Contoh berikut mencantumkan semua opsi instance untuk Amazon DocumentDB untuk suatu wilayah.

```

aws docdb describe-orderable-db-instance-options \
  --engine docdb \
  --region us-east-1

```

Output:

```
{
  "OrderableDBInstanceOptions": [
    {
      "Vpc": true,
      "AvailabilityZones": [
        {
          "Name": "us-east-1a"
        },
        {
          "Name": "us-east-1b"
        },
        {
          "Name": "us-east-1c"
        },
        {
          "Name": "us-east-1d"
        }
      ],
      "EngineVersion": "3.6.0",
      "DBInstanceClass": "db.r4.16xlarge",
      "LicenseModel": "na",
      "Engine": "docdb"
    },
    {
      "Vpc": true,
      "AvailabilityZones": [
        {
          "Name": "us-east-1a"
        },
        {
          "Name": "us-east-1b"
        },
        {
          "Name": "us-east-1c"
        },
        {
          "Name": "us-east-1d"
        }
      ],
      "EngineVersion": "3.6.0",
      "DBInstanceClass": "db.r4.2xlarge",
      "LicenseModel": "na",
```

```
    "Engine": "docdb"
  },
  {
    "Vpc": true,
    "AvailabilityZones": [
      {
        "Name": "us-east-1a"
      },
      {
        "Name": "us-east-1b"
      },
      {
        "Name": "us-east-1c"
      },
      {
        "Name": "us-east-1d"
      }
    ],
    "EngineVersion": "3.6.0",
    "DBInstanceClass": "db.r4.4xlarge",
    "LicenseModel": "na",
    "Engine": "docdb"
  },
  {
    "Vpc": true,
    "AvailabilityZones": [
      {
        "Name": "us-east-1a"
      },
      {
        "Name": "us-east-1b"
      },
      {
        "Name": "us-east-1c"
      },
      {
        "Name": "us-east-1d"
      }
    ],
    "EngineVersion": "3.6.0",
    "DBInstanceClass": "db.r4.8xlarge",
    "LicenseModel": "na",
    "Engine": "docdb"
  },
}
```

```
{
  "Vpc": true,
  "AvailabilityZones": [
    {
      "Name": "us-east-1a"
    },
    {
      "Name": "us-east-1b"
    },
    {
      "Name": "us-east-1c"
    },
    {
      "Name": "us-east-1d"
    }
  ],
  "EngineVersion": "3.6.0",
  "DBInstanceClass": "db.r4.large",
  "LicenseModel": "na",
  "Engine": "docdb"
},
{
  "Vpc": true,
  "AvailabilityZones": [
    {
      "Name": "us-east-1a"
    },
    {
      "Name": "us-east-1b"
    },
    {
      "Name": "us-east-1c"
    },
    {
      "Name": "us-east-1d"
    }
  ],
  "EngineVersion": "3.6.0",
  "DBInstanceClass": "db.r4.xlarge",
  "LicenseModel": "na",
  "Engine": "docdb"
}
]
```

```
}
```

Untuk informasi selengkapnya, lihat [Menambahkan Instans Amazon DocumentDB ke Cluster](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribeOrderableDbInstanceOptions](#) di Referensi AWS CLI Perintah.

## **describe-pending-maintenance-actions**

Contoh kode berikut menunjukkan cara menggunakan `describe-pending-maintenance-actions`.

### AWS CLI

Untuk membuat daftar tindakan pemeliharaan Amazon DocumentDB Anda yang tertunda

`describe-pending-maintenance-actions` Contoh berikut mencantumkan semua tindakan pemeliharaan Amazon DocumentDB Anda yang tertunda.

```
aws docdb describe-pending-maintenance-actions
```

Output:

```
{
  "PendingMaintenanceActions": []
}
```

Untuk informasi selengkapnya, lihat [Memelihara Amazon DocumentDB di](#) Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [DescribePendingMaintenanceActions](#) di Referensi AWS CLI Perintah.

## **failover-db-cluster**

Contoh kode berikut menunjukkan cara menggunakan `failover-db-cluster`.

### AWS CLI

Untuk memaksa cluster Amazon DocumentDB melakukan failover ke replika

`failover-db-cluster` Contoh berikut menyebabkan instance utama di cluster sampel klaster Amazon DocumentDB mengalami failover ke replika.

```
aws docdb failover-db-cluster \  
--db-cluster-identifier sample-cluster
```

### Output:

```
{  
  "DBCluster": {  
    "AssociatedRoles": [],  
    "DBClusterIdentifier": "sample-cluster",  
    "EngineVersion": "3.6.0",  
    "DBSubnetGroup": "default",  
    "MasterUsername": "master-user",  
    "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",  
    "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2c",  
      "us-west-2b"  
    ],  
    "LatestRestorableTime": "2019-03-18T21:35:23.548Z",  
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",  
    "PreferredBackupWindow": "00:00-00:30",  
    "Port": 27017,  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-77186e0d",  
        "Status": "active"  
      }  
    ],  
    "StorageEncrypted": false,  
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z",  
    "MultiAZ": true,  
    "Status": "available",  
    "DBClusterMembers": [  
      {  
        "DBClusterParameterGroupStatus": "in-sync",  
        "IsClusterWriter": false,  
        "DBInstanceIdentifier": "sample-cluster",  
        "PromotionTier": 1  
      },  
      {  
        "DBClusterParameterGroupStatus": "in-sync",
```



```

        "IsClusterWriter": true,
        "DBInstanceIdentifier": "sample-cluster2",
        "PromotionTier": 2
    }
],
"EnabledCloudwatchLogsExports": [
    "audit"
],
"DBClusterParameterGroup": "default.docdb3.6",
"HostedZoneId": "ZNKXH85TT8WW",
"DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
"BackupRetentionPeriod": 3,
"DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
"ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
"Engine": "docdb"
}
}

```

Untuk informasi selengkapnya, lihat [Amazon DocumentDB Failover di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [FailoverDbCluster](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan semua tag pada sumber daya Amazon DocumentDB

`list-tags-for-resource` Contoh berikut mencantumkan semua tag di cluster Amazon `sample-cluster` DocumentDB.

```
aws docdb list-tags-for-resource \
  --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster
```

Output:

```
{
  "TagList": [
    {
```

```
    "Key": "A",
    "Value": "ALPHA"
  },
  {
    "Key": "B",
    "Value": ""
  },
  {
    "Key": "C",
    "Value": "CHARLIE"
  }
]
```

Untuk informasi selengkapnya, lihat [Daftar Tag di Sumber Daya Amazon DocumentDB di Panduan](#) Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## modify-db-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster-parameter-group`.

### AWS CLI

Untuk memodifikasi grup parameter cluster Amazon DocumentDB DB

`modify-db-cluster-parameter-group` Contoh berikut memodifikasi grup `custom3-6-param-grp` parameter cluster Amazon DocumentDB dengan mengatur `audit_logs` dua `ttl_monitor` parameter dan diaktifkan. Perubahan diterapkan pada reboot berikutnya.

```
aws docdb modify-db-cluster-parameter-group \
  --db-cluster-parameter-group-name custom3-6-param-grp \
  --parameters
  ParameterName=audit_logs,ParameterValue=enabled,ApplyMethod=pending-reboot \
  ParameterName=ttl_monitor,ParameterValue=enabled,ApplyMethod=pending-reboot
```

Output:

```
{
  "DBClusterParameterGroupName": "custom3-6-param-grp"
```

```
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Grup Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [ModifyDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## modify-db-cluster-snapshot-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster-snapshot-attribute`.

### AWS CLI

Contoh 1: Untuk menambahkan atribut ke snapshot Amazon DocumentDB

`modify-db-cluster-snapshot-attribute` Contoh berikut menambahkan empat nilai atribut ke snapshot cluster Amazon DocumentDB.

```
aws docdb modify-db-cluster-snapshot-attribute \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot \  
  --attribute-name restore \  
  --values-to-add all 123456789011 123456789012 123456789013
```

Output:

```
{  
  "DBClusterSnapshotAttributesResult": {  
    "DBClusterSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "all",  
          "123456789011",  
          "123456789012",  
          "123456789013"  
        ]  
      }  
    ],  
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"  
  }  
}
```

## Contoh 2: Untuk menghapus atribut dari snapshot Amazon DocumentDB

`modify-db-cluster-snapshot-attribute` Contoh berikut menghapus dua nilai atribut dari snapshot cluster Amazon DocumentDB.

```
aws docdb modify-db-cluster-snapshot-attribute \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot \  
  --attribute-name restore \  
  --values-to-remove 123456789012 all
```

### Output:

```
{  
  "DBClusterSnapshotAttributesResult": {  
    "DBClusterSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "123456789011",  
          "123456789013"  
        ]  
      }  
    ],  
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"  
  }  
}
```

Untuk informasi selengkapnya, lihat [ModifyDB ClusterSnapshotAttribute](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [ModifyDbClusterSnapshotAttribute](#) di Referensi AWS CLI Perintah.

## **modify-db-cluster**

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster`.

### AWS CLI

Untuk memodifikasi cluster Amazon DocumentDB

`modify-db-cluster` Contoh berikut memodifikasi klaster Amazon `sample-cluster` DocumentDB dengan membuat periode retensi untuk pencadangan otomatis 7 hari, dan

mengubah jendela pilihan untuk pencadangan dan pemeliharaan. Semua perubahan diterapkan di jendela pemeliharaan berikutnya.

```
aws docdb modify-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --no-apply-immediately \  
  --backup-retention-period 7 \  
  --preferred-backup-window 18:00-18:30 \  
  --preferred-maintenance-window sun:20:00-sun:20:30
```

Output:

```
{  
  "DBCluster": {  
    "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "DBClusterMembers": [  
      {  
        "DBClusterParameterGroupStatus": "in-sync",  
        "DBInstanceIdentifier": "sample-cluster",  
        "IsClusterWriter": true,  
        "PromotionTier": 1  
      },  
      {  
        "DBClusterParameterGroupStatus": "in-sync",  
        "DBInstanceIdentifier": "sample-cluster2",  
        "IsClusterWriter": false,  
        "PromotionTier": 2  
      }  
    ],  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "StorageEncrypted": false,  
    "PreferredBackupWindow": "18:00-18:30",  
    "MultiAZ": true,  
    "EngineVersion": "3.6.0",  
    "MasterUsername": "master-user",  
    "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "DBSubnetGroup": "default",  
    "LatestRestorableTime": "2019-03-18T22:08:13.408Z",  
    "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",  
    "PreferredMaintenanceWindow": "sun:20:00-sun:20:30",  
    "AssociatedRoles": [],
```

```

    "EnabledCloudwatchLogsExports": [
      "audit"
    ],
    "Engine": "docdb",
    "DBClusterParameterGroup": "default.docdb3.6",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
    "BackupRetentionPeriod": 7,
    "DBClusterIdentifier": "sample-cluster",
    "AvailabilityZones": [
      "us-west-2a",
      "us-west-2c",
      "us-west-2b"
    ],
    "Status": "available",
    "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-77186e0d",
        "Status": "active"
      }
    ],
    "Port": 27017
  }
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Cluster Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [ModifyDbCluster](#) di Referensi AWS CLI Perintah.

## modify-db-instance

Contoh kode berikut menunjukkan cara menggunakan `modify-db-instance`.

### AWS CLI

Untuk memodifikasi instance Amazon DocumentDB

`modify-db-instance` Contoh berikut memodifikasi instance Amazon `sample-cluster2` DocumentDB dengan mengubah `db.r4.4xlarge` kelas instance-nya menjadi dan tingkat promosinya menjadi. 5 Perubahan diterapkan segera tetapi hanya dapat dilihat setelah status instance tersedia.

```
aws docdb modify-db-instance \  
  --db-instance-identifier sample-cluster2 \  
  --apply-immediately \  
  --db-instance-class db.r4.4xlarge \  
  --promotion-tier 5
```

**Output:**

```
{  
  "DBInstance": {  
    "EngineVersion": "3.6.0",  
    "StorageEncrypted": false,  
    "DBInstanceClass": "db.r4.large",  
    "PreferredMaintenanceWindow": "mon:08:39-mon:09:09",  
    "AutoMinorVersionUpgrade": true,  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-77186e0d",  
        "Status": "active"  
      }  
    ],  
    "PreferredBackupWindow": "18:00-18:30",  
    "EnabledCloudwatchLogsExports": [  
      "audit"  
    ],  
    "AvailabilityZone": "us-west-2f",  
    "DBInstanceIdentifier": "sample-cluster2",  
    "InstanceCreateTime": "2019-03-15T20:36:06.338Z",  
    "Engine": "docdb",  
    "BackupRetentionPeriod": 7,  
    "DBSubnetGroup": {  
      "DBSubnetGroupName": "default",  
      "DBSubnetGroupDescription": "default",  
      "SubnetGroupStatus": "Complete",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-4e26d263",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2a"  
          }  
        },  
        {  
          "SubnetStatus": "Active"  
        }  
      ]  
    }  
  }  
}
```

```

        "SubnetIdentifier": "subnet-afc329f4",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2c"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-53ab3636",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2d"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-991cb8d0",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2b"
        },
        "SubnetStatus": "Active"
    }
],
    "VpcId": "vpc-91280df6"
},
    "PromotionTier": 2,
    "Endpoint": {
        "Address": "sample-cluster2.corcjozrlsfc.us-west-2.docdb.amazonaws.com",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "Port": 27017
    },
    "DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA",
    "DBClusterIdentifier": "sample-cluster",
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "PendingModifiedValues": {
        "DBInstanceClass": "db.r4.4xlarge"
    },
    "PubliclyAccessible": false,
    "DBInstanceStatus": "available"
}
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Instans Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [ModifyDbInstance](#) di Referensi AWS CLI Perintah.



## modify-db-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `modify-db-subnet-group`.

### AWS CLI

Untuk memodifikasi grup subnet Amazon DocumentDB

`modify-db-subnet-group` Contoh berikut memodifikasi grup subnet `sample-subnet-group` dengan menambahkan subnet yang ditentukan dan deskripsi baru.

```
aws docdb modify-db-subnet-group \  
  --db-subnet-group-name sample-subnet-group \  
  --subnet-ids subnet-b3806e8f subnet-53ab3636 subnet-991cb8d0 \  
  --db-subnet-group-description "New subnet description"
```

Output:

```
{  
  "DBSubnetGroup": {  
    "DBSubnetGroupName": "sample-subnet-group",  
    "SubnetGroupStatus": "Complete",  
    "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:sample-  
subnet-group",  
    "VpcId": "vpc-91280df6",  
    "DBSubnetGroupDescription": "New subnet description",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-b3806e8f",  
        "SubnetStatus": "Active",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2a"  
        }  
      },  
      {  
        "SubnetIdentifier": "subnet-53ab3636",  
        "SubnetStatus": "Active",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2c"  
        }  
      },  
      {  
        "SubnetIdentifier": "subnet-991cb8d0",
```

```

        "SubnetStatus": "Active",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2b"
        }
    ]
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Grup Subnet Amazon DocumentDB di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [ModifyDbSubnetGroup](#) di Referensi AWS CLI Perintah.

## reboot-db-instance

Contoh kode berikut menunjukkan cara menggunakan `reboot-db-instance`.

### AWS CLI

Untuk me-reboot instance Amazon DocumentDB

`reboot-db-instance` Contoh berikut me-reboot instance Amazon DocumentDB. `sample-cluster2`

```
aws docdb reboot-db-instance \
  --db-instance-identifier sample-cluster2
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "DBInstance": {
    "PreferredBackupWindow": "18:00-18:30",
    "DBInstanceIdentifier": "sample-cluster2",
    "VpcSecurityGroups": [
      {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
      }
    ],
    "DBSubnetGroup": {
      "VpcId": "vpc-91280df6",
      "Subnets": [

```

```
    {
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2a"
      },
      "SubnetIdentifier": "subnet-4e26d263"
    },
    {
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2c"
      },
      "SubnetIdentifier": "subnet-afc329f4"
    },
    {
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2d"
      },
      "SubnetIdentifier": "subnet-53ab3636"
    },
    {
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      },
      "SubnetIdentifier": "subnet-991cb8d0"
    }
  ],
  "SubnetGroupStatus": "Complete",
  "DBSubnetGroupName": "default",
  "DBSubnetGroupDescription": "default"
},
"PendingModifiedValues": {},
"Endpoint": {
  "Address": "sample-cluster2.corcjozrlsfc.us-west-2.docdb.amazonaws.com",
  "HostedZoneId": "ZNKXH85TT8WVW",
  "Port": 27017
},
"EnabledCloudwatchLogsExports": [
  "audit"
],
"StorageEncrypted": false,
"DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA",
```

```
"AutoMinorVersionUpgrade": true,  
"Engine": "docdb",  
"InstanceCreateTime": "2019-03-15T20:36:06.338Z",  
"EngineVersion": "3.6.0",  
"PromotionTier": 5,  
"BackupRetentionPeriod": 7,  
"DBClusterIdentifier": "sample-cluster",  
"PreferredMaintenanceWindow": "mon:08:39-mon:09:09",  
"PubliclyAccessible": false,  
"DBInstanceClass": "db.r4.4xlarge",  
"AvailabilityZone": "us-west-2d",  
"DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",  
"DBInstanceStatus": "rebooting"  
}  
}
```

Untuk informasi selengkapnya, lihat [Mem-boot ulang Amazon DocumentDB Instance di Panduan Pengembang Amazon DocumentDB](#).

- Untuk detail API, lihat [RebootDbInstance](#) di Referensi AWS CLI Perintah.

## **remove-tags-from-resource**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya Amazon DocumentDB

`remove-tags-from-resource` Contoh berikut menghapus tag dengan kunci bernama B dari cluster Amazon `sample-cluster` DocumentDB.

```
aws docdb remove-tags-from-resource \  
  --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \  
  --tag-keys B
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Tag dari Amazon DocumentDB resource](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [RemoveTagsFromResource](#) di Referensi AWS CLI Perintah.

## reset-db-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `reset-db-cluster-parameter-group`.

### AWS CLI

Untuk mengatur ulang nilai parameter yang ditentukan ke defaultnya di grup parameter Amazon DocumentDB

`reset-db-cluster-parameter-group` Contoh berikut me-reset parameter `ttl_monitor` dalam grup `custom3-6-param-grp` parameter Amazon DocumentDB ke nilai defaultnya.

```
aws docdb reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name custom3-6-param-grp \  
  --parameters ParameterName=ttl_monitor,ApplyMethod=immediate
```

Output:

```
{  
  "DBClusterParameterGroupName": "custom3-6-param-grp"  
}
```

Untuk informasi selengkapnya, lihat judul di Panduan Pengembang Amazon DocumentDB.

Untuk mengatur ulang nilai parameter yang ditentukan atau semua ke defaultnya di grup parameter Amazon DocumentDB

`reset-db-cluster-parameter-group` Contoh berikut me-reset semua parameter dalam grup `custom3-6-param-grp` parameter Amazon DocumentDB ke nilai defaultnya.

```
aws docdb reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name custom3-6-param-grp \  
  --reset-all-parameters
```

Output:

```
{  
  "DBClusterParameterGroupName": "custom3-6-param-grp"  
}
```

Untuk informasi selengkapnya, lihat [Menyetel ulang Grup Parameter Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [ResetDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## restore-db-cluster-from-snapshot

Contoh kode berikut menunjukkan cara menggunakan `restore-db-cluster-from-snapshot`.

### AWS CLI

Untuk memulihkan cluster Amazon DocumentDB dari snapshot otomatis atau manual

`restore-db-cluster-from-snapshot` Contoh berikut membuat cluster Amazon DocumentDB baru `sample-cluster-2019-03-16-00-01-restored` bernama dari snapshot. `rds:sample-cluster-2019-03-16-00-01`

```
aws docdb restore-db-cluster-from-snapshot \  
  --db-cluster-identifier sample-cluster-2019-03-16-00-01-restored \  
  --engine docdb \  
  --snapshot-identifier rds:sample-cluster-2019-03-16-00-01
```

### Output:

```
{  
  "DBCluster": {  
    "ClusterCreateTime": "2019-03-19T18:45:01.857Z",  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "Engine": "docdb",  
    "DBClusterMembers": [],  
    "MultiAZ": false,  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2c",  
      "us-west-2b"  
    ],  
    "StorageEncrypted": false,  
    "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-  
corcjozrlsfc.us-west-2.docdb.amazonaws.com",  
    "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-  
corcjozrlsfc.us-west-2.docdb.amazonaws.com",  
    "Port": 27017,  
    "PreferredBackupWindow": "00:00-00:30",
```

```

    "DBSubnetGroup": "default",
    "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
    "DBClusterParameterGroup": "default.docdb3.6",
    "DbClusterResourceId": "cluster-X0046Q3RH4LWSYNH3NMZKXPISU",
    "MasterUsername": "master-user",
    "EngineVersion": "3.6.0",
    "BackupRetentionPeriod": 3,
    "AssociatedRoles": [],
    "Status": "creating",
    "VpcSecurityGroups": [
      {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Memulihkan dari Snapshot Cluster](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [RestoreDbClusterFromSnapshot](#) di Referensi AWS CLI Perintah.

## restore-db-cluster-to-point-in-time

Contoh kode berikut menunjukkan cara menggunakan `restore-db-cluster-to-point-in-time`.

### AWS CLI

Untuk memulihkan cluster Amazon DocumentDB ke snapshot point-in-time manual

`restore-db-cluster-to-point-in-time` Contoh berikut menggunakan `sample-cluster-snapshot` untuk membuat cluster Amazon DocumentDB baru `sample-cluster-pit`, menggunakan waktu restorable terbaru.

```

aws docdb restore-db-cluster-to-point-in-time \
  --db-cluster-identifier sample-cluster-pit \
  --source-db-cluster-identifier arn:aws:rds:us-
west-2:123456789012:cluster:sample-cluster \

```

```
--use-latest-restorable-time
```

## Output:

```
{
  "DBCluster": {
    "StorageEncrypted": false,
    "BackupRetentionPeriod": 3,
    "MasterUsername": "master-user",
    "HostedZoneId": "ZNKXH85TT8WVW",
    "PreferredBackupWindow": "00:00-00:30",
    "MultiAZ": false,
    "DBClusterIdentifier": "sample-cluster-pit",
    "DBSubnetGroup": "default",
    "ClusterCreateTime": "2019-04-03T15:55:21.320Z",
    "AssociatedRoles": [],
    "DBClusterParameterGroup": "default.docdb3.6",
    "DBClusterMembers": [],
    "Status": "creating",
    "AvailabilityZones": [
      "us-west-2a",
      "us-west-2d",
      "us-west-2b"
    ],
    "ReaderEndpoint": "sample-cluster-pit.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
    "Port": 27017,
    "Engine": "docdb",
    "EngineVersion": "3.6.0",
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-77186e0d",
        "Status": "active"
      }
    ],
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
    "Endpoint": "sample-cluster-pit.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
    "DbClusterResourceId": "cluster-NLCABBX0SE2QPQ4GOLZIFWEPLM",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster-
pit"
  }
}
```



Untuk informasi selengkapnya, lihat [Memulihkan Snapshot ke Titik Waktu di Panduan Pengembang](#) Amazon DocumentDB.

- Untuk detail API, lihat [RestoreDbClusterToPointInTime](#) di Referensi AWS CLI Perintah.

## start-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `start-db-cluster`.

### AWS CLI

Untuk memulai cluster Amazon DocumentDB yang dihentikan

`start-db-cluster` Contoh berikut memulai cluster Amazon DocumentDB yang ditentukan.

```
aws docdb start-db-cluster \  
--db-cluster-identifier sample-cluster
```

Output:

```
{  
  "DBCluster": {  
    "ClusterCreateTime": "2019-03-19T18:45:01.857Z",  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "Engine": "docdb",  
    "DBClusterMembers": [],  
    "MultiAZ": false,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1c",  
      "us-east-1f"  
    ],  
    "StorageEncrypted": false,  
    "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-  
corcjzrlsfc.us-east-1.docdb.amazonaws.com",  
    "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-  
corcjzrlsfc.us-east-1.docdb.amazonaws.com",  
    "Port": 27017,  
    "PreferredBackupWindow": "00:00-00:30",  
    "DBSubnetGroup": "default",  
    "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",  
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
```

```

    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
    "DBClusterParameterGroup": "default.docdb3.6",
    "DbClusterResourceId": "cluster-X0046Q3RH4LWSYNH3NMZKXPISU",
    "MasterUsername": "master-user",
    "EngineVersion": "3.6.0",
    "BackupRetentionPeriod": 3,
    "AssociatedRoles": [],
    "Status": "creating",
    "VpcSecurityGroups": [
      {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Menghentikan dan Memulai Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [StartDbCluster](#) di Referensi AWS CLI Perintah.

## stop-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `stop-db-cluster`.

### AWS CLI

Untuk menghentikan klaster Amazon DocumentDB yang sedang berjalan

`stop-db-cluster` Contoh berikut menghentikan cluster Amazon DocumentDB yang ditentukan.

```

aws docdb stop-db-cluster \
  --db-cluster-identifier sample-cluster

```

### Output:

```

{
  "DBCluster": {
    "ClusterCreateTime": "2019-03-19T18:45:01.857Z",
    "HostedZoneId": "ZNXH85TT8WVW",

```

```
"Engine": "docdb",
"DBClusterMembers": [],
"MultiAZ": false,
"AvailabilityZones": [
  "us-east-1a",
  "us-east-1c",
  "us-east-1f"
],
"StorageEncrypted": false,
"ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-
corcjzrlsfc.us-east-1.docdb.amazonaws.com",
"Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-
corcjzrlsfc.us-east-1.docdb.amazonaws.com",
"Port": 27017,
"PreferredBackupWindow": "00:00-00:30",
"DBSubnetGroup": "default",
"DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",
"PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
"DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
"DBClusterParameterGroup": "default.docdb3.6",
"DbClusterResourceId": "cluster-X0046Q3RH4LWSYNH3NMZKXPISU",
"MasterUsername": "master-user",
"EngineVersion": "3.6.0",
"BackupRetentionPeriod": 3,
"AssociatedRoles": [],
"Status": "creating",
"VpcSecurityGroups": [
  {
    "Status": "active",
    "VpcSecurityGroupId": "sg-77186e0d"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Menghentikan dan Memulai Cluster Amazon DocumentDB](#) di Panduan Pengembang Amazon DocumentDB.

- Untuk detail API, lihat [StopDbCluster](#) di Referensi AWS CLI Perintah.

## Contoh DynamoDB menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with DynamoDB.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-get-item**

Contoh kode berikut menunjukkan cara menggunakan `batch-get-item`.

AWS CLI

Untuk mengambil beberapa item dari tabel

`batch-get-items` Contoh berikut membaca beberapa item dari `MusicCollection` tabel menggunakan batch tiga `GetItem` permintaan, dan meminta jumlah unit kapasitas baca yang dikonsumsi oleh operasi. Perintah hanya mengembalikan `AlbumTitle` atribut.

```
aws dynamodb batch-get-item \  
  --request-items file://request-items.json \  
  --return-consumed-capacity TOTAL
```

Isi dari `request-items.json`:

```
{  
  "MusicCollection": {
```

```

    "Keys": [
      {
        "Artist": {"S": "No One You Know"},
        "SongTitle": {"S": "Call Me Today"}
      },
      {
        "Artist": {"S": "Acme Band"},
        "SongTitle": {"S": "Happy Day"}
      },
      {
        "Artist": {"S": "No One You Know"},
        "SongTitle": {"S": "Scared of My Shadow"}
      }
    ],
    "ProjectionExpression": "AlbumTitle"
  }
}

```

### Output:

```

{
  "Responses": {
    "MusicCollection": [
      {
        "AlbumTitle": {
          "S": "Somewhat Famous"
        }
      },
      {
        "AlbumTitle": {
          "S": "Blue Sky Blues"
        }
      },
      {
        "AlbumTitle": {
          "S": "Louder Than Ever"
        }
      }
    ]
  },
  "UnprocessedKeys": {},
  "ConsumedCapacity": [
    {

```

```

        "TableName": "MusicCollection",
        "CapacityUnits": 1.5
      }
    ]
  }

```

Untuk informasi selengkapnya, lihat [Operasi Batch](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [BatchGetItem](#) di Referensi AWS CLI Perintah.

## batch-write-item

Contoh kode berikut menunjukkan cara menggunakan `batch-write-item`.

### AWS CLI

Untuk menambahkan beberapa item ke tabel

`batch-write-item` Contoh berikut menambahkan tiga item baru ke `MusicCollection` tabel menggunakan batch tiga `PutItem` permintaan. Ini juga meminta informasi tentang jumlah unit kapasitas tulis yang dikonsumsi oleh operasi dan koleksi item apa pun yang dimodifikasi oleh operasi.

```

aws dynamodb batch-write-item \
  --request-items file://request-items.json \
  --return-consumed-capacity INDEXES \
  --return-item-collection-metrics SIZE

```

Isi dari `request-items.json`:

```

{
  "MusicCollection": [
    {
      "PutRequest": {
        "Item": {
          "Artist": {"S": "No One You Know"},
          "SongTitle": {"S": "Call Me Today"},
          "AlbumTitle": {"S": "Somewhat Famous"}
        }
      }
    }
  ],

```

```

    {
      "PutRequest": {
        "Item": {
          "Artist": {"S": "Acme Band"},
          "SongTitle": {"S": "Happy Day"},
          "AlbumTitle": {"S": "Songs About Life"}
        }
      }
    },
    {
      "PutRequest": {
        "Item": {
          "Artist": {"S": "No One You Know"},
          "SongTitle": {"S": "Scared of My Shadow"},
          "AlbumTitle": {"S": "Blue Sky Blues"}
        }
      }
    }
  ]
}

```

### Output:

```

{
  "UnprocessedItems": {},
  "ItemCollectionMetrics": {
    "MusicCollection": [
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "No One You Know"
          }
        }
      },
      {
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      }
    ],
    {
      "ItemCollectionKey": {
        "Artist": {
          "S": "Acme Band"
        }
      }
    }
  ]
}

```

```

        },
        "SizeEstimateRangeGB": [
            0.0,
            1.0
        ]
    }
]
},
"ConsumedCapacity": [
    {
        "TableName": "MusicCollection",
        "CapacityUnits": 6.0,
        "Table": {
            "CapacityUnits": 3.0
        },
        "LocalSecondaryIndexes": {
            "AlbumTitleIndex": {
                "CapacityUnits": 3.0
            }
        }
    }
]
}

```

Untuk informasi selengkapnya, lihat [Operasi Batch](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [BatchWriteItem](#) di Referensi AWS CLI Perintah.

## create-backup

Contoh kode berikut menunjukkan cara menggunakan `create-backup`.

### AWS CLI

Untuk membuat cadangan untuk tabel DynamoDB yang ada

`create-backup` Contoh berikut membuat cadangan dari `MusicCollection` tabel.

```

aws dynamodb create-backup \
  --table-name MusicCollection \
  --backup-name MusicCollectionBackup

```



**Output:**

```
{
  "BackupDetails": {
    "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
backup/01576616366715-b4e58d3a",
    "BackupName": "MusicCollectionBackup",
    "BackupSizeBytes": 0,
    "BackupStatus": "CREATING",
    "BackupType": "USER",
    "BackupCreationDateTime": 1576616366.715
  }
}
```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [CreateBackup](#) di Referensi AWS CLI Perintah.

**create-global-table**

Contoh kode berikut menunjukkan cara menggunakan `create-global-table`.

**AWS CLI**

Untuk membuat tabel global

`create-global-table` Contoh berikut membuat tabel global dari dua tabel identik di AWS Regions terpisah yang ditentukan.

```
aws dynamodb create-global-table \
  --global-table-name MusicCollection \
  --replication-group RegionName=us-east-2 RegionName=us-east-1 \
  --region us-east-2
```

**Output:**

```
{
  "GlobalTableDescription": {
    "ReplicationGroup": [
      {
```

```

        "RegionName": "us-east-2"
    },
    {
        "RegionName": "us-east-1"
    }
],
"GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/
MusicCollection",
"CreationDateTime": 1576625818.532,
"GlobalTableStatus": "CREATING",
"GlobalTableName": "MusicCollection"
}
}

```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [CreateGlobalTable](#) di Referensi AWS CLI Perintah.

## create-table

Contoh kode berikut menunjukkan cara menggunakan `create-table`.

### AWS CLI

Contoh 1: Untuk membuat tabel dengan tag

`create-table` Contoh berikut menggunakan atribut tertentu dan skema kunci untuk membuat tabel bernama `MusicCollection`. Tabel ini menggunakan throughput yang disediakan dan dienkripsi saat istirahat menggunakan CMK yang dimiliki default. AWS Perintah ini juga menerapkan tag ke tabel, dengan kunci dari `Owner` dan nilai `blueTeam`.

```

aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-definitions AttributeName=Artist,AttributeType=S
  AttributeName=SongTitle,AttributeType=S \
  --key-schema AttributeName=Artist,KeyType=HASH
  AttributeName=SongTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
  --tags Key=Owner,Value=blueTeam

```

Output:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "WriteCapacityUnits": 5,
      "ReadCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "TableName": "MusicCollection",
    "TableStatus": "CREATING",
    "KeySchema": [
      {
        "KeyType": "HASH",
        "AttributeName": "Artist"
      },
      {
        "KeyType": "RANGE",
        "AttributeName": "SongTitle"
      }
    ],
    "ItemCount": 0,
    "CreationDateTime": "2020-05-26T16:04:41.627000-07:00",
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 2: Untuk membuat tabel dalam Mode On-Demand

Contoh berikut membuat tabel yang disebut `MusicCollection` menggunakan mode on-demand, bukan mode throughput yang disediakan. Ini berguna untuk tabel dengan beban kerja yang tidak terduga.

```
aws dynamodb create-table \  
  --table-name MusicCollection \  
  --attribute-definitions AttributeName=Artist,AttributeType=S  
  AttributeName=SongTitle,AttributeType=S \  
  --key-schema AttributeName=Artist,KeyType=HASH  
  AttributeName=SongTitle,KeyType=RANGE \  
  --billing-mode PAY_PER_REQUEST
```

Output:

```
{  
  "TableDescription": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "Artist",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "AttributeType": "S"  
      }  
    ],  
    "TableName": "MusicCollection",  
    "KeySchema": [  
      {  
        "AttributeName": "Artist",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "TableStatus": "CREATING",  
    "CreationDateTime": "2020-05-27T11:44:10.807000-07:00",  
    "ProvisionedThroughput": {  
      "NumberOfDecreasesToday": 0,  
      "ReadCapacityUnits": 0,  
      "WriteCapacityUnits": 0  
    }  
  }  
}
```

```

    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "BillingModeSummary": {
      "BillingMode": "PAY_PER_REQUEST"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 3: Untuk membuat tabel dan mengenkripsi dengan Customer Managed CMK

Contoh berikut membuat tabel bernama MusicCollection dan mengenkripsi menggunakan CMK yang dikelola pelanggan.

```

aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-definitions AttributeName=Artist,AttributeType=S \
  AttributeName=SongTitle,AttributeType=S \
  --key-schema AttributeName=Artist,KeyType=HASH \
  AttributeName=SongTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
  --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=abcd1234-abcd-1234-
a123-ab1234a1b234

```

Output:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],

```

```

    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": "2020-05-27T11:12:16.431000-07:00",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 5,
      "WriteCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "SSEDescription": {
      "Status": "ENABLED",
      "SSEType": "KMS",
      "KMSMasterKeyArn": "arn:aws:kms:us-west-2:123456789012:key/abcd1234-
abcd-1234-a123-ab1234a1b234"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 4: Untuk membuat tabel dengan Indeks Sekunder Lokal

Contoh berikut menggunakan atribut tertentu dan skema kunci untuk membuat tabel bernama `MusicCollection` dengan Indeks Sekunder Lokal bernama `AlbumTitleIndex`.

```

aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-definitions AttributeName=Artist,AttributeType=S
  AttributeName=SongTitle,AttributeType=S AttributeName=AlbumTitle,AttributeType=S \

```

```

--key-schema AttributeName=Artist,KeyType=HASH
AttributeName=SongTitle,KeyType=RANGE \
--provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
--local-secondary-indexes \
  "[
    {
      \"IndexName\": \"AlbumTitleIndex\",
      \"KeySchema\": [
        {\"AttributeName\": \"Artist\", \"KeyType\": \"HASH\"},
        {\"AttributeName\": \"AlbumTitle\", \"KeyType\": \"RANGE\"}
      ],
      \"Projection\": {
        \"ProjectionType\": \"INCLUDE\",
        \"NonKeyAttributes\": [\"Genre\", \"Year\"]
      }
    }
  ]"

```

## Output:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {

```

```

        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
    }
],
"TableStatus": "CREATING",
"CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
"ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
},
"TableSizeBytes": 0,
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
"TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"LocalSecondaryIndexes": [
    {
        "IndexName": "AlbumTitleIndex",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "AlbumTitle",
                "KeyType": "RANGE"
            }
        ],
        "Projection": {
            "ProjectionType": "INCLUDE",
            "NonKeyAttributes": [
                "Genre",
                "Year"
            ]
        },
        "IndexSizeBytes": 0,
        "ItemCount": 0,
        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
    }
]
}
}

```



Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 5: Untuk membuat tabel dengan Indeks Sekunder Global

Contoh berikut membuat tabel bernama `GameScores` dengan Global Secondary Index disebut `GameTitleIndex`. Tabel dasar memiliki kunci partisi `UserId` dan kunci urutan `GameTitle`, sehingga Anda dapat menemukan skor terbaik pengguna individu untuk game tertentu secara efisien, sedangkan GSI memiliki kunci partisi `GameTitle` dan kunci urutan `TopScore`, memungkinkan Anda untuk cepat menemukan skor tertinggi secara keseluruhan untuk game tertentu.

```
aws dynamodb create-table \
  --table-name GameScores \
  --attribute-definitions AttributeName=UserId,AttributeType=S
  AttributeName=GameTitle,AttributeType=S AttributeName=TopScore,AttributeType=N \
  --key-schema AttributeName=UserId,KeyType=HASH \
    AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --global-secondary-indexes \
    "[
      {
        \"IndexName\": \"GameTitleIndex\",
        \"KeySchema\": [
          {\"AttributeName\": \"GameTitle\", \"KeyType\": \"HASH\"},
          {\"AttributeName\": \"TopScore\", \"KeyType\": \"RANGE\"}
        ],
        \"Projection\": {
          \"ProjectionType\": \"INCLUDE\",
          \"NonKeyAttributes\": [\"UserId\"]
        },
        \"ProvisionedThroughput\": {
          \"ReadCapacityUnits\": 10,
          \"WriteCapacityUnits\": 5
        }
      }
    ]"
```

Output:

```
{
  "TableDescription": {
```

```
"AttributeDefinitions": [
  {
    "AttributeName": "GameTitle",
    "AttributeType": "S"
  },
  {
    "AttributeName": "TopScore",
    "AttributeType": "N"
  },
  {
    "AttributeName": "UserId",
    "AttributeType": "S"
  }
],
"TableName": "GameScores",
"KeySchema": [
  {
    "AttributeName": "UserId",
    "KeyType": "HASH"
  },
  {
    "AttributeName": "GameTitle",
    "KeyType": "RANGE"
  }
],
"TableStatus": "CREATING",
"CreationDateTime": "2020-05-26T17:28:15.602000-07:00",
"ProvisionedThroughput": {
  "NumberOfDecreasesToday": 0,
  "ReadCapacityUnits": 10,
  "WriteCapacityUnits": 5
},
"TableSizeBytes": 0,
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
"TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"GlobalSecondaryIndexes": [
  {
    "IndexName": "GameTitleIndex",
    "KeySchema": [
      {
        "AttributeName": "GameTitle",
        "KeyType": "HASH"
      }
    ],
  },
]
```

```

        {
            "AttributeName": "TopScore",
            "KeyType": "RANGE"
        }
    ],
    "Projection": {
        "ProjectionType": "INCLUDE",
        "NonKeyAttributes": [
            "UserId"
        ]
    },
    "IndexStatus": "CREATING",
    "ProvisionedThroughput": {
        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 5
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameTitleIndex"
    }
]
}
}

```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 6: Untuk membuat tabel dengan beberapa Indeks Sekunder Global sekaligus

Contoh berikut membuat tabel bernama GameScores dengan dua Global Secondary Indexes. Skema GSI diteruskan melalui file, bukan pada baris perintah.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-definitions AttributeName=UserId,AttributeType=S
AttributeName=GameTitle,AttributeType=S AttributeName=TopScore,AttributeType=N
AttributeName=Date,AttributeType=S \
  --key-schema AttributeName=UserId,KeyType=HASH
AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --global-secondary-indexes file://gsi.json

```

Isi dari `gsi.json`:

```
[
  {
    "IndexName": "GameTitleIndex",
    "KeySchema": [
      {
        "AttributeName": "GameTitle",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "TopScore",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "ProvisionedThroughput": {
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 5
    }
  },
  {
    "IndexName": "GameDateIndex",
    "KeySchema": [
      {
        "AttributeName": "GameTitle",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "Date",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "ProvisionedThroughput": {
      "ReadCapacityUnits": 5,
      "WriteCapacityUnits": 5
    }
  }
]
```

```
]
```

**Output:**

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Date",
        "AttributeType": "S"
      },
      {
        "AttributeName": "GameTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "TopScore",
        "AttributeType": "N"
      },
      {
        "AttributeName": "UserId",
        "AttributeType": "S"
      }
    ],
    "TableName": "GameScores",
    "KeySchema": [
      {
        "AttributeName": "UserId",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "GameTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": "2020-08-04T16:40:55.524000-07:00",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 5
    },
    "TableSizeBytes": 0,
  }
}
```

```
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
"TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"GlobalSecondaryIndexes": [
  {
    "IndexName": "GameTitleIndex",
    "KeySchema": [
      {
        "AttributeName": "GameTitle",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "TopScore",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "IndexStatus": "CREATING",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 5
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameTitleIndex"
  },
  {
    "IndexName": "GameDateIndex",
    "KeySchema": [
      {
        "AttributeName": "GameTitle",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "Date",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    }
  }
]
```

```

    },
    "IndexStatus": "CREATING",
    "ProvisionedThroughput": {
        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 5,
        "WriteCapacityUnits": 5
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameDateIndex"
    }
  ]
}
}

```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 7: Untuk membuat tabel dengan Streams diaktifkan

Contoh berikut membuat tabel yang disebut GameScores dengan DynamoDB Streams diaktifkan. Gambar baru dan lama dari setiap item akan ditulis ke aliran.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-definitions AttributeName=UserId,AttributeType=S
AttributeName=GameTitle,AttributeType=S \
  --key-schema AttributeName=UserId,KeyType=HASH
AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --stream-specification StreamEnabled=TRUE,StreamViewType=NEW_AND_OLD_IMAGES

```

Output:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "GameTitle",
        "AttributeType": "S"
      },
      {

```

```

        "AttributeName": "UserId",
        "AttributeType": "S"
    }
],
"TableName": "GameScores",
"KeySchema": [
    {
        "AttributeName": "UserId",
        "KeyType": "HASH"
    },
    {
        "AttributeName": "GameTitle",
        "KeyType": "RANGE"
    }
],
"TableStatus": "CREATING",
"CreationDateTime": "2020-05-27T10:49:34.056000-07:00",
"ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
},
"TableSizeBytes": 0,
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
"TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"StreamSpecification": {
    "StreamEnabled": true,
    "StreamViewType": "NEW_AND_OLD_IMAGES"
},
"LatestStreamLabel": "2020-05-27T17:49:34.056",
"LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/stream/2020-05-27T17:49:34.056"
}
}

```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Tabel](#) dalam Panduan Developer Amazon DynamoDB.

Contoh 8: Untuk membuat tabel dengan Keys-Only Stream diaktifkan

Contoh berikut membuat tabel yang disebut GameScores dengan DynamoDB Streams diaktifkan. Hanya atribut kunci dari item yang dimodifikasi yang ditulis ke aliran.



```
aws dynamodb create-table \  
  --table-name GameScores \  
  --attribute-definitions AttributeName=UserId,AttributeType=S  
  AttributeName=GameTitle,AttributeType=S \  
  --key-schema AttributeName=UserId,KeyType=HASH  
  AttributeName=GameTitle,KeyType=RANGE \  
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \  
  --stream-specification StreamEnabled=TRUE,StreamViewType=KEYS_ONLY
```

## Output:

```
{  
  "TableDescription": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "GameTitle",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "UserId",  
        "AttributeType": "S"  
      }  
    ],  
    "TableName": "GameScores",  
    "KeySchema": [  
      {  
        "AttributeName": "UserId",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "GameTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "TableStatus": "CREATING",  
    "CreationDateTime": "2023-05-25T18:45:34.140000+00:00",  
    "ProvisionedThroughput": {  
      "NumberOfDecreasesToday": 0,  
      "ReadCapacityUnits": 10,  
      "WriteCapacityUnits": 5  
    },  
    "TableSizeBytes": 0,  
    "ItemCount": 0,
```

```

    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "StreamSpecification": {
      "StreamEnabled": true,
      "StreamViewType": "KEYS_ONLY"
    },
    "LatestStreamLabel": "2023-05-25T18:45:34.140",
    "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/stream/2023-05-25T18:45:34.140",
    "DeletionProtectionEnabled": false
  }
}

```

Untuk informasi selengkapnya, lihat [Mengubah pengambilan data untuk DynamoDB Streams di Panduan Pengembang Amazon DynamoDB](#).

Contoh 9: Untuk membuat tabel dengan kelas Standard Infrequent Access

Contoh berikut membuat tabel yang disebut GameScores dan menetapkan kelas tabel Standard-Infrequent Access (DynamoDB Standard-IA). Kelas tabel ini dioptimalkan untuk penyimpanan menjadi biaya dominan.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-definitions AttributeName=UserId,AttributeType=S
AttributeName=GameTitle,AttributeType=S \
  --key-schema AttributeName=UserId,KeyType=HASH
AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --table-class STANDARD_INFREQUENT_ACCESS

```

Output:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "GameTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "UserId",

```

```

        "AttributeType": "S"
    }
],
"TableName": "GameScores",
"KeySchema": [
    {
        "AttributeName": "UserId",
        "KeyType": "HASH"
    },
    {
        "AttributeName": "GameTitle",
        "KeyType": "RANGE"
    }
],
"TableStatus": "CREATING",
"CreationDateTime": "2023-05-25T18:33:07.581000+00:00",
"ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
},
"TableSizeBytes": 0,
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
"TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"TableClassSummary": {
    "TableClass": "STANDARD_INFREQUENT_ACCESS"
},
"DeletionProtectionEnabled": false
}
}

```

Untuk informasi selengkapnya, lihat [Kelas tabel](#) di Panduan Pengembang Amazon DynamoDB.

#### Contoh 10: Membuat tabel dengan Hapus Perlindungan diaktifkan

Contoh berikut membuat tabel yang disebut GameScores dan memungkinkan perlindungan penghapusan.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-definitions AttributeName=UserId,AttributeType=S
  AttributeName=GameTitle,AttributeType=S \

```

```
--key-schema AttributeName=UserId,KeyType=HASH
AttributeName=GameTitle,KeyType=RANGE \
--provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
--deletion-protection-enabled
```

## Output:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "GameTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "UserId",
        "AttributeType": "S"
      }
    ],
    "TableName": "GameScores",
    "KeySchema": [
      {
        "AttributeName": "UserId",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "GameTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": "2023-05-25T23:02:17.093000+00:00",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "DeletionProtectionEnabled": true
  }
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Menggunakan perlindungan penghapusan](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [CreateTable](#) di Referensi AWS CLI Perintah.

## delete-backup

Contoh kode berikut menunjukkan cara menggunakan `delete-backup`.

### AWS CLI

Untuk menghapus cadangan DynamoDB yang ada

`delete-backup` Contoh berikut menghapus cadangan yang ada yang ditentukan.

```
aws dynamodb delete-backup \  
  --backup-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/  
  backup/01576616366715-b4e58d3a
```

Output:

```
{  
  "BackupDescription": {  
    "BackupDetails": {  
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection/backup/01576616366715-b4e58d3a",  
      "BackupName": "MusicCollectionBackup",  
      "BackupSizeBytes": 0,  
      "BackupStatus": "DELETED",  
      "BackupType": "USER",  
      "BackupCreationDateTime": 1576616366.715  
    },  
    "SourceTableDetails": {  
      "TableName": "MusicCollection",  
      "TableId": "b0c04bcc-309b-4352-b2ae-9088af169fe2",  
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection",  
      "TableSizeBytes": 0,  
      "KeySchema": [  
        {
```

```

        "AttributeName": "Artist",
        "KeyType": "HASH"
    },
    {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
    }
],
"TableCreationDateTime": 1576615228.571,
"ProvisionedThroughput": {
    "ReadCapacityUnits": 5,
    "WriteCapacityUnits": 5
},
"ItemCount": 0,
"BillingMode": "PROVISIONED"
},
"SourceTableFeatureDetails": {}
}
}

```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DeleteBackup](#) di Referensi AWS CLI Perintah.

## delete-item

Contoh kode berikut menunjukkan cara menggunakan `delete-item`.

### AWS CLI

Contoh 1: Untuk menghapus item

`delete-item` Contoh berikut menghapus item dari `MusicCollection` tabel dan meminta rincian tentang item yang telah dihapus dan kapasitas yang digunakan oleh permintaan.

```

aws dynamodb delete-item \
  --table-name MusicCollection \
  --key file://key.json \
  --return-values ALL_OLD \
  --return-consumed-capacity TOTAL \
  --return-item-collection-metrics SIZE

```

Isi dari `key.json`:

```
{
  "Artist": {"S": "No One You Know"},
  "SongTitle": {"S": "Scared of My Shadow"}
}
```

Output:

```
{
  "Attributes": {
    "AlbumTitle": {
      "S": "Blue Sky Blues"
    },
    "Artist": {
      "S": "No One You Know"
    },
    "SongTitle": {
      "S": "Scared of My Shadow"
    }
  },
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 2.0
  },
  "ItemCollectionMetrics": {
    "ItemCollectionKey": {
      "Artist": {
        "S": "No One You Know"
      }
    }
  },
  "SizeEstimateRangeGB": [
    0.0,
    1.0
  ]
}
```

Untuk informasi selengkapnya, lihat [Menulis Item](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 2: Untuk menghapus item secara kondisional

Contoh berikut menghapus item dari ProductCatalog tabel hanya jika salah satu Sporting Goods atau Gardening Supplies dan harganya antara 500 dan 600. ProductCategory la mengembalikan rincian tentang item yang telah dihapus.

```
aws dynamodb delete-item \  
  --table-name ProductCatalog \  
  --key '{"Id":{"N":"456"}}' \  
  --condition-expression "(ProductCategory IN (:cat1, :cat2)) and (#P between :lo  
and :hi)" \  
  --expression-attribute-names file://names.json \  
  --expression-attribute-values file://values.json \  
  --return-values ALL_OLD
```

Isi dari names.json:

```
{  
  "#P": "Price"  
}
```

Isi dari values.json:

```
{  
  ":cat1": {"S": "Sporting Goods"},  
  ":cat2": {"S": "Gardening Supplies"},  
  ":lo": {"N": "500"},  
  ":hi": {"N": "600"}  
}
```

Output:

```
{  
  "Attributes": {  
    "Id": {  
      "N": "456"  
    },  
    "Price": {  
      "N": "550"  
    },  
    "ProductCategory": {  
      "S": "Sporting Goods"  
    }  
  }  
}
```



```
}  
}
```

Untuk informasi selengkapnya, lihat [Menulis Item](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DeleteItem](#) di Referensi AWS CLI Perintah.

## delete-table

Contoh kode berikut menunjukkan cara menggunakan `delete-table`.

### AWS CLI

Untuk menghapus tabel

`delete-table` Contoh berikut menghapus `MusicCollection` tabel.

```
aws dynamodb delete-table \  
  --table-name MusicCollection
```

Output:

```
{  
  "TableDescription": {  
    "TableStatus": "DELETING",  
    "TableSizeBytes": 0,  
    "ItemCount": 0,  
    "TableName": "MusicCollection",  
    "ProvisionedThroughput": {  
      "NumberOfDecreasesToday": 0,  
      "WriteCapacityUnits": 5,  
      "ReadCapacityUnits": 5  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Tabel di Panduan](#) Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DeleteTable](#) di Referensi AWS CLI Perintah.

## describe-backup

Contoh kode berikut menunjukkan cara menggunakan `describe-backup`.

### AWS CLI

Untuk mendapatkan informasi tentang cadangan tabel yang ada

`describe-backup` Contoh berikut menampilkan informasi tentang cadangan yang ada yang ditentukan.

```
aws dynamodb describe-backup \
  --backup-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
  backup/01576616366715-b4e58d3a
```

Output:

```
{
  "BackupDescription": {
    "BackupDetails": {
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
      "BackupName": "MusicCollectionBackup",
      "BackupSizeBytes": 0,
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupCreationDateTime": 1576616366.715
    },
    "SourceTableDetails": {
      "TableName": "MusicCollection",
      "TableId": "b0c04bcc-309b-4352-b2ae-9088af169fe2",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "TableSizeBytes": 0,
      "KeySchema": [
        {
          "AttributeName": "Artist",
          "KeyType": "HASH"
        },
        {
          "AttributeName": "SongTitle",
          "KeyType": "RANGE"
        }
      ]
    }
  }
}
```

```

    ],
    "TableCreationDateTime": 1576615228.571,
    "ProvisionedThroughput": {
        "ReadCapacityUnits": 5,
        "WriteCapacityUnits": 5
    },
    "ItemCount": 0,
    "BillingMode": "PROVISIONED"
  },
  "SourceTableFeatureDetails": {}
}

```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DescribeBackup](#) di Referensi AWS CLI Perintah.

## describe-continuous-backups

Contoh kode berikut menunjukkan cara menggunakan `describe-continuous-backups`.

### AWS CLI

Untuk mendapatkan informasi tentang backup berkelanjutan untuk tabel DynamoDB

`describe-continuous-backups` Contoh berikut menampilkan rincian tentang pengaturan cadangan berkelanjutan untuk `MusicCollection` tabel.

```
aws dynamodb describe-continuous-backups \
  --table-name MusicCollection
```

### Output:

```

{
  "ContinuousBackupsDescription": {
    "ContinuousBackupsStatus": "ENABLED",
    "PointInTimeRecoveryDescription": {
      "PointInTimeRecoveryStatus": "DISABLED"
    }
  }
}

```

```
}
```

Untuk informasi selengkapnya, lihat Pemulihan [Point-in-Time untuk DynamoDB](#) di [Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DescribeContinuousBackups](#) di Referensi AWS CLI Perintah.

## describe-contributor-insights

Contoh kode berikut menunjukkan cara menggunakan `describe-contributor-insights`.

### AWS CLI

Untuk melihat setelan Contributor Insights untuk tabel DynamoDB

`describe-contributor-insights` Contoh berikut menampilkan pengaturan Contributor Insights untuk `MusicCollection` tabel dan indeks sekunder `AlbumTitle-index` global.

```
aws dynamodb describe-contributor-insights \  
  --table-name MusicCollection \  
  --index-name AlbumTitle-index
```

### Output:

```
{  
  "TableName": "MusicCollection",  
  "IndexName": "AlbumTitle-index",  
  "ContributorInsightsRuleList": [  
    "DynamoDBContributorInsights-PKC-MusicCollection-1576629651520",  
    "DynamoDBContributorInsights-SKC-MusicCollection-1576629651520",  
    "DynamoDBContributorInsights-PKT-MusicCollection-1576629651520",  
    "DynamoDBContributorInsights-SKT-MusicCollection-1576629651520"  
  ],  
  "ContributorInsightsStatus": "ENABLED",  
  "LastUpdateDateTime": 1576629654.78  
}
```

Untuk informasi selengkapnya, lihat [Menganalisis Akses Data Menggunakan Wawasan CloudWatch Kontributor untuk DynamoDB](#) di [Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DescribeContributorInsights](#) di Referensi AWS CLI Perintah.

## describe-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoints`.

### AWS CLI

Untuk melihat informasi titik akhir regional

`describe-endpoints` Contoh berikut menampilkan detail tentang titik akhir untuk AWS Wilayah saat ini.

```
aws dynamodb describe-endpoints
```

Output:

```
{
  "Endpoints": [
    {
      "Address": "dynamodb.us-west-2.amazonaws.com",
      "CachePeriodInMinutes": 1440
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir dan Kuota Amazon DynamoDB di Referensi Umum.AWS](#)

- Untuk detail API, lihat [DescribeEndpoints](#) di Referensi AWS CLI Perintah.

## describe-global-table-settings

Contoh kode berikut menunjukkan cara menggunakan `describe-global-table-settings`.

### AWS CLI

Untuk mendapatkan informasi tentang pengaturan tabel global DynamoDB

`describe-global-table-settings` Contoh berikut menampilkan pengaturan untuk tabel `MusicCollection` global.

```
aws dynamodb describe-global-table-settings \
```

```
--global-table-name MusicCollection
```

Output:

```
{
  "GlobalTableName": "MusicCollection",
  "ReplicaSettings": [
    {
      "RegionName": "us-east-1",
      "ReplicaStatus": "ACTIVE",
      "ReplicaProvisionedReadCapacityUnits": 10,
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "AutoScalingDisabled": true
      },
      "ReplicaProvisionedWriteCapacityUnits": 5,
      "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "AutoScalingDisabled": true
      }
    },
    {
      "RegionName": "us-east-2",
      "ReplicaStatus": "ACTIVE",
      "ReplicaProvisionedReadCapacityUnits": 10,
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "AutoScalingDisabled": true
      },
      "ReplicaProvisionedWriteCapacityUnits": 5,
      "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "AutoScalingDisabled": true
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DescribeGlobalTableSettings](#) di Referensi AWS CLI Perintah.

## **describe-global-table**

Contoh kode berikut menunjukkan cara menggunakan `describe-global-table`.

## AWS CLI

Untuk menampilkan informasi tentang tabel global DynamoDB

`describe-global-table` Contoh berikut menampilkan rincian tentang tabel `MusicCollection` global.

```
aws dynamodb describe-global-table \  
--global-table-name MusicCollection
```

Output:

```
{  
  "GlobalTableDescription": {  
    "ReplicationGroup": [  
      {  
        "RegionName": "us-east-2"  
      },  
      {  
        "RegionName": "us-east-1"  
      }  
    ],  
    "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/  
MusicCollection",  
    "CreationDateTime": 1576625818.532,  
    "GlobalTableStatus": "ACTIVE",  
    "GlobalTableName": "MusicCollection"  
  }  
}
```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DescribeGlobalTable](#) di Referensi AWS CLI Perintah.

## **describe-limits**

Contoh kode berikut menunjukkan cara menggunakan `describe-limits`.

## AWS CLI

Untuk melihat batas kapasitas yang disediakan

`describe-limits` Contoh berikut menampilkan batas kapasitas yang disediakan untuk akun Anda di Wilayah saat ini. AWS

```
aws dynamodb describe-limits
```

Output:

```
{
  "AccountMaxReadCapacityUnits": 80000,
  "AccountMaxWriteCapacityUnits": 80000,
  "TableMaxReadCapacityUnits": 40000,
  "TableMaxWriteCapacityUnits": 40000
}
```

Untuk informasi selengkapnya, lihat [Membatasi di DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DescribeLimits](#) di Referensi AWS CLI Perintah.

## **describe-table-replica-auto-scaling**

Contoh kode berikut menunjukkan cara menggunakan `describe-table-replica-auto-scaling`.

AWS CLI

Untuk melihat pengaturan penskalaan otomatis di seluruh replika tabel global

`describe-table-replica-auto-scaling` Contoh berikut menampilkan pengaturan penskalaan otomatis di seluruh replika tabel `MusicCollection` global.

```
aws dynamodb describe-table-replica-auto-scaling \
  --table-name MusicCollection
```

Output:

```
{
  "TableAutoScalingDescription": {
    "TableName": "MusicCollection",
    "TableStatus": "ACTIVE",
    "Replicas": [
      {
```



```

    "RegionName": "us-east-1",
    "GlobalSecondaryIndexes": [],
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
      "MinimumUnits": 5,
      "MaximumUnits": 40000,
      "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
      "ScalingPolicies": [
        {
          "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
          "TargetTrackingScalingPolicyConfiguration": {
            "TargetValue": 70.0
          }
        }
      ]
    },
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
      "MinimumUnits": 5,
      "MaximumUnits": 40000,
      "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
      "ScalingPolicies": [
        {
          "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
          "TargetTrackingScalingPolicyConfiguration": {
            "TargetValue": 70.0
          }
        }
      ]
    },
    "ReplicaStatus": "ACTIVE"
  },
  {
    "RegionName": "us-east-2",
    "GlobalSecondaryIndexes": [],
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
      "MinimumUnits": 5,
      "MaximumUnits": 40000,

```

```

        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ],
    },
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ],
    },
    "ReplicaStatus": "ACTIVE"
}
]
}
}

```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DescribeTableReplicaAutoScaling](#) di Referensi AWS CLI Perintah.

## describe-table

Contoh kode berikut menunjukkan cara menggunakan `describe-table`.

## AWS CLI

Untuk menggambarkan tabel

`describe-table` Contoh berikut menjelaskan `MusicCollection` tabel.

```
aws dynamodb describe-table \  
  --table-name MusicCollection
```

Output:

```
{  
  "Table": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "Artist",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "AttributeType": "S"  
      }  
    ],  
    "ProvisionedThroughput": {  
      "NumberOfDecreasesToday": 0,  
      "WriteCapacityUnits": 5,  
      "ReadCapacityUnits": 5  
    },  
    "TableSizeBytes": 0,  
    "TableName": "MusicCollection",  
    "TableStatus": "ACTIVE",  
    "KeySchema": [  
      {  
        "KeyType": "HASH",  
        "AttributeName": "Artist"  
      },  
      {  
        "KeyType": "RANGE",  
        "AttributeName": "SongTitle"  
      }  
    ],  
    "ItemCount": 0,  
    "CreationDateTime": 1421866952.062  
  }  
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Menjelaskan Tabel di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DescribeTable](#) di Referensi AWS CLI Perintah.

## **describe-time-to-live**

Contoh kode berikut menunjukkan cara menggunakan `describe-time-to-live`.

### AWS CLI

Untuk melihat pengaturan Waktu ke Langsung untuk tabel

`describe-time-to-live` Contoh berikut menampilkan Pengaturan Time to Live untuk `MusicCollection` tabel.

```
aws dynamodb describe-time-to-live \  
  --table-name MusicCollection
```

Output:

```
{  
  "TimeToLiveDescription": {  
    "TimeToLiveStatus": "ENABLED",  
    "AttributeName": "ttl"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Waktu untuk Hidup](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [DescribeTimeToLive](#) di Referensi AWS CLI Perintah.

## **get-item**

Contoh kode berikut menunjukkan cara menggunakan `get-item`.

## AWS CLI

Contoh 1: Untuk membaca item dalam tabel

`get-item` Contoh berikut mengambil item dari `MusicCollection` tabel. Tabel memiliki kunci hash-and-range utama (`Artist` dan `SongTitle`), jadi Anda harus menentukan kedua atribut ini. Perintah tersebut juga meminta informasi tentang kapasitas baca yang dikonsumsi oleh operasi.

```
aws dynamodb get-item \  
  --table-name MusicCollection \  
  --key file://key.json \  
  --return-consumed-capacity TOTAL
```

Isi dari `key.json`:

```
{  
  "Artist": {"S": "Acme Band"},  
  "SongTitle": {"S": "Happy Day"}  
}
```

Output:

```
{  
  "Item": {  
    "AlbumTitle": {  
      "S": "Songs About Life"  
    },  
    "SongTitle": {  
      "S": "Happy Day"  
    },  
    "Artist": {  
      "S": "Acme Band"  
    }  
  },  
  "ConsumedCapacity": {  
    "TableName": "MusicCollection",  
    "CapacityUnits": 0.5  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membaca Item](#) di Panduan Pengembang Amazon DynamoDB.

## Contoh 2: Untuk membaca item menggunakan pembacaan yang konsisten

Contoh berikut mengambil item dari MusicCollection tabel menggunakan pembacaan yang sangat konsisten.

```
aws dynamodb get-item \  
  --table-name MusicCollection \  
  --key file://key.json \  
  --consistent-read \  
  --return-consumed-capacity TOTAL
```

Isi dari key.json:

```
{  
  "Artist": {"S": "Acme Band"},  
  "SongTitle": {"S": "Happy Day"}  
}
```

Output:

```
{  
  "Item": {  
    "AlbumTitle": {  
      "S": "Songs About Life"  
    },  
    "SongTitle": {  
      "S": "Happy Day"  
    },  
    "Artist": {  
      "S": "Acme Band"  
    }  
  },  
  "ConsumedCapacity": {  
    "TableName": "MusicCollection",  
    "CapacityUnits": 1.0  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membaca Item](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 3: Untuk mengambil atribut tertentu dari suatu item

Contoh berikut menggunakan ekspresi proyeksi untuk mengambil hanya tiga atribut dari item yang diinginkan.

```
aws dynamodb get-item \  
  --table-name ProductCatalog \  
  --key '{"Id": {"N": "102"}}' \  
  --projection-expression "#T, #C, #P" \  
  --expression-attribute-names file://names.json
```

Isi dari `names.json`:

```
{  
  "#T": "Title",  
  "#C": "ProductCategory",  
  "#P": "Price"  
}
```

Output:

```
{  
  "Item": {  
    "Price": {  
      "N": "20"  
    },  
    "Title": {  
      "S": "Book 102 Title"  
    },  
    "ProductCategory": {  
      "S": "Book"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membaca Item](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [GetItem](#) di Referensi AWS CLI Perintah.

## list-backups

Contoh kode berikut menunjukkan cara menggunakan `list-backups`.

## AWS CLI

Contoh 1: Untuk daftar semua backup DynamoDB yang ada

`list-backups` Contoh berikut mencantumkan semua backup yang ada.

```
aws dynamodb list-backups
```

Output:

```
{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
      "BackupName": "MusicCollectionBackup1",
      "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupSizeBytes": 170
    },
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
      "BackupName": "MusicCollectionBackup2",
      "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupSizeBytes": 400
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.



Contoh 2: Untuk mencantumkan cadangan yang dibuat pengguna dalam rentang waktu tertentu

Contoh berikut hanya mencantumkan cadangan MusicCollection tabel yang dibuat oleh pengguna (bukan yang dibuat secara otomatis oleh DynamoDB) dengan tanggal pembuatan antara 1 Januari 2020 dan 1 Maret 2020.

```
aws dynamodb list-backups \  
  --table-name MusicCollection \  
  --time-range-lower-bound 1577836800 \  
  --time-range-upper-bound 1583020800 \  
  --backup-type USER
```

Output:

```
{  
  "BackupSummaries": [  
    {  
      "TableName": "MusicCollection",  
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection",  
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection/backup/01234567890123-a1bcd234",  
      "BackupName": "MusicCollectionBackup1",  
      "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",  
      "BackupStatus": "AVAILABLE",  
      "BackupType": "USER",  
      "BackupSizeBytes": 170  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 3: Untuk membatasi ukuran halaman

Contoh berikut mengembalikan daftar semua cadangan yang ada, tetapi hanya mengambil satu item di setiap panggilan, melakukan beberapa panggilan jika perlu untuk mendapatkan seluruh daftar. Membatasi ukuran halaman berguna saat menjalankan perintah daftar pada sejumlah besar sumber daya, yang dapat mengakibatkan kesalahan “waktu habis” saat menggunakan ukuran halaman default 1000.

```
aws dynamodb list-backups \  
--page-size 1
```

**Output:**

```
{  
  "BackupSummaries": [  
    {  
      "TableName": "MusicCollection",  
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection",  
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection/backup/01234567890123-a1bcd234",  
      "BackupName": "MusicCollectionBackup1",  
      "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",  
      "BackupStatus": "AVAILABLE",  
      "BackupType": "USER",  
      "BackupSizeBytes": 170  
    },  
    {  
      "TableName": "MusicCollection",  
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection",  
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection/backup/01234567890123-b2abc345",  
      "BackupName": "MusicCollectionBackup2",  
      "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",  
      "BackupStatus": "AVAILABLE",  
      "BackupType": "USER",  
      "BackupSizeBytes": 400  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 4: Untuk membatasi jumlah item yang dikembalikan

Contoh berikut membatasi jumlah item yang dikembalikan ke 1. Respons mencakup NextToken nilai yang dapat digunakan untuk mengambil halaman hasil berikutnya.

```
aws dynamodb list-backups \
  --max-items 1
```

Output:

```
{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
      "BackupName": "MusicCollectionBackup1",
      "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupSizeBytes": 170
    }
  ],
  "NextToken":
  "abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0ppqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
}
```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 5: Untuk mengambil halaman hasil berikutnya

Perintah berikut menggunakan NextToken nilai dari panggilan sebelumnya ke list-backups perintah untuk mengambil halaman lain dari hasil. Karena respons dalam kasus ini tidak termasuk NextToken nilai, kami tahu bahwa kami telah mencapai akhir hasil.

```
aws dynamodb list-backups \
  --starting-token
  abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0ppqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Output

```
{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
      "BackupName": "MusicCollectionBackup2",
      "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupSizeBytes": 400
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [ListBackups](#) di Referensi AWS CLI Perintah.

## list-contributor-insights

Contoh kode berikut menunjukkan cara menggunakan `list-contributor-insights`.

### AWS CLI

Contoh 1: Untuk melihat daftar ringkasan Contributor Insights

`list-contributor-insights` Contoh berikut menampilkan daftar ringkasan Contributor Insights.

```
aws dynamodb list-contributor-insights
```

Output:

```
{
  "ContributorInsightsSummaries": [
    {
```

```

        "TableName": "MusicCollection",
        "IndexName": "AlbumTitle-index",
        "ContributorInsightsStatus": "ENABLED"
    },
    {
        "TableName": "ProductCatalog",
        "ContributorInsightsStatus": "ENABLED"
    },
    {
        "TableName": "Forum",
        "ContributorInsightsStatus": "ENABLED"
    },
    {
        "TableName": "Reply",
        "ContributorInsightsStatus": "ENABLED"
    },
    {
        "TableName": "Thread",
        "ContributorInsightsStatus": "ENABLED"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menganalisis Akses Data Menggunakan Wawasan CloudWatch Kontributor untuk DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

Contoh 2: Untuk membatasi jumlah item yang dikembalikan

Contoh berikut membatasi jumlah item yang dikembalikan ke 4. Respons mencakup NextToken nilai yang dapat digunakan untuk mengambil halaman hasil berikutnya.

```

aws dynamodb list-contributor-insights \
  --max-results 4

```

Output:

```

{
  "ContributorInsightsSummaries": [
    {
      "TableName": "MusicCollection",
      "IndexName": "AlbumTitle-index",
      "ContributorInsightsStatus": "ENABLED"
    }
  ]
}

```

```

    },
    {
      "TableName": "ProductCatalog",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "Forum",
      "ContributorInsightsStatus": "ENABLED"
    }
  ],
  "NextToken":
  "abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFghI2Jk3LmnoPQ6RST9"
}

```

Untuk informasi selengkapnya, lihat [Menganalisis Akses Data Menggunakan Wawasan CloudWatch Kontributor untuk DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

Contoh 3: Untuk mengambil halaman berikutnya dari hasil

Perintah berikut menggunakan NextToken nilai dari panggilan sebelumnya ke `list-contributor-insights` perintah untuk mengambil halaman lain dari hasil. Karena respons dalam kasus ini tidak termasuk NextToken nilai, kami tahu bahwa kami telah mencapai akhir hasil.

```

aws dynamodb list-contributor-insights \
  --max-results 4 \
  --next-token
abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFghI2Jk3LmnoPQ6RST9

```

Output:

```

{
  "ContributorInsightsSummaries": [
    {
      "TableName": "Reply",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "Thread",
      "ContributorInsightsStatus": "ENABLED"
    }
  ]
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Menganalisis Akses Data Menggunakan Wawasan CloudWatch Kontributor untuk DynamoDB](#) di [Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [ListContributorInsights](#) di Referensi AWS CLI Perintah.

## list-global-tables

Contoh kode berikut menunjukkan cara menggunakan `list-global-tables`.

### AWS CLI

Untuk daftar tabel global DynamoDB yang ada

`list-global-tables` Contoh berikut mencantumkan semua tabel global yang ada.

```
aws dynamodb list-global-tables
```

Output:

```
{
  "GlobalTables": [
    {
      "GlobalTableName": "MusicCollection",
      "ReplicationGroup": [
        {
          "RegionName": "us-east-2"
        },
        {
          "RegionName": "us-east-1"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di [Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [ListGlobalTables](#) di Referensi AWS CLI Perintah.

## list-tables

Contoh kode berikut menunjukkan cara menggunakan `list-tables`.

### AWS CLI

Contoh 1: Untuk daftar tabel

`list-tables` Contoh berikut mencantumkan semua tabel yang terkait dengan AWS akun saat ini dan Wilayah.

```
aws dynamodb list-tables
```

Output:

```
{
  "TableNames": [
    "Forum",
    "ProductCatalog",
    "Reply",
    "Thread"
  ]
}
```

Untuk informasi selengkapnya, lihat [Daftar Nama Tabel](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 2: Untuk membatasi ukuran halaman

Contoh berikut mengembalikan daftar semua tabel yang ada, tetapi mengambil hanya satu item dalam setiap panggilan, melakukan beberapa panggilan jika perlu untuk mendapatkan seluruh daftar. Membatasi ukuran halaman berguna saat menjalankan perintah daftar pada sejumlah besar sumber daya, yang dapat mengakibatkan kesalahan “waktu habis” saat menggunakan ukuran halaman default 1000.

```
aws dynamodb list-tables \
  --page-size 1
```

Output:

```
{
```



```
"TableNames": [  
  "Forum",  
  "ProductCatalog",  
  "Reply",  
  "Thread"  
]  
}
```

Untuk informasi selengkapnya, lihat [Daftar Nama Tabel](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 3: Untuk membatasi jumlah item yang dikembalikan

Contoh berikut membatasi jumlah item yang dikembalikan ke 2. Respons mencakup NextToken nilai yang dapat digunakan untuk mengambil halaman hasil berikutnya.

```
aws dynamodb list-tables \  
  --max-items 2
```

Output:

```
{  
  "TableNames": [  
    "Forum",  
    "ProductCatalog"  
  ],  
  "NextToken":  
  "abCDeFGhIJKlMnOPqrSTuvwXYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"  
}
```

Untuk informasi selengkapnya, lihat [Daftar Nama Tabel](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 4: Untuk mengambil halaman hasil berikutnya

Perintah berikut menggunakan NextToken nilai dari panggilan sebelumnya ke `list-tables` perintah untuk mengambil halaman lain dari hasil. Karena respons dalam kasus ini tidak termasuk NextToken nilai, kami tahu bahwa kami telah mencapai akhir hasil.

```
aws dynamodb list-tables \  
  --next-token
```

```
--starting-token  
abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFghI2Jk3LmnoPQ6RST9
```

Output:

```
{  
  "TableNames": [  
    "Reply",  
    "Thread"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar Nama Tabel](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [ListTables](#) di Referensi AWS CLI Perintah.

## list-tags-of-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-of-resource`.

AWS CLI

Contoh 1: Untuk daftar tag dari sumber daya DynamoDB

`list-tags-of-resource` Contoh berikut menampilkan tag untuk `MusicCollection` tabel.

```
aws dynamodb list-tags-of-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Owner",  
      "Value": "blueTeam"  
    },  
    {  
      "Key": "Environment",  
      "Value": "Production"  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 2: Untuk membatasi jumlah tag yang dikembalikan

Contoh berikut membatasi jumlah tag yang dikembalikan ke 1. Respons mencakup NextToken nilai yang dapat digunakan untuk mengambil halaman hasil berikutnya.

```
aws dynamodb list-tags-of-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \  
  --max-items 1
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Owner",  
      "Value": "blueTeam"  
    }  
  ],  
  "NextToken":  
  "abCDeFGhiJKlmnOPqrSTuvwXYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"  
}
```

Untuk informasi selengkapnya, lihat [Menandai DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 3: Untuk mengambil halaman berikutnya dari hasil

Perintah berikut menggunakan NextToken nilai dari panggilan sebelumnya ke list-tags-of-resource perintah untuk mengambil halaman lain dari hasil. Karena respons dalam kasus ini tidak termasuk NextToken nilai, kami tahu bahwa kami telah mencapai akhir hasil.

```
aws dynamodb list-tags-of-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \  
  --next-token abCDeFGhiJKlmnOPqrSTuvwXYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

```
--starting-token  
abCDeFGhiJKlMnOPqrSTuvwXYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFghI2Jk3LmnoPQ6RST9
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Environment",  
      "Value": "Production"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [ListTagsOfResource](#) di Referensi AWS CLI Perintah.

## put-item

Contoh kode berikut menunjukkan cara menggunakan `put-item`.

### AWS CLI

Contoh 1: Untuk menambahkan item ke tabel

`put-item` Contoh berikut menambahkan item baru ke `MusicCollection` tabel.

```
aws dynamodb put-item \  
  --table-name MusicCollection \  
  --item file://item.json \  
  --return-consumed-capacity TOTAL \  
  --return-item-collection-metrics SIZE
```

Isi dari `item.json`:

```
{  
  "Artist": {"S": "No One You Know"},  
  "SongTitle": {"S": "Call Me Today"},  
  "AlbumTitle": {"S": "Greatest Hits"}  
}
```

```
}

```

### Output:

```
{
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 1.0
  },
  "ItemCollectionMetrics": {
    "ItemCollectionKey": {
      "Artist": {
        "S": "No One You Know"
      }
    },
    "SizeEstimateRangeGB": [
      0.0,
      1.0
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Menulis Item](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 2: Untuk menerima item secara kondisional dalam tabel

`put-item` Contoh berikut menerima item yang ada dalam `MusicCollection` tabel hanya jika item yang ada memiliki `AlbumTitle` atribut dengan nilai. `Greatest Hits` Perintah mengembalikan nilai item sebelumnya.

```
aws dynamodb put-item \
  --table-name MusicCollection \
  --item file://item.json \
  --condition-expression "#A = :A" \
  --expression-attribute-names file://names.json \
  --expression-attribute-values file://values.json \
  --return-values ALL_OLD
```

Isi dari `item.json`:

```
{

```

```
"Artist": {"S": "No One You Know"},
"SongTitle": {"S": "Call Me Today"},
"AlbumTitle": {"S": "Somewhat Famous"}
}
```

Isi dari `names.json`:

```
{
  "#A": "AlbumTitle"
}
```

Isi dari `values.json`:

```
{
  ":A": {"S": "Greatest Hits"}
}
```

Output:

```
{
  "Attributes": {
    "AlbumTitle": {
      "S": "Greatest Hits"
    },
    "Artist": {
      "S": "No One You Know"
    },
    "SongTitle": {
      "S": "Call Me Today"
    }
  }
}
```

Jika kunci sudah ada, Anda akan melihat output berikut:

```
A client error (ConditionalCheckFailedException) occurred when calling the PutItem
operation: The conditional request failed.
```

Untuk informasi selengkapnya, lihat [Menulis Item](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [PutItem](#) di Referensi AWS CLI Perintah.

## query

Contoh kode berikut menunjukkan cara menggunakan `query`.

### AWS CLI

Contoh 1: Untuk menanyakan tabel

query Contoh berikut `query` item dalam `MusicCollection` tabel. Tabel memiliki kunci hash-and-range utama (`Artist` dan `SongTitle`), tetapi kueri ini hanya menentukan nilai kunci hash. Ini mengembalikan judul lagu oleh artis bernama "No One You Know".

```
aws dynamodb query \  
  --table-name MusicCollection \  
  --projection-expression "SongTitle" \  
  --key-condition-expression "Artist = :v1" \  
  --expression-attribute-values file://expression-attributes.json \  
  --return-consumed-capacity TOTAL
```

Isi dari `expression-attributes.json`:

```
{  
  ":v1": {"S": "No One You Know"}  
}
```

Output:

```
{  
  "Items": [  
    {  
      "SongTitle": {  
        "S": "Call Me Today"  
      },  
      "SongTitle": {  
        "S": "Scared of My Shadow"  
      }  
    }  
  ],  
  "Count": 2,  
  "ScannedCount": 2,  
  "ConsumedCapacity": {  
    "TableName": "MusicCollection",
```

```
    "CapacityUnits": 0.5
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kueri di DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

Contoh 2: Untuk menanyakan tabel menggunakan pembacaan yang sangat konsisten dan melintasi indeks dalam urutan menurun

Contoh berikut melakukan kueri yang sama dengan contoh pertama, tetapi mengembalikan hasil dalam urutan terbalik dan menggunakan pembacaan yang sangat konsisten.

```
aws dynamodb query \
  --table-name MusicCollection \
  --projection-expression "SongTitle" \
  --key-condition-expression "Artist = :v1" \
  --expression-attribute-values file://expression-attributes.json \
  --consistent-read \
  --no-scan-index-forward \
  --return-consumed-capacity TOTAL
```

Isi dari `expression-attributes.json`:

```
{
  ":v1": {"S": "No One You Know"}
}
```

Output:

```
{
  "Items": [
    {
      "SongTitle": {
        "S": "Scared of My Shadow"
      }
    },
    {
      "SongTitle": {
        "S": "Call Me Today"
      }
    }
  ]
}
```



```

    ],
    "Count": 2,
    "ScannedCount": 2,
    "ConsumedCapacity": {
      "TableName": "MusicCollection",
      "CapacityUnits": 1.0
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kueri di DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

Contoh 3: Untuk menyaring hasil tertentu

Contoh berikut query MusicCollection tetapi mengecualikan hasil dengan nilai-nilai tertentu dalam atribut AlbumTitle. Perhatikan bahwa ini tidak mempengaruhi ScannedCount atau ConsumedCapacity, karena filter diterapkan setelah item telah dibaca.

```

aws dynamodb query \
  --table-name MusicCollection \
  --key-condition-expression "#n1 = :v1" \
  --filter-expression "NOT (#n2 IN (:v2, :v3))" \
  --expression-attribute-names file://names.json \
  --expression-attribute-values file://values.json \
  --return-consumed-capacity TOTAL

```

Isi dari values.json:

```

{
  ":v1": {"S": "No One You Know"},
  ":v2": {"S": "Blue Sky Blues"},
  ":v3": {"S": "Greatest Hits"}
}

```

Isi dari names.json:

```

{
  "#n1": "Artist",
  "#n2": "AlbumTitle"
}

```

Output:

```
{
  "Items": [
    {
      "AlbumTitle": {
        "S": "Somewhat Famous"
      },
      "Artist": {
        "S": "No One You Know"
      },
      "SongTitle": {
        "S": "Call Me Today"
      }
    }
  ],
  "Count": 1,
  "ScannedCount": 2,
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 0.5
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kueri di DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

Contoh 4: Untuk mengambil hanya jumlah item

Contoh berikut mengambil hitungan item yang cocok dengan query, tetapi tidak mengambil salah satu item itu sendiri.

```
aws dynamodb query \
  --table-name MusicCollection \
  --select COUNT \
  --key-condition-expression "Artist = :v1" \
  --expression-attribute-values file://expression-attributes.json
```

Isi dari `expression-attributes.json`:

```
{
  ":v1": {"S": "No One You Know"}
}
```

**Output:**

```
{
  "Count": 2,
  "ScannedCount": 2,
  "ConsumedCapacity": null
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kueri di DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

**Contoh 5: Untuk menanyakan indeks**

Contoh berikut query indeks AlbumTitleIndex sekunder lokal. Query mengembalikan semua atribut dari tabel dasar yang telah diproyeksikan ke indeks sekunder lokal. Perhatikan bahwa saat menanyakan indeks sekunder lokal atau indeks sekunder global, Anda juga harus memberikan nama tabel dasar menggunakan `table-name` parameter.

```
aws dynamodb query \
  --table-name MusicCollection \
  --index-name AlbumTitleIndex \
  --key-condition-expression "Artist = :v1" \
  --expression-attribute-values file://expression-attributes.json \
  --select ALL_PROJECTED_ATTRIBUTES \
  --return-consumed-capacity INDEXES
```

**Isi dari expression-attributes.json:**

```
{
  ":v1": {"S": "No One You Know"}
}
```

**Output:**

```
{
  "Items": [
    {
      "AlbumTitle": {
        "S": "Blue Sky Blues"
      },

```

```

    "Artist": {
      "S": "No One You Know"
    },
    "SongTitle": {
      "S": "Scared of My Shadow"
    }
  },
  {
    "AlbumTitle": {
      "S": "Somewhat Famous"
    },
    "Artist": {
      "S": "No One You Know"
    },
    "SongTitle": {
      "S": "Call Me Today"
    }
  }
],
"Count": 2,
"ScannedCount": 2,
"ConsumedCapacity": {
  "TableName": "MusicCollection",
  "CapacityUnits": 0.5,
  "Table": {
    "CapacityUnits": 0.0
  },
  "LocalSecondaryIndexes": {
    "AlbumTitleIndex": {
      "CapacityUnits": 0.5
    }
  }
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kueri di DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [Kueri](#) di Referensi Perintah AWS CLI .

## restore-table-from-backup

Contoh kode berikut menunjukkan cara menggunakan `restore-table-from-backup`.

## AWS CLI

Untuk mengembalikan tabel DynamoDB dari cadangan yang ada

`restore-table-from-backup` Contoh berikut mengembalikan tabel yang ditentukan dari cadangan yang ada.

```
aws dynamodb restore-table-from-backup \  
  --target-table-name MusicCollection \  
  --backup-arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/  
  backup/01576616366715-b4e58d3a
```

Output:

```
{  
  "TableDescription": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "Artist",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "AttributeType": "S"  
      }  
    ],  
    "TableName": "MusicCollection2",  
    "KeySchema": [  
      {  
        "AttributeName": "Artist",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "TableStatus": "CREATING",  
    "CreationDateTime": 1576618274.326,  
    "ProvisionedThroughput": {  
      "NumberOfDecreasesToday": 0,  
      "ReadCapacityUnits": 5,  
      "WriteCapacityUnits": 5  
    }  
  }  
}
```

```

    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection2",
    "TableId": "114865c9-5ef3-496c-b4d1-c4cbdd2d44fb",
    "BillingModeSummary": {
      "BillingMode": "PROVISIONED"
    },
    "RestoreSummary": {
      "SourceBackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
      "SourceTableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "RestoreDateTime": 1576616366.715,
      "RestoreInProgress": true
    }
  }
}
}

```

Untuk informasi selengkapnya, lihat [Pencadangan dan Pemulihan Sesuai Permintaan untuk DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [RestoreTableFromBackup](#) di Referensi AWS CLI Perintah.

## restore-table-to-point-in-time

Contoh kode berikut menunjukkan cara menggunakan `restore-table-to-point-in-time`.

### AWS CLI

Untuk mengembalikan tabel DynamoDB ke titik waktu

`restore-table-to-point-in-time` Contoh berikut mengembalikan `MusicCollection` tabel ke titik waktu yang ditentukan.

```

aws dynamodb restore-table-to-point-in-time \
  --source-table-name MusicCollection \
  --target-table-name MusicCollectionRestore \
  --restore-date-time 1576622404.0

```

Output:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollectionRestore",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": 1576623311.86,
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 5,
      "WriteCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollectionRestore",
    "TableId": "befd9e0e-1843-4dc6-a147-d6d00e85cb1f",
    "BillingModeSummary": {
      "BillingMode": "PROVISIONED"
    },
    "RestoreSummary": {
      "SourceTableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "RestoreDateTime": 1576622404.0,
      "RestoreInProgress": true
    }
  }
}
```

```
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat Pemulihan [Point-in-Time untuk DynamoDB](#) di [Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [RestoreTableToPointInTime](#) di Referensi AWS CLI Perintah.

## scan

Contoh kode berikut menunjukkan cara menggunakan `scan`.

### AWS CLI

Untuk memindai tabel

`scan` Contoh berikut memindai seluruh `MusicCollection` tabel, dan kemudian mempersempit hasilnya menjadi lagu-lagu oleh artis “No One You Know”. Untuk setiap item, hanya judul album dan judul lagu yang dikembalikan.

```
aws dynamodb scan \  
  --table-name MusicCollection \  
  --filter-expression "Artist = :a" \  
  --projection-expression "#ST, #AT" \  
  --expression-attribute-names file://expression-attribute-names.json \  
  --expression-attribute-values file://expression-attribute-values.json
```

Isi dari `expression-attribute-names.json`:

```
{  
  "#ST": "SongTitle",  
  "#AT": "AlbumTitle"  
}
```

Isi dari `expression-attribute-values.json`:

```
{  
  ":a": {"S": "No One You Know"}  
}
```

Output:



```
{
  "Count": 2,
  "Items": [
    {
      "SongTitle": {
        "S": "Call Me Today"
      },
      "AlbumTitle": {
        "S": "Somewhat Famous"
      }
    },
    {
      "SongTitle": {
        "S": "Scared of My Shadow"
      },
      "AlbumTitle": {
        "S": "Blue Sky Blues"
      }
    }
  ],
  "ScannedCount": 3,
  "ConsumedCapacity": null
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Pemindaian di DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [Scan](#) in Referensi Perintah AWS CLI .

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan tag-resource.

### AWS CLI

Untuk menambahkan tag ke sumber daya DynamoDB

tag-resource Contoh berikut menambahkan tag key/value pair ke tabel. MusicCollection

```
aws dynamodb tag-resource \
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
  --tags Key=Owner,Value=blueTeam
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## transact-get-items

Contoh kode berikut menunjukkan cara menggunakan `transact-get-items`.

### AWS CLI

Untuk mengambil beberapa item secara atom dari satu atau beberapa tabel

`transact-get-items` Contoh berikut mengambil beberapa item secara atom.

```
aws dynamodb transact-get-items \  
  --transact-items file://transact-items.json \  
  --return-consumed-capacity TOTAL
```

Isi dari `transact-items.json`:

```
[  
  {  
    "Get": {  
      "Key": {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      },  
      "TableName": "MusicCollection"  
    }  
  },  
  {  
    "Get": {  
      "Key": {  
        "Artist": {"S": "No One You Know"},  
        "SongTitle": {"S": "Call Me Today"}  
      },  
      "TableName": "MusicCollection"  
    }  
  }  
]
```

## Output:

```
{
  "ConsumedCapacity": [
    {
      "TableName": "MusicCollection",
      "CapacityUnits": 4.0,
      "ReadCapacityUnits": 4.0
    }
  ],
  "Responses": [
    {
      "Item": {
        "AlbumTitle": {
          "S": "Songs About Life"
        },
        "Artist": {
          "S": "Acme Band"
        },
        "SongTitle": {
          "S": "Happy Day"
        }
      }
    },
    {
      "Item": {
        "AlbumTitle": {
          "S": "Somewhat Famous"
        },
        "Artist": {
          "S": "No One You Know"
        },
        "SongTitle": {
          "S": "Call Me Today"
        }
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Alur Kerja Kompleks dengan Transaksi DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [TransactGetItems](#) di Referensi AWS CLI Perintah.

## transact-write-items

Contoh kode berikut menunjukkan cara menggunakan `transact-write-items`.

### AWS CLI

Contoh 1: Untuk menulis item secara atom ke satu atau lebih tabel

`transact-write-items` Contoh berikut memperbarui satu item dan menghapus yang lain. Operasi gagal jika salah satu operasi gagal, atau jika salah satu item berisi Rating atribut.

```
aws dynamodb transact-write-items \  
  --transact-items file://transact-items.json \  
  --return-consumed-capacity TOTAL \  
  --return-item-collection-metrics SIZE
```

Isi `transact-items.json` file:

```
[  
  {  
    "Update": {  
      "Key": {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      },  
      "UpdateExpression": "SET AlbumTitle = :newval",  
      "ExpressionAttributeValues": {  
        ":newval": {"S": "Updated Album Title"}  
      },  
      "TableName": "MusicCollection",  
      "ConditionExpression": "attribute_not_exists(Rating)"  
    },  
    {  
      "Delete": {  
        "Key": {  
          "Artist": {"S": "No One You Know"},  
          "SongTitle": {"S": "Call Me Today"}  
        },  
        "TableName": "MusicCollection",  
        "ConditionExpression": "attribute_not_exists(Rating)"  
      }  
    }  
  }  
]
```

```
]
```

**Output:**

```
{
  "ConsumedCapacity": [
    {
      "TableName": "MusicCollection",
      "CapacityUnits": 10.0,
      "WriteCapacityUnits": 10.0
    }
  ],
  "ItemCollectionMetrics": {
    "MusicCollection": [
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "No One You Know"
          }
        },
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      },
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "Acme Band"
          }
        },
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola Alur Kerja Kompleks dengan Transaksi DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

## Contoh 2: Untuk menulis item secara atom menggunakan token permintaan klien

Perintah berikut menggunakan token permintaan klien untuk membuat panggilan ke `transact-write-items` idempoten, yang berarti bahwa beberapa panggilan memiliki efek yang sama dengan satu panggilan tunggal.

```
aws dynamodb transact-write-items \  
  --transact-items file://transact-items.json \  
  --client-request-token abc123
```

Isi `transact-items.json` file:

```
[  
  {  
    "Update": {  
      "Key": {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      },  
      "UpdateExpression": "SET AlbumTitle = :newval",  
      "ExpressionAttributeValues": {  
        ":newval": {"S": "Updated Album Title"}  
      },  
      "TableName": "MusicCollection",  
      "ConditionExpression": "attribute_not_exists(Rating)"  
    }  
  },  
  {  
    "Delete": {  
      "Key": {  
        "Artist": {"S": "No One You Know"},  
        "SongTitle": {"S": "Call Me Today"}  
      },  
      "TableName": "MusicCollection",  
      "ConditionExpression": "attribute_not_exists(Rating)"  
    }  
  }  
]
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Alur Kerja Kompleks dengan Transaksi DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [TransactWriteItems](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya DynamoDB

`untag-resource` Contoh berikut menghapus tag dengan kunci `Owner` dari `MusicCollection` tabel.

```
aws dynamodb untag-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \  
  --tag-keys Owner
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai DynamoDB](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-continuous-backups

Contoh kode berikut menunjukkan cara menggunakan `update-continuous-backups`.

### AWS CLI

Untuk memperbarui pengaturan cadangan berkelanjutan untuk tabel DynamoDB

`update-continuous-backups` Contoh berikut memungkinkan point-in-time pemulihan untuk `MusicCollection` tabel.

```
aws dynamodb update-continuous-backups \  
  --table-name MusicCollection \  
  --point-in-time-recovery-specification PointInTimeRecoveryEnabled=true
```

**Output:**

```
{
  "ContinuousBackupsDescription": {
    "ContinuousBackupsStatus": "ENABLED",
    "PointInTimeRecoveryDescription": {
      "PointInTimeRecoveryStatus": "ENABLED",
      "EarliestRestorableDateTime": 1576622404.0,
      "LatestRestorableDateTime": 1576622404.0
    }
  }
}
```

Untuk informasi selengkapnya, lihat Pemulihan [Point-in-Time untuk DynamoDB](#) di [Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [UpdateContinuousBackups](#) di Referensi AWS CLI Perintah.

**update-contributor-insights**

Contoh kode berikut menunjukkan cara menggunakan `update-contributor-insights`.

**AWS CLI**

Untuk mengaktifkan Contributor Insights pada tabel

`update-contributor-insights` Contoh berikut memungkinkan Contributor Insights pada `MusicCollection` tabel dan indeks sekunder `AlbumTitle-index` global.

```
aws dynamodb update-contributor-insights \
  --table-name MusicCollection \
  --index-name AlbumTitle-index \
  --contributor-insights-action ENABLE
```

**Output:**

```
{
  "TableName": "MusicCollection",
  "IndexName": "AlbumTitle-index",
  "ContributorInsightsStatus": "ENABLING"
}
```



Untuk informasi selengkapnya, lihat [Menganalisis Akses Data Menggunakan Wawasan CloudWatch Kontributor untuk DynamoDB di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [UpdateContributorInsights](#) di Referensi AWS CLI Perintah.

## update-global-table-settings

Contoh kode berikut menunjukkan cara menggunakan `update-global-table-settings`.

### AWS CLI

Untuk memperbarui pengaturan kapasitas tulis yang disediakan pada tabel global DynamoDB

`update-global-table-settings` Contoh berikut menetapkan kapasitas tulis yang disediakan dari tabel `MusicCollection` global menjadi 15.

```
aws dynamodb update-global-table-settings \  
  --global-table-name MusicCollection \  
  --global-table-provisioned-write-capacity-units 15
```

### Output:

```
{  
  "GlobalTableName": "MusicCollection",  
  "ReplicaSettings": [  
    {  
      "RegionName": "eu-west-1",  
      "ReplicaStatus": "UPDATING",  
      "ReplicaProvisionedReadCapacityUnits": 10,  
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {  
        "AutoScalingDisabled": true  
      },  
      "ReplicaProvisionedWriteCapacityUnits": 10,  
      "ReplicaProvisionedWriteCapacityAutoScalingSettings": {  
        "AutoScalingDisabled": true  
      }  
    },  
    {  
      "RegionName": "us-east-1",  
      "ReplicaStatus": "UPDATING",  
      "ReplicaProvisionedReadCapacityUnits": 10,  
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {  
        "AutoScalingDisabled": true  
      }  
    }  
  ]  
}
```

```

    },
    "ReplicaProvisionedWriteCapacityUnits": 10,
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
      "AutoScalingDisabled": true
    }
  },
  {
    "RegionName": "us-east-2",
    "ReplicaStatus": "UPDATING",
    "ReplicaProvisionedReadCapacityUnits": 10,
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
      "AutoScalingDisabled": true
    },
    "ReplicaProvisionedWriteCapacityUnits": 10,
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
      "AutoScalingDisabled": true
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UpdateGlobalTableSettings](#) di Referensi AWS CLI Perintah.

## update-global-table

Contoh kode berikut menunjukkan cara menggunakan `update-global-table`.

### AWS CLI

Untuk memperbarui tabel global DynamoDB

`update-global-table` Contoh berikut menambahkan replika di Region tertentu ke tabel `MusicCollection` global.

```

aws dynamodb update-global-table \
  --global-table-name MusicCollection \
  --replica-updates Create={RegionName=eu-west-1}

```

Output:

```
{
  "GlobalTableDescription": {
    "ReplicationGroup": [
      {
        "RegionName": "eu-west-1"
      },
      {
        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-east-1"
      }
    ],
    "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/
MusicCollection",
    "CreationDateTime": 1576625818.532,
    "GlobalTableStatus": "ACTIVE",
    "GlobalTableName": "MusicCollection"
  }
}
```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UpdateGlobalTable](#) di Referensi AWS CLI Perintah.

## update-item

Contoh kode berikut menunjukkan cara menggunakan `update-item`.

### AWS CLI

Contoh 1: Untuk memperbarui item dalam tabel

Contoh `update-item` berikut memperbarui item dalam tabel `MusicCollection`. Ia menambahkan atribut baru (`Year`) dan memodifikasi `AlbumTitle` atribut. Semua atribut dalam item, seperti yang muncul setelah pembaruan, dikembalikan sebagai respons.

```
aws dynamodb update-item \
  --table-name MusicCollection \
  --key file://key.json \
  --update-expression "SET #Y = :y, #AT = :t" \
```

```
--expression-attribute-names file://expression-attribute-names.json \  
--expression-attribute-values file://expression-attribute-values.json \  
--return-values ALL_NEW \  
--return-consumed-capacity TOTAL \  
--return-item-collection-metrics SIZE
```

Isi dari key.json:

```
{  
  "Artist": {"S": "Acme Band"},  
  "SongTitle": {"S": "Happy Day"}  
}
```

Isi dari expression-attribute-names.json:

```
{  
  "#Y": "Year", "#AT": "AlbumTitle"  
}
```

Isi dari expression-attribute-values.json:

```
{  
  ":y":{"N": "2015"},  
  ":t":{"S": "Louder Than Ever"}  
}
```

Output:

```
{  
  "Attributes": {  
    "AlbumTitle": {  
      "S": "Louder Than Ever"  
    },  
    "Awards": {  
      "N": "10"  
    },  
    "Artist": {  
      "S": "Acme Band"  
    },  
    "Year": {  
      "N": "2015"  
    }  
  }  
}
```

```

    },
    "SongTitle": {
      "S": "Happy Day"
    }
  },
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 3.0
  },
  "ItemCollectionMetrics": {
    "ItemCollectionKey": {
      "Artist": {
        "S": "Acme Band"
      }
    },
    "SizeEstimateRangeGB": [
      0.0,
      1.0
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Menulis Item](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 2: Untuk memperbarui item secara kondisional

Contoh berikut memperbarui item dalam MusicCollection tabel, tetapi hanya jika item yang ada belum memiliki Year atribut.

```

aws dynamodb update-item \
  --table-name MusicCollection \
  --key file://key.json \
  --update-expression "SET #Y = :y, #AT = :t" \
  --expression-attribute-names file://expression-attribute-names.json \
  --expression-attribute-values file://expression-attribute-values.json \
  --condition-expression "attribute_not_exists(#Y)"

```

Isi dari key.json:

```

{
  "Artist": {"S": "Acme Band"},
  "SongTitle": {"S": "Happy Day"}
}

```

```
}
```

Isi dari `expression-attribute-names.json`:

```
{
  "#Y": "Year",
  "#AT": "AlbumTitle"
}
```

Isi dari `expression-attribute-values.json`:

```
{
  ":y": {"N": "2015"},
  ":t": {"S": "Louder Than Ever"}
}
```

Jika item sudah memiliki Year atribut, DynamoDB mengembalikan output berikut.

```
An error occurred (ConditionalCheckFailedException) when calling the UpdateItem
operation: The conditional request failed
```

Untuk informasi selengkapnya, lihat [Menulis Item](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UpdateItem](#) di Referensi AWS CLI Perintah.

## update-table-replica-auto-scaling

Contoh kode berikut menunjukkan cara menggunakan `update-table-replica-auto-scaling`.

### AWS CLI

Untuk memperbarui pengaturan penskalaan otomatis di seluruh replika tabel global

`update-table-replica-auto-scaling` Contoh berikut memperbarui pengaturan penskalaan otomatis kapasitas tulis di seluruh replika tabel global yang ditentukan.

```
aws dynamodb update-table-replica-auto-scaling \
  --table-name MusicCollection \
  --provisioned-write-capacity-auto-scaling-update file://auto-scaling-policy.json
```

Isi dari `auto-scaling-policy.json`:

```
{
  "MinimumUnits": 10,
  "MaximumUnits": 100,
  "AutoScalingDisabled": false,
  "ScalingPolicyUpdate": {
    "PolicyName": "DynamoDBWriteCapacityUtilization:table/MusicCollection",
    "TargetTrackingScalingPolicyConfiguration": {
      "TargetValue": 80
    }
  }
}
```

Output:

```
{
  "TableAutoScalingDescription": {
    "TableName": "MusicCollection",
    "TableStatus": "ACTIVE",
    "Replicas": [
      {
        "RegionName": "eu-central-1",
        "GlobalSecondaryIndexes": [],
        "ReplicaProvisionedReadCapacityAutoScalingSettings": {
          "MinimumUnits": 5,
          "MaximumUnits": 40000,
          "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
          "ScalingPolicies": [
            {
              "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
              "TargetTrackingScalingPolicyConfiguration": {
                "TargetValue": 70.0
              }
            }
          ]
        },
        "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
          "MinimumUnits": 10,
          "MaximumUnits": 100,
```

```

        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 80.0
                }
            }
        ],
        "ReplicaStatus": "ACTIVE"
    },
    {
        "RegionName": "us-east-1",
        "GlobalSecondaryIndexes": [],
        "ReplicaProvisionedReadCapacityAutoScalingSettings": {
            "MinimumUnits": 5,
            "MaximumUnits": 40000,
            "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
            "ScalingPolicies": [
                {
                    "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                    "TargetTrackingScalingPolicyConfiguration": {
                        "TargetValue": 70.0
                    }
                }
            ]
        },
        "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
            "MinimumUnits": 10,
            "MaximumUnits": 100,
            "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
            "ScalingPolicies": [
                {
                    "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",

```



```

        "TargetTrackingScalingPolicyConfiguration": {
            "TargetValue": 80.0
        }
    ],
    },
    "ReplicaStatus": "ACTIVE"
},
{
    "RegionName": "us-east-2",
    "GlobalSecondaryIndexes": [],
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ],
    },
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "MinimumUnits": 10,
        "MaximumUnits": 100,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 80.0
                }
            }
        ],
    },
    "ReplicaStatus": "ACTIVE"
}

```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [DynamoDB Global Tables](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UpdateTableReplicaAutoScaling](#) di Referensi AWS CLI Perintah.

## update-table

Contoh kode berikut menunjukkan cara menggunakan `update-table`.

### AWS CLI

Contoh 1: Untuk memodifikasi mode penagihan tabel

`update-table` Contoh berikut meningkatkan kapasitas baca dan tulis yang disediakan di atas meja. `MusicCollection`

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --billing-mode PROVISIONED \
  --provisioned-throughput ReadCapacityUnits=15,WriteCapacityUnits=10

```

Output:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ]
  }
}

```

```

    }
  ],
  "TableName": "MusicCollection",
  "KeySchema": [
    {
      "AttributeName": "Artist",
      "KeyType": "HASH"
    },
    {
      "AttributeName": "SongTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "UPDATING",
  "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
  "ProvisionedThroughput": {
    "LastIncreaseDateTime": "2020-07-28T13:18:18.921000-07:00",
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 15,
    "WriteCapacityUnits": 10
  },
  "TableSizeBytes": 182,
  "ItemCount": 2,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
  "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
  "BillingModeSummary": {
    "BillingMode": "PROVISIONED",
    "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
  }
}
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Tabel](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 2: Untuk membuat indeks sekunder global

Contoh berikut menambahkan indeks sekunder global ke MusicCollection tabel.

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --attribute-definitions AttributeName=AlbumTitle,AttributeType=S \
  --global-secondary-index-updates file://gsi-updates.json

```

Isi dari `gsi-updates.json`:

```
[
  {
    "Create": {
      "IndexName": "AlbumTitle-index",
      "KeySchema": [
        {
          "AttributeName": "AlbumTitle",
          "KeyType": "HASH"
        }
      ],
      "ProvisionedThroughput": {
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 10
      },
      "Projection": {
        "ProjectionType": "ALL"
      }
    }
  }
]
```

Output:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
```

```
{
  "AttributeName": "Artist",
  "KeyType": "HASH"
},
{
  "AttributeName": "SongTitle",
  "KeyType": "RANGE"
}
],
"TableStatus": "UPDATING",
"CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
"ProvisionedThroughput": {
  "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
  "NumberOfDecreasesToday": 0,
  "ReadCapacityUnits": 15,
  "WriteCapacityUnits": 10
},
"TableSizeBytes": 182,
"ItemCount": 2,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
"TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
"BillingModeSummary": {
  "BillingMode": "PROVISIONED",
  "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
},
"GlobalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitle-index",
    "KeySchema": [
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "HASH"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "IndexStatus": "CREATING",
    "Backfilling": false,
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 10
    }
  },

```

```

        "IndexSizeBytes": 0,
        "ItemCount": 0,
        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
    }
]
}
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Tabel](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 3: Untuk mengaktifkan DynamoDB Streams di atas meja

Perintah berikut memungkinkan DynamoDB Streams di atas meja. MusicCollection

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --stream-specification StreamEnabled=true,StreamViewType=NEW_IMAGE

```

Output:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      }
    ]
  }
}

```

```
    },
    {
      "AttributeName": "SongTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "UPDATING",
  "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
  "ProvisionedThroughput": {
    "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 15,
    "WriteCapacityUnits": 10
  },
  "TableSizeBytes": 182,
  "ItemCount": 2,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
  "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
  "BillingModeSummary": {
    "BillingMode": "PROVISIONED",
    "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
  },
  "LocalSecondaryIndexes": [
    {
      "IndexName": "AlbumTitleIndex",
      "KeySchema": [
        {
          "AttributeName": "Artist",
          "KeyType": "HASH"
        },
        {
          "AttributeName": "AlbumTitle",
          "KeyType": "RANGE"
        }
      ],
      "Projection": {
        "ProjectionType": "INCLUDE",
        "NonKeyAttributes": [
          "Year",
          "Genre"
        ]
      }
    },
    {
      "IndexSizeBytes": 139,
      "ItemCount": 2,
```

```

        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
    }
  ],
  "GlobalSecondaryIndexes": [
    {
      "IndexName": "AlbumTitle-index",
      "KeySchema": [
        {
          "AttributeName": "AlbumTitle",
          "KeyType": "HASH"
        }
      ],
      "Projection": {
        "ProjectionType": "ALL"
      },
      "IndexStatus": "ACTIVE",
      "ProvisionedThroughput": {
        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 10
      },
      "IndexSizeBytes": 0,
      "ItemCount": 0,
      "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
    }
  ],
  "StreamSpecification": {
    "StreamEnabled": true,
    "StreamViewType": "NEW_IMAGE"
  },
  "LatestStreamLabel": "2020-07-28T21:53:39.112",
  "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/stream/2020-07-28T21:53:39.112"
}
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Tabel](#) di Panduan Pengembang Amazon DynamoDB.

Contoh 4: Untuk mengaktifkan enkripsi sisi server

Contoh berikut memungkinkan enkripsi sisi server di atas meja. `MusicCollection`



```
aws dynamodb update-table \  
  --table-name MusicCollection \  
  --sse-specification Enabled=true,SSEType=KMS
```

**Output:**

```
{  
  "TableDescription": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "AlbumTitle",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "Artist",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "AttributeType": "S"  
      }  
    ],  
    "TableName": "MusicCollection",  
    "KeySchema": [  
      {  
        "AttributeName": "Artist",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "TableStatus": "ACTIVE",  
    "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",  
    "ProvisionedThroughput": {  
      "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",  
      "NumberOfDecreasesToday": 0,  
      "ReadCapacityUnits": 15,  
      "WriteCapacityUnits": 10  
    },  
    "TableSizeBytes": 182,  
    "ItemCount": 2,  
  }  
}
```

```
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
"TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
"BillingModeSummary": {
  "BillingMode": "PROVISIONED",
  "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
},
"LocalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitleIndex",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "INCLUDE",
      "NonKeyAttributes": [
        "Year",
        "Genre"
      ]
    },
    "IndexSizeBytes": 139,
    "ItemCount": 2,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
  }
],
"GlobalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitle-index",
    "KeySchema": [
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "HASH"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    }
  }
],
```

```

        "IndexStatus": "ACTIVE",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 10
        },
        "IndexSizeBytes": 0,
        "ItemCount": 0,
        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
    }
],
"StreamSpecification": {
    "StreamEnabled": true,
    "StreamViewType": "NEW_IMAGE"
},
"LatestStreamLabel": "2020-07-28T21:53:39.112",
"LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/stream/2020-07-28T21:53:39.112",
"SSEDescription": {
    "Status": "UPDATING"
}
}
}
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Tabel](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UpdateTable](#) di Referensi AWS CLI Perintah.

## update-time-to-live

Contoh kode berikut menunjukkan cara menggunakan `update-time-to-live`.

### AWS CLI

Untuk memperbarui pengaturan Waktu ke Langsung di atas meja

`update-time-to-live` Contoh berikut memungkinkan Time to Live pada tabel yang ditentukan.

```

aws dynamodb update-time-to-live \
  --table-name MusicCollection \
  --time-to-live-specification Enabled=true,AttributeName=ttl

```

## Output:

```
{
  "TimeToLiveSpecification": {
    "Enabled": true,
    "AttributeName": "ttl"
  }
}
```

Untuk informasi selengkapnya, lihat [Waktu untuk Hidup](#) di Panduan Pengembang Amazon DynamoDB.

- Untuk detail API, lihat [UpdateTimeToLive](#) di Referensi AWS CLI Perintah.

## Contoh DynamoDB Streams menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan DynamoDB Streams AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **describe-stream**

Contoh kode berikut menunjukkan cara menggunakan `describe-stream`.

### AWS CLI

Untuk mendapatkan informasi tentang aliran DynamoDB

`describe-stream` Perintah berikut menampilkan informasi tentang aliran DynamoDB tertentu.

```
aws dynamodbstreams describe-stream \  
  --stream-arn arn:aws:dynamodb:us-west-1:123456789012:table/Music/  
stream/2019-10-22T18:02:01.576
```

Output:

```
{  
  "StreamDescription": {  
    "StreamArn": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/  
stream/2019-10-22T18:02:01.576",  
    "StreamLabel": "2019-10-22T18:02:01.576",  
    "StreamStatus": "ENABLED",  
    "StreamViewType": "NEW_AND_OLD_IMAGES",  
    "CreationRequestDateTime": 1571767321.571,  
    "TableName": "Music",  
    "KeySchema": [  
      {  
        "AttributeName": "Artist",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "Shards": [  
      {  
        "ShardId": "shardId-00000001571767321804-697ce3d2",  
        "SequenceNumberRange": {  
          "StartingSequenceNumber": "4000000000000642977831",  
          "EndingSequenceNumber": "4000000000000642977831"  
        }  
      },  
      {  
        "ShardId": "shardId-00000001571780995058-40810d86",  
        "SequenceNumberRange": {  
          "StartingSequenceNumber": "75740000000005655171150"  
        },  
        "ParentShardId": "shardId-00000001571767321804-697ce3d2"  
      }  
    ]  
  }  
}
```

```
}
}
```

Untuk informasi selengkapnya, lihat [Menangkap Aktivitas Tabel dengan DynamoDB Streams di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [DescribeStream](#) di Referensi AWS CLI Perintah.

## get-records

Contoh kode berikut menunjukkan cara menggunakan `get-records`.

### AWS CLI

Untuk mendapatkan catatan dari aliran Dynamodb

`get-records` Perintah berikut mengambil catatan menggunakan iterator pecahan Amazon DynamoDB yang ditentukan.

```
aws dynamodbstreams get-records \
  --shard-iterator "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576|1|
AAAAAAAAAAGgM3YZ89vLZZxjmoQeo33r9M4x3+zmmTLsiL86MfrF4+B4EbsByi52InVmi0Nmy6xVW4IRcIIbs1z07MNI
+CjNP1qQjnyRSAnf0wWmKhL1/KNParWSfz2odf780o00bIDIWRRMkt7+Hyzh9SD
+hFxFAWR5C7QI10XPc8mRBfNIazfrVCjJK8/jsjCzsqNyXKzJbhh+GXCoxYN
+Kpmg4nyj1EAsYhbGL35muvHFoHjcyuynbsczbWaXNfThDwRAYvoTmc8XhHKtAWUbJiaVd8ZPtQwDsThCrmDRPI dmTRG
+w/1EGS05ha1qNP+V14+tuhz2TRnhnJo/pny9GI/yGpce97mWvSPr5KPwy+DtcM5BHayBs
+PVYHITaTliInFlT
+LCwvaz1QH3MY3b8A05Z800wjpktm60iQqtMeDwN4NX6FrcxR34JoFKGsgR8XkHVJzz2xr1xqSJ12ycpNTyHnndusw==
```

Output:

```
{
  "Records": [
    {
      "eventID": "c3b5d798eef6215d42f8137b19a88e50",
      "eventName": "INSERT",
      "eventVersion": "1.1",
      "eventSource": "aws:dynamodb",
      "awsRegion": "us-west-1",
      "dynamodb": {
        "ApproximateCreationDateTime": 1571849028.0,
        "Keys": {
```

```
        "Artist": {
            "S": "No One You Know"
        },
        "SongTitle": {
            "S": "Call Me Today"
        }
    },
    "NewImage": {
        "AlbumTitle": {
            "S": "Somewhat Famous"
        },
        "Artist": {
            "S": "No One You Know"
        },
        "Awards": {
            "N": "1"
        },
        "SongTitle": {
            "S": "Call Me Today"
        }
    },
    "SequenceNumber": "700000000013256296913",
    "SizeBytes": 119,
    "StreamViewType": "NEW_AND_OLD_IMAGES"
}
},
{
    "eventID": "878960a6967867e2da16b27380a27328",
    "eventName": "INSERT",
    "eventVersion": "1.1",
    "eventSource": "aws:dynamodb",
    "awsRegion": "us-west-1",
    "dynamodb": {
        "ApproximateCreationDateTime": 1571849029.0,
        "Keys": {
            "Artist": {
                "S": "Acme Band"
            },
            "SongTitle": {
                "S": "Happy Day"
            }
        }
    },
    "NewImage": {
        "AlbumTitle": {
```

```
        "S": "Songs About Life"
      },
      "Artist": {
        "S": "Acme Band"
      },
      "Awards": {
        "N": "10"
      },
      "SongTitle": {
        "S": "Happy Day"
      }
    },
    "SequenceNumber": "800000000013256297217",
    "SizeBytes": 100,
    "StreamViewType": "NEW_AND_OLD_IMAGES"
  }
},
{
  "eventID": "520fabde080e159fc3710b15ee1d4daa",
  "eventName": "MODIFY",
  "eventVersion": "1.1",
  "eventSource": "aws:dynamodb",
  "awsRegion": "us-west-1",
  "dynamodb": {
    "ApproximateCreationDateTime": 1571849734.0,
    "Keys": {
      "Artist": {
        "S": "Acme Band"
      },
      "SongTitle": {
        "S": "Happy Day"
      }
    },
    "NewImage": {
      "AlbumTitle": {
        "S": "Updated Album Title"
      },
      "Artist": {
        "S": "Acme Band"
      },
      "Awards": {
        "N": "10"
      },
      "SongTitle": {
```



```

        "S": "Happy Day"
      }
    },
    "OldImage": {
      "AlbumTitle": {
        "S": "Songs About Life"
      },
      "Artist": {
        "S": "Acme Band"
      },
      "Awards": {
        "N": "10"
      },
      "SongTitle": {
        "S": "Happy Day"
      }
    },
    "SequenceNumber": "900000000013256687845",
    "SizeBytes": 170,
    "StreamViewType": "NEW_AND_OLD_IMAGES"
  }
},
  ],
  "NextShardIterator": "arn:aws:dynamodb:us-west-1:123456789012:table/
Music/stream/2019-10-23T16:41:08.740|1|AAAAAAAAAAAEhEI04jkFLW
+LK0wivjT8d/IHEh3iExV2xK00aTxEzVy1C1C7Kbb5+Z0W6bT9VQ2n1/
mrs7+PRia0ZCHJu7JHJVW7zlsq0i/ges3fw8GYEymyL+piEk35cx67rQqwKKyq
+Q6w9JyjreIOj4F2lWLV26lBwRTrIYC4IB7C3BZZK4715QwYdDxNdVHiSBRZX8UqoS6W0t0F87xZLNB9F/
NhYBLXi/wcGvAcBcC0TNI0H+N0Nqwt0B/
FGcKNrf8YZ0xRoNN6RgGuVWHF3px0hxEJeFZoSoJTIKeG9YcYxzi5Ci/
mhdtm7tBXnbw5c6xmsGsBqTirNj1DyJLcWl8C10UOLX63Ufo/5QliztcjEbKsQe28x8LM8o7VH1Is0fF/
ITt8awSA4igyJS0P87GN8Qri8kj8iaE35805jBHF2wvwT6Iy2xGrR2r2HzYps9dwG0arVdEITaJfWzNoL4HajMhmREZ
+V04i1YIeHMXJfcwetNRuIbdQXfJht2NQZa4PVV6iknY6d19MrdbSTMKoqAuvp6g3Q2jH4t7GKCLWgodcPAn8g5+43Da
}

```

Untuk informasi selengkapnya, lihat [Menangkap Aktivitas Tabel dengan DynamoDB Streams di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [GetRecords](#) di Referensi AWS CLI Perintah.

## get-shard-iterator

Contoh kode berikut menunjukkan cara menggunakan `get-shard-iterator`.

## AWS CLI

Untuk mendapatkan iterator pecahan

`get-shard-iterator` Perintah berikut mengambil iterator shard untuk shard tertentu.

```
aws dynamodbstreams get-shard-iterator \
  --stream-arn arn:aws:dynamodb:us-west-1:12356789012:table/Music/
stream/2019-10-22T18:02:01.576 \
  --shard-id shardId-00000001571780995058-40810d86 \
  --shard-iterator-type LATEST
```

Output:

```
{
  "ShardIterator": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576|1|
AAAAAAAAAAAGgM3YZ89vLZZxjmoQeo33r9M4x3+zmmTLsiL86MfrF4+B4EbsByi52InVmi0Nmy6xVW4IRcIIbs1z07MNI
+CjNP1qQjnyRSAnf0wWmKhL1/KNParWSfz2odf780o00bIDIWRRMkt7+Hyzh9SD
+hFxFAWR5C7QI10XPc8mRBfNIazfrVCjJK8/jsjCzsqNyXKzJbhh+GXCoxYN
+Kpmg4nyj1EAsYhbGL35muvHFoHjcyuynbsczbWaXNfThDwRAyvoTmc8XhHKtAWUbJiaVd8ZPtQwDsThCrmDRPI dmTRG
+w/1EGS05ha1qNP+Vl4+tuhz2TRnhnJo/pny9GI/yGpce97mWvSPr5KPwy+Dtcm5BHayBs
+PVYHITaTliInFlT
+LCwvaz1QH3MY3b8A05Z800wjpktm60iQqtMeDwN4NX6FrcxR34JoFKGsgR8XkHVJzz2xr1xqSJ12ycpNTyHndusw==
}
```

Untuk informasi selengkapnya, lihat [Menangkap Aktivitas Tabel dengan DynamoDB Streams di Panduan Pengembangan Amazon DynamoDB](#).

- Untuk detail API, lihat [GetShardIterator](#) di Referensi AWS CLI Perintah.

## list-streams

Contoh kode berikut menunjukkan cara menggunakan `list-streams`.

## AWS CLI

Untuk daftar aliran DynamoDB

`list-streams` Perintah berikut mencantumkan semua aliran Amazon DynamoDB yang ada dalam Wilayah default. AWS

```
aws dynamodbstreams list-streams
```

Output:

```
{
  "Streams": [
    {
      "StreamArn": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/stream/2019-10-22T18:02:01.576",
      "TableName": "Music",
      "StreamLabel": "2019-10-22T18:02:01.576"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menangkap Aktivitas Tabel dengan DynamoDB Streams di Panduan Pengembang Amazon DynamoDB](#).

- Untuk detail API, lihat [ListStreams](#) di Referensi AWS CLI Perintah.

## Contoh Amazon EC2 menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon EC2. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

## Tindakan

### **accept-address-transfer**

Contoh kode berikut menunjukkan cara menggunakan `accept-address-transfer`.

#### AWS CLI

Untuk menerima alamat IP Elastis yang ditransfer ke akun Anda

`accept-address-transfer` Contoh berikut menerima transfer alamat IP Elastis yang ditentukan ke akun Anda.

```
aws ec2 accept-address-transfer \
  --address 100.21.184.216
```

#### Output:

```
{
  "AddressTransfer": {
    "PublicIp": "100.21.184.216",
    "AllocationId": "eipalloc-09ad461b0d03f6aaf",
    "TransferAccountId": "123456789012",
    "TransferOfferExpirationTimestamp": "2023-02-22T20:51:10.000Z",
    "TransferOfferAcceptedTimestamp": "2023-02-22T22:52:54.000Z",
    "AddressTransferStatus": "accepted"
  }
}
```

Untuk informasi selengkapnya, lihat [Mentransfer alamat IP Elastis](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [AcceptAddressTransfer](#) di Referensi AWS CLI Perintah.

### **accept-reserved-instances-exchange-quote**

Contoh kode berikut menunjukkan cara menggunakan `accept-reserved-instances-exchange-quote`.

#### AWS CLI

Untuk melakukan pertukaran Instans Cadangan Konvertibel

Contoh ini melakukan pertukaran Instans Cadangan Konvertibel yang ditentukan.

Perintah:

```
aws ec2 accept-reserved-instances-exchange-quote --reserved-instance-ids 7b8750c3-397e-4da4-bbcb-a45ebexample --target-configurations OfferingId=b747b472-423c-48f3-8cee-679bcexample
```

Output:

```
{
  "ExchangeId": "riex-e68ed3c1-8bc8-4c17-af77-811afexample"
}
```

- Untuk detail API, lihat [AcceptReservedInstancesExchangeQuote](#) di Referensi AWS CLI Perintah.

## accept-transit-gateway-peering-attachment

Contoh kode berikut menunjukkan cara menggunakan `accept-transit-gateway-peering-attachment`.

AWS CLI

Untuk menerima lampiran peering gateway transit

`accept-transit-gateway-peering-attachment` Contoh berikut menerima lampiran peering gateway transit yang ditentukan. `--region` Parameter menentukan Wilayah tempat gateway transit penerima berada.

```
aws ec2 accept-transit-gateway-peering-attachment \
  --transit-gateway-attachment-id tgw-attach-4455667788aabbccd \
  --region us-east-2
```

Output:

```
{
  "TransitGatewayPeeringAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
```

```

    "RequesterTgwInfo": {
      "TransitGatewayId": "tgw-123abc05e04123abc",
      "OwnerId": "123456789012",
      "Region": "us-west-2"
    },
    "AcceptorTgwInfo": {
      "TransitGatewayId": "tgw-11223344aabbcc112",
      "OwnerId": "123456789012",
      "Region": "us-east-2"
    },
    "State": "pending",
    "CreationTime": "2019-12-09T11:38:31.000Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Lampiran Pengintip Transit Gateway](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [AcceptTransitGatewayPeeringAttachment](#) di Referensi AWS CLI Perintah.

## accept-transit-gateway-vpc-attachment

Contoh kode berikut menunjukkan cara menggunakan `accept-transit-gateway-vpc-attachment`.

### AWS CLI

Untuk menerima permintaan untuk melampirkan VPC ke gateway transit.

`accept-transit-gateway-vpc-attachment` Contoh berikut menerima permintaan forte lampiran yang ditentukan.

```

aws ec2 accept-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE

```

### Output:

```

{
  "TransitGatewayVpcAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "VpcId": "vpc-07e8ffd50fEXAMPLE",
  }
}

```

```
    "VpcOwnerId": "123456789012",
    "State": "pending",
    "SubnetIds": [
      "subnet-0752213d59EXAMPLE"
    ],
    "CreationTime": "2019-07-10T17:33:46.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Lampiran Gateway Transit ke VPC](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [AcceptTransitGatewayVpcAttachment](#) di Referensi AWS CLI Perintah.

## accept-vpc-endpoint-connections

Contoh kode berikut menunjukkan cara menggunakan `accept-vpc-endpoint-connections`.

### AWS CLI

Untuk menerima permintaan koneksi titik akhir antarmuka

Contoh ini menerima permintaan koneksi endpoint yang ditentukan untuk layanan endpoint yang ditentukan.

Perintah:

```
aws ec2 accept-vpc-endpoint-connections --service-id vpce-svc-03d5ebb7d9579a2b3 --
vpc-endpoint-ids vpce-0c1308d7312217abc
```

Output:

```
{
  "Unsuccessful": []
}
```

- Untuk detail API, lihat [AcceptVpcEndpointConnections](#) di Referensi AWS CLI Perintah.

## accept-vpc-peering-connection

Contoh kode berikut menunjukkan cara menggunakan `accept-vpc-peering-connection`.

### AWS CLI

Untuk menerima koneksi peering VPC

Contoh ini menerima permintaan koneksi peering VPC yang ditentukan.

Perintah:

```
aws ec2 accept-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Output:

```
{
  "VpcPeeringConnection": {
    "Status": {
      "Message": "Provisioning",
      "Code": "provisioning"
    },
    "Tags": [],
    "AccepterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-44455566",
      "CidrBlock": "10.0.1.0/28"
    },
    "VpcPeeringConnectionId": "pcx-1a2b3c4d",
    "RequesterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-111abc45",
      "CidrBlock": "10.0.0.0/28"
    }
  }
}
```

- Untuk detail API, lihat [AcceptVpcPeeringConnection](#) di Referensi AWS CLI Perintah.

## advertise-byoip-cidr

Contoh kode berikut menunjukkan cara menggunakan `advertise-byoip-cidr`.



## AWS CLI

Untuk mengiklankan rentang alamat

`advertise-byoip-cidr` Contoh berikut mengiklankan rentang alamat IPv4 publik yang ditentukan.

```
aws ec2 advertise-byoip-cidr \  
  --cidr 203.0.113.25/24
```

Output:

```
{  
  "ByoipCidr": {  
    "Cidr": "203.0.113.25/24",  
    "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",  
    "State": "provisioned"  
  }  
}
```

- Untuk detail API, lihat [AdvertiseByoipCidr](#) di Referensi AWS CLI Perintah.

## `allocate-address`

Contoh kode berikut menunjukkan cara menggunakan `allocate-address`.

## AWS CLI

Contoh 1: Untuk mengalokasikan alamat IP Elastis dari kolam alamat Amazon

Contoh `allocate-address` berikut mengalokasikan alamat IP Elastis. Amazon EC2 memilih alamat dari kolam alamat Amazon.

```
aws ec2 allocate-address
```

Output:

```
{  
  "PublicIp": "70.224.234.241",  
  "AllocationId": "eipalloc-01435ba59eEXAMPLE",  
  "PublicIpv4Pool": "amazon",
```

```
"NetworkBorderGroup": "us-west-2",
"Domain": "vpc"
}
```

Untuk informasi selengkapnya, lihat [Alamat IP Elastis](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk mengalokasikan alamat IP Elastis dan mengaitkannya dengan grup batas jaringan

Contoh `allocate-address` berikut mengalokasikan alamat IP Elastis dan mengaitkannya dengan grup batas jaringan tertentu.

```
aws ec2 allocate-address \
  --network-border-group us-west-2-lax-1
```

Output:

```
{
  "PublicIp": "70.224.234.241",
  "AllocationId": "eipalloc-e03dd489ceEXAMPLE",
  "PublicIpv4Pool": "amazon",
  "NetworkBorderGroup": "us-west-2-lax-1",
  "Domain": "vpc"
}
```

Untuk informasi selengkapnya, lihat [Alamat IP Elastis](#) di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk mengalokasikan alamat IP Elastis dari kolam alamat milik Anda

Contoh `allocate-address` berikut mengalokasikan alamat IP Elastis dari kolam alamat yang Anda bawa ke akun Amazon Web Services. Amazon EC2 memilih alamat dari kolam alamat.

```
aws ec2 allocate-address \
  --public-ipv4-pool ipv4pool-ec2-1234567890abcdef0
```

Output:

```
{
  "AllocationId": "eipalloc-02463d08ceEXAMPLE",
  "NetworkBorderGroup": "us-west-2",
  "CustomerOwnedIp": "18.218.95.81",
}
```

```
"CustomerOwnedIpv4Pool": "ipv4pool-ec2-1234567890abcdef0",
"Domain": "vpc"
"NetworkBorderGroup": "us-west-2",
}
```

Untuk informasi selengkapnya, lihat [Alamat IP Elastis](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [AllocateAddress](#) di Referensi AWS CLI Perintah.

## allocate-hosts

Contoh kode berikut menunjukkan cara menggunakan `allocate-hosts`.

### AWS CLI

Contoh 1: Untuk mengalokasikan Host Khusus

`allocate-hosts` Contoh berikut mengalokasikan satu Host Khusus di `eu-west-1a` Availability Zone, tempat Anda dapat meluncurkan `m5.large` instance. Secara default, Host Khusus hanya menerima peluncuran instans target, dan tidak mendukung pemulihan host.

```
aws ec2 allocate-hosts \
  --instance-type m5.large \
  --availability-zone eu-west-1a \
  --quantity 1
```

Output:

```
{
  "HostIds": [
    "h-07879acf49EXAMPLE"
  ]
}
```

Contoh 2: Untuk mengalokasikan Host Khusus dengan penempatan otomatis dan pemulihan host diaktifkan

`allocate-hosts` Contoh berikut mengalokasikan satu Host Khusus di `eu-west-1a` Availability Zone dengan penempatan otomatis dan pemulihan host diaktifkan.

```
aws ec2 allocate-hosts \
  --instance-type m5.large \
```

```
--availability-zone eu-west-1a \  
--auto-placement on \  
--host-recovery on \  
--quantity 1
```

Output:

```
{  
  "HostIds": [  
    "h-07879acf49EXAMPLE"  
  ]  
}
```

Contoh 3: Untuk mengalokasikan Host Khusus dengan tag

`allocate-hosts` Contoh berikut mengalokasikan satu Host Khusus dan menerapkan tag dengan kunci bernama `purpose` dan nilai `production`

```
aws ec2 allocate-hosts \  
  --instance-type m5.large \  
  --availability-zone eu-west-1a \  
  --quantity 1 \  
  --tag-specifications 'ResourceType=dedicated-  
host,Tags={Key=purpose,Value=production}'
```

Output:

```
{  
  "HostIds": [  
    "h-07879acf49EXAMPLE"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengalokasikan Host Khusus](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [AllocateHosts](#) di Referensi AWS CLI Perintah.

## **allocate-ipam-pool-cidr**

Contoh kode berikut menunjukkan cara menggunakan `allocate-ipam-pool-cidr`.

## AWS CLI

Untuk mengalokasikan CIDR dari kolam IPAM

`allocate-ipam-pool-cidr` Contoh berikut mengalokasikan CIDR dari kolam IPAM.

(Linux):

```
aws ec2 allocate-ipam-pool-cidr \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --netmask-length 24
```

(Windows):

```
aws ec2 allocate-ipam-pool-cidr ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --netmask-length 24
```

Output:

```
{  
  "IpamPoolAllocation": {  
    "Cidr": "10.0.0.0/24",  
    "IpamPoolAllocationId": "ipam-pool-alloc-018ecc28043b54ba38e2cd99943cebfb",  
    "ResourceType": "custom",  
    "ResourceOwner": "123456789012"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengalokasikan CIDR secara manual ke kumpulan untuk memesan ruang alamat IP di Panduan Pengguna Amazon VPC IPAM](#).

- Untuk detail API, lihat [AllocateIpamPoolCidr](#) di Referensi AWS CLI Perintah.

## **apply-security-groups-to-client-vpn-target-network**

Contoh kode berikut menunjukkan cara menggunakan `apply-security-groups-to-client-vpn-target-network`.

## AWS CLI

Untuk menerapkan grup keamanan ke jaringan target untuk titik akhir Client VPN

`apply-security-groups-to-client-vpn-target-network` Contoh berikut menerapkan grup keamanan `sg-01f6e627a89f4db32` untuk asosiasi antara jaringan target yang ditentukan dan titik akhir Client VPN.

```
aws ec2 apply-security-groups-to-client-vpn-target-network \  
  --security-group-ids sg-01f6e627a89f4db32 \  
  --vpc-id vpc-0e2110c2f324332e0 \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{  
  "SecurityGroupIds": [  
    "sg-01f6e627a89f4db32"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Jaringan Target](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [ApplySecurityGroupsToClientVpnTargetNetwork](#) di Referensi AWS CLI Perintah.

## **assign-ipv6-addresses**

Contoh kode berikut menunjukkan cara menggunakan `assign-ipv6-addresses`.

AWS CLI

Untuk menetapkan alamat IPv6 tertentu ke antarmuka jaringan

Contoh ini menetapkan alamat IPv6 yang ditentukan ke antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 assign-ipv6-addresses --network-interface-id eni-38664473 --ipv6-addresses  
  2001:db8:1234:1a00:3304:8879:34cf:4071 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Output:

```
{  
  "AssignedIpv6Addresses": [  
    "2001:db8:1234:1a00:3304:8879:34cf:4071"  
    "2001:db8:1234:1a00:9691:9503:25ad:1761"  
  ]  
}
```

```
"2001:db8:1234:1a00:3304:8879:34cf:4071",  
"2001:db8:1234:1a00:9691:9503:25ad:1761"  
],  
"NetworkInterfaceId": "eni-38664473"  
}
```

Untuk menetapkan alamat IPv6 yang dipilih Amazon ke antarmuka jaringan

Contoh ini memberikan dua alamat IPv6 ke antarmuka jaringan yang ditentukan. Amazon secara otomatis menetapkan alamat IPv6 ini dari alamat IPv6 yang tersedia di rentang blok IPv6 CIDR subnet.

Perintah:

```
aws ec2 assign-ipv6-addresses --network-interface-id eni-38664473 --ipv6-address-  
count 2
```

Output:

```
{  
  "AssignedIpv6Addresses": [  
    "2001:db8:1234:1a00:3304:8879:34cf:4071",  
    "2001:db8:1234:1a00:9691:9503:25ad:1761"  
  ],  
  "NetworkInterfaceId": "eni-38664473"  
}
```

- Untuk detail API, lihat [AssignIpv6Alamat](#) di Referensi AWS CLI Perintah.

## assign-private-ip-addresses

Contoh kode berikut menunjukkan cara menggunakan `assign-private-ip-addresses`.

### AWS CLI

Untuk menetapkan alamat IP pribadi sekunder tertentu antarmuka jaringan

Contoh ini memberikan alamat IP pribadi sekunder yang ditentukan ke antarmuka jaringan yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 assign-private-ip-addresses --network-interface-id eni-e5aa89a3 --private-  
ip-addresses 10.0.0.82
```

Untuk menetapkan alamat IP pribadi sekunder yang Amazon EC2 pilih ke antarmuka jaringan

Contoh ini memberikan dua alamat IP pribadi sekunder ke antarmuka jaringan yang ditentukan. Amazon EC2 secara otomatis menetapkan alamat IP ini dari alamat IP yang tersedia dalam rentang blok CIDR dari subnet yang terkait dengan antarmuka jaringan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 assign-private-ip-addresses --network-interface-id eni-e5aa89a3 --secondary-  
private-ip-address-count 2
```

- Untuk detail API, lihat [AssignPrivateIpAddresses](#) di Referensi AWS CLI Perintah.

## assign-private-nat-gateway-address

Contoh kode berikut menunjukkan cara menggunakan `assign-private-nat-gateway-address`.

### AWS CLI

Untuk menetapkan alamat IP pribadi ke gateway NAT pribadi Anda

`assign-private-nat-gateway-address` Contoh berikut memberikan dua alamat IP pribadi ke gateway NAT pribadi yang ditentukan.

```
aws ec2 assign-private-nat-gateway-address \  
  --nat-gateway-id nat-1234567890abcdef0 \  
  --private-ip-address-count 2
```

Output:

```
{  
  "NatGatewayId": "nat-1234567890abcdef0",  
  "NatGatewayAddresses": [  
    {  
      "NetworkInterfaceId": "eni-0065a61b324d1897a",  
      "IsPrimary": false,  
      "Status": "assigning"  
    }  
  ]  
}
```



```
    },
    {
      "NetworkInterfaceId": "eni-0065a61b324d1897a",
      "IsPrimary": false,
      "Status": "assigning"
    }
  ]
}
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [AssignPrivateNatGatewayAddress](#) di Referensi AWS CLI Perintah.

## associate-address

Contoh kode berikut menunjukkan cara menggunakan `associate-address`.

### AWS CLI

Untuk mengaitkan alamat IP Elastis di EC2-Classic

Contoh ini mengaitkan alamat IP Elastis dengan instans di EC2-Classic. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 associate-address --instance-id i-07ffe74c7330ebf53 --public-ip 198.51.100.0
```

Untuk mengaitkan alamat IP Elastis di EC2-VPC

Contoh ini mengaitkan alamat IP Elastis dengan instans di VPC.

Perintah:

```
aws ec2 associate-address --instance-id i-0b263919b6498b123 --allocation-id eipalloc-64d5890a
```

Output:

```
{
  "AssociationId": "eipassoc-2bebb745"
}
```

Contoh ini mengaitkan alamat IP Elastis dengan antarmuka jaringan.

Perintah:

```
aws ec2 associate-address --allocation-id eipalloc-64d5890a --network-interface-id
eni-1a2b3c4d
```

Contoh ini mengaitkan IP Elastis dengan alamat IP privat yang terkait dengan antarmuka jaringan.

Perintah:

```
aws ec2 associate-address --allocation-id eipalloc-64d5890a --network-interface-id
eni-1a2b3c4d --private-ip-address 10.0.0.85
```

- Untuk detail API, lihat [AssociateAddress](#) di Referensi AWS CLI Perintah.

## **associate-client-vpn-target-network**

Contoh kode berikut menunjukkan cara menggunakan `associate-client-vpn-target-network`.

AWS CLI

Untuk mengaitkan jaringan target dengan titik akhir Client VPN

`associate-client-vpn-target-network` Contoh berikut mengaitkan subnet dengan titik akhir Client VPN yang ditentukan.

```
aws ec2 associate-client-vpn-target-network \
  --subnet-id subnet-0123456789abcabca \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
  "AssociationId": "cvpn-assoc-12312312312312312",
  "Status": {
    "Code": "associating"
  }
}
```

Untuk informasi selengkapnya, lihat [Jaringan Target](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [AssociateClientVpnTargetNetwork](#) di Referensi AWS CLI Perintah.

## **associate-dhcp-options**

Contoh kode berikut menunjukkan cara menggunakan `associate-dhcp-options`.

### AWS CLI

Untuk mengaitkan opsi DHCP yang disetel dengan VPC Anda

Contoh ini mengaitkan opsi DHCP tertentu yang ditetapkan dengan VPC yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 associate-dhcp-options --dhcp-options-id dopt-d9070ebb --vpc-id vpc-a01106c2
```

Untuk mengaitkan opsi DHCP default yang ditetapkan dengan VPC Anda

Contoh ini mengaitkan opsi DHCP default yang ditetapkan dengan VPC yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 associate-dhcp-options --dhcp-options-id default --vpc-id vpc-a01106c2
```

- Untuk detail API, lihat [AssociateDhcpOptions](#) di Referensi AWS CLI Perintah.

## **associate-iam-instance-profile**

Contoh kode berikut menunjukkan cara menggunakan `associate-iam-instance-profile`.

### AWS CLI

Untuk mengaitkan profil instans IAM dengan instance

Contoh ini mengaitkan profil instans IAM bernama `admin-role` dengan instance.  
`i-123456789abcde123`

Perintah:

```
aws ec2 associate-iam-instance-profile --instance-id i-123456789abcde123 --iam-
instance-profile Name=admin-role
```

Output:

```
{
  "IamInstanceProfileAssociation": {
    "InstanceId": "i-123456789abcde123",
    "State": "associating",
    "AssociationId": "iip-assoc-0e7736511a163c209",
    "IamInstanceProfile": {
      "Id": "AIPAJBLK7RKJKWDXVHIEC",
      "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
    }
  }
}
```

- Untuk detail API, lihat [AssociateIamInstanceProfile](#) di Referensi AWS CLI Perintah.

## associate-instance-event-window

Contoh kode berikut menunjukkan cara menggunakan `associate-instance-event-window`.

### AWS CLI

Contoh 1: Untuk mengaitkan satu atau beberapa instance dengan jendela acara

`associate-instance-event-window` Contoh berikut mengaitkan satu atau lebih instance dengan jendela acara.

```
aws ec2 associate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Output:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
```

```

    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-0598c7d356eba48d7"
      ],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating"
  }
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 2: Untuk mengaitkan tag instance dengan jendela acara

`associate-instance-event-window` Contoh berikut mengaitkan tag instance dengan jendela acara. Masukkan `instance-event-window-id` parameter untuk menentukan jendela acara. Untuk mengaitkan tag instance, tentukan `association-target` parameter, dan untuk nilai parameter, tentukan satu atau beberapa tag.

```

aws ec2 associate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"

```

Output:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [
        {
          "Key": "k2",
          "Value": "v2"
        }
      ],
    }
  }
}

```

```

        {
            "Key": "k1",
            "Value": "v1"
        }
    ],
    "DedicatedHostIds": []
},
"State": "creating"
}
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 3: Untuk mengaitkan Host Khusus dengan jendela acara

Contoh berikut mengaitkan Host Khusus dengan jendela acara. Masukkan `instance-event-window-id` parameter untuk menentukan jendela acara. Untuk mengaitkan Host Khusus, tentukan `--association-target` parameter, dan untuk nilai parameter, tentukan salah satu ID Host Khusus lainnya.

```

aws ec2 associate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "DedicatedHostIds=h-029fa35a02b99801d"

```

Output:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": [
        "h-029fa35a02b99801d"
      ]
    }
  },
  "State": "creating"
}

```

```
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [AssociateInstanceEventWindow](#) di Referensi AWS CLI Perintah.

## associate-ipam-resource-discovery

Contoh kode berikut menunjukkan cara menggunakan `associate-ipam-resource-discovery`.

### AWS CLI

Untuk mengaitkan penemuan sumber daya dengan IPAM

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM dan penemuan sumber daya telah dibuat dan dibagikan dengan Anda oleh AWS akun lain sehingga Anda dapat menggunakan IPAM untuk mengelola dan memantau CIDR sumber daya yang dimiliki oleh akun lain.

### Catatan

Untuk menyelesaikan permintaan ini, Anda memerlukan ID penemuan sumber daya yang bisa Anda dapatkan [describe-ipam-resource-discoveries](#) dan ID IPAM yang bisa Anda dapatkan dengan [deskripsi-ipams](#). Penemuan sumber daya yang Anda kaitkan harus dibagikan terlebih dahulu dengan akun Anda menggunakan AWS RAM. Anda masukkan harus cocok dengan Wilayah rumah IPAM yang `--region` Anda kaitkan dengannya.

`associate-ipam-resource-discovery` Contoh berikut mengaitkan penemuan sumber daya dengan IPAM.

```
aws ec2 associate-ipam-resource-discovery \
  --ipam-id ipam-005f921c17ebd5107 \
  --ipam-resource-discovery-id ipam-res-disco-03e0406de76a044ee \
  --tag-specifications 'ResourceType=ipam-resource-discovery,Tags=[{Key=cost-
center,Value=cc123}]' \
  --region us-east-1
```

### Output:

```
{
  {
```

```

    "IpamResourceDiscoveryAssociation": {
      "OwnerId": "320805250157",
      "IpamResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-04382a6346357cf82",
      "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::320805250157:ipam-
resource-discovery-association/ipam-res-disco-assoc-04382a6346357cf82",
      "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
      "IpamId": "ipam-005f921c17ebd5107",
      "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
      "IpamRegion": "us-east-1",
      "IsDefault": false,
      "ResourceDiscoveryStatus": "active",
      "State": "associate-in-progress",
      "Tags": []
    }
  }
}

```

Setelah Anda mengaitkan penemuan sumber daya, Anda dapat memantau dan/atau mengelola alamat IP sumber daya yang dibuat oleh akun lain. Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [AssociateIpamResourceDiscovery](#) di Referensi AWS CLI Perintah.

## associate-nat-gateway-address

Contoh kode berikut menunjukkan cara menggunakan `associate-nat-gateway-address`.

### AWS CLI

Untuk mengaitkan alamat IP Elastis dengan gateway NAT publik

`associate-nat-gateway-address` Contoh berikut mengaitkan alamat IP Elastis yang ditentukan dengan gateway NAT publik yang ditentukan. AWS secara otomatis menetapkan alamat IPv4 pribadi sekunder.

```

aws ec2 associate-nat-gateway-address \
  --nat-gateway-id nat-1234567890abcdef0 \
  --allocation-ids eipalloc-0be6ecac95EXAMPLE

```

Output:



```
{
  "NatGatewayId": "nat-1234567890abcdef0",
  "NatGatewayAddresses": [
    {
      "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
      "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
      "IsPrimary": false,
      "Status": "associating"
    }
  ]
}
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [AssociateNatGatewayAddress](#) di Referensi AWS CLI Perintah.

## **associate-route-table**

Contoh kode berikut menunjukkan cara menggunakan `associate-route-table`.

### AWS CLI

Untuk mengaitkan tabel rute dengan subnet

Contoh ini mengaitkan tabel rute yang ditentukan dengan subnet yang ditentukan.

Perintah:

```
aws ec2 associate-route-table --route-table-id rtb-22574640 --subnet-id
subnet-9d4a7b6c
```

Output:

```
{
  "AssociationId": "rtbassoc-781d0d1a"
}
```

- Untuk detail API, lihat [AssociateRouteTable](#) di Referensi AWS CLI Perintah.

## **associate-subnet-cidr-block**

Contoh kode berikut menunjukkan cara menggunakan `associate-subnet-cidr-block`.

## AWS CLI

Untuk mengaitkan blok IPv6 CIDR dengan subnet

Contoh ini mengaitkan blok IPv6 CIDR dengan subnet yang ditentukan.

Perintah:

```
aws ec2 associate-subnet-cidr-block --subnet-id subnet-5f46ec3b --ipv6-cidr-block 2001:db8:1234:1a00::/64
```

Output:

```
{
  "SubnetId": "subnet-5f46ec3b",
  "Ipv6CidrBlockAssociation": {
    "Ipv6CidrBlock": "2001:db8:1234:1a00::/64",
    "AssociationId": "subnet-cidr-assoc-3aa54053",
    "Ipv6CidrBlockState": {
      "State": "associating"
    }
  }
}
```

- Untuk detail API, lihat [AssociateSubnetCidrBlock](#) di Referensi AWS CLI Perintah.

## associate-transit-gateway-multicast-domain

Contoh kode berikut menunjukkan cara menggunakan `associate-transit-gateway-multicast-domain`.

### AWS CLI

Untuk mengaitkan gateway transit dengan domain multicast

`associate-transit-gateway-multicast-domain` Contoh berikut mengaitkan subnet dan lampiran tertentu dengan domain multicast yang ditentukan.

```
aws ec2 associate-transit-gateway-multicast-domain \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --transit-gateway-attachment-id tgw-attach-028c1dd0f8f5cbe8e \
  --subnet-ids subnet-000de86e3b49c932a \
```

```
--transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{
  "Associations": [
    {
      "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
      "TransitGatewayAttachmentId": "tgw-attach-028c1dd0f8f5cbe8e",
      "ResourceId": "vpc-01128d2c240c09bd5",
      "ResourceType": "vpc",
      "Subnets": [
        {
          "SubnetId": "subnet-000de86e3b49c932a",
          "State": "associating"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Transit Gateways.

- Untuk detail API, lihat [AssociateTransitGatewayMulticastDomain](#) di Referensi AWS CLI Perintah.

## associate-transit-gateway-route-table

Contoh kode berikut menunjukkan cara menggunakan `associate-transit-gateway-route-table`.

AWS CLI

Untuk mengaitkan tabel rute gateway transit dengan lampiran gateway transit

Contoh berikut mengaitkan tabel rute gateway transit yang ditentukan dengan lampiran VPC yang ditentukan.

```
aws ec2 associate-transit-gateway-route-table \
  --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE \
  --transit-gateway-attachment-id tgw-attach-0b5968d3b6EXAMPLE
```

Output:

```
{
  "Association": {
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
    "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
    "ResourceId": "vpc-0065acced4EXAMPLE",
    "ResourceType": "vpc",
    "State": "associating"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengaitkan Tabel Rute Transit Gateway](#) di Panduan Gerbang AWS Transit.

- Untuk detail API, lihat [AssociateTransitGatewayRouteTable](#) di Referensi AWS CLI Perintah.

## associate-vpc-cidr-block

Contoh kode berikut menunjukkan cara menggunakan `associate-vpc-cidr-block`.

### AWS CLI

Contoh 1: Untuk mengaitkan blok CIDR IPv6 yang disediakan Amazon dengan VPC

`associate-vpc-cidr-block` Contoh berikut mengaitkan blok IPv6 CIDR dengan VPC yang ditentukan. :

```
aws ec2 associate-vpc-cidr-block \
  --amazon-provided-ipv6-cidr-block \
  --ipv6-cidr-block-network-border-group us-west-2-lax-1 \
  --vpc-id vpc-8EXAMPLE
```

Output:

```
{
  "Ipv6CidrBlockAssociation": {
    "AssociationId": "vpc-cidr-assoc-0838ce7d9dEXAMPLE",
    "Ipv6CidrBlockState": {
      "State": "associating"
    },
    "NetworkBorderGroup": "us-west-2-lax-1"
  },
  "VpcId": "vpc-8EXAMPLE"
```

```
}
```

Contoh 2: Untuk mengaitkan blok CIDR IPv4 tambahan dengan VPC

`associate-vpc-cidr-block` Contoh berikut mengaitkan blok IPv4 CIDR dengan VPC `10.2.0.0/16` yang ditentukan.

```
aws ec2 associate-vpc-cidr-block \  
  --vpc-id vpc-1EXAMPLE \  
  --cidr-block 10.2.0.0/16
```

Output:

```
{  
  "CidrBlockAssociation": {  
    "AssociationId": "vpc-cidr-assoc-2EXAMPLE",  
    "CidrBlock": "10.2.0.0/16",  
    "CidrBlockState": {  
      "State": "associating"  
    }  
  },  
  "VpcId": "vpc-1EXAMPLE"  
}
```

- Untuk detail API, lihat [AssociateVpcCidrBlock](#) di Referensi AWS CLI Perintah.

## **attach-classic-link-vpc**

Contoh kode berikut menunjukkan cara menggunakan `attach-classic-link-vpc`.

### AWS CLI

Untuk menautkan (melampirkan) instance EC2-Classik ke VPC

Contoh ini menautkan instance `i-1234567890abcdef0` ke VPC `vpc-88888888` melalui grup keamanan VPC `sg-12312312`.

Perintah:

```
aws ec2 attach-classic-link-vpc --instance-id i-1234567890abcdef0 --vpc-id  
vpc-88888888 --groups sg-12312312
```

**Output:**

```
{
  "Return": true
}
```

- Untuk detail API, lihat [AttachClassicLinkVpc](#) di Referensi AWS CLI Perintah.

**attach-internet-gateway**

Contoh kode berikut menunjukkan cara menggunakan `attach-internet-gateway`.

**AWS CLI**

Untuk melampirkan gateway internet ke VPC Anda

`attach-internet-gateway` Contoh berikut melampirkan gateway internet yang ditentukan ke VPC tertentu.

```
aws ec2 attach-internet-gateway \
  --internet-gateway-id igw-0d0fb496b3EXAMPLE \
  --vpc-id vpc-0a60eb65b4EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Gateway internet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [AttachInternetGateway](#) di Referensi AWS CLI Perintah.

**attach-network-interface**

Contoh kode berikut menunjukkan cara menggunakan `attach-network-interface`.

**AWS CLI**

Contoh 1: Untuk melampirkan antarmuka jaringan ke sebuah instance

`attach-network-interface` Contoh berikut melampirkan antarmuka jaringan yang ditentukan untuk contoh yang ditentukan.

```
aws ec2 attach-network-interface \
```

```
--network-interface-id eni-0dc56a8d4640ad10a \  
--instance-id i-1234567890abcdef0 \  
--device-index 1
```

Output:

```
{  
  "AttachmentId": "eni-attach-01a8fc87363f07cf9"  
}
```

Untuk informasi selengkapnya, lihat [Antarmuka jaringan elastis](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk melampirkan antarmuka jaringan ke instance dengan beberapa kartu jaringan `attach-network-interface` Contoh berikut melampirkan antarmuka jaringan yang ditentukan untuk contoh yang ditentukan dan kartu jaringan.

```
aws ec2 attach-network-interface \  
  --network-interface-id eni-07483b1897541ad83 \  
  --instance-id i-01234567890abcdef \  
  --network-card-index 1 \  
  --device-index 1
```

Output:

```
{  
  "AttachmentId": "eni-attach-0fbd7ee87a88cd06c"  
}
```

Untuk informasi selengkapnya, lihat [Antarmuka jaringan elastis](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [AttachNetworkInterface](#) di Referensi AWS CLI Perintah.

## **attach-verified-access-trust-provider**

Contoh kode berikut menunjukkan cara menggunakan `attach-verified-access-trust-provider`.

## AWS CLI

Untuk melampirkan penyedia kepercayaan ke sebuah instans

`attach-verified-access-trust-provider` Contoh berikut melampirkan penyedia kepercayaan Akses Terverifikasi yang ditentukan ke instance Akses Terverifikasi yang ditentukan.

```
aws ec2 attach-verified-access-trust-provider \  
  --verified-access-instance-id vai-0ce000c0b7643abea \  
  --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Output:

```
{  
  "VerifiedAccessTrustProvider": {  
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",  
    "Description": "",  
    "TrustProviderType": "user",  
    "UserTrustProviderType": "iam-identity-center",  
    "PolicyReferenceName": "idc",  
    "CreationTime": "2023-08-25T19:00:38",  
    "LastUpdatedTime": "2023-08-25T19:00:38"  
  },  
  "VerifiedAccessInstance": {  
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",  
    "Description": "",  
    "VerifiedAccessTrustProviders": [  
      {  
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",  
        "TrustProviderType": "user",  
        "UserTrustProviderType": "iam-identity-center"  
      }  
    ],  
    "CreationTime": "2023-08-25T18:27:56",  
    "LastUpdatedTime": "2023-08-25T18:27:56"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Instans Akses Terverifikasi](#) di Panduan Pengguna Akses AWS Terverifikasi.

- Untuk detail API, lihat [AttachVerifiedAccessTrustProvider](#) di Referensi AWS CLI Perintah.



## attach-volume

Contoh kode berikut menunjukkan cara menggunakan `attach-volume`.

### AWS CLI

Untuk melampirkan volume ke sebuah instance

Perintah contoh ini melampirkan volume (`vol-1234567890abcdef0`) ke instance (`i-01474ef662b89480`) sebagai `/dev/sdf`.

Perintah:

```
aws ec2 attach-volume --volume-id vol-1234567890abcdef0 --instance-id
i-01474ef662b89480 --device /dev/sdf
```

Output:

```
{
  "AttachTime": "YYYY-MM-DDTHH:MM:SS.000Z",
  "InstanceId": "i-01474ef662b89480",
  "VolumeId": "vol-1234567890abcdef0",
  "State": "attaching",
  "Device": "/dev/sdf"
}
```

- Untuk detail API, lihat [AttachVolume](#) di Referensi AWS CLI Perintah.

## attach-vpn-gateway

Contoh kode berikut menunjukkan cara menggunakan `attach-vpn-gateway`.

### AWS CLI

Untuk melampirkan gateway pribadi virtual ke VPC Anda

`attach-vpn-gateway` Contoh berikut melampirkan gateway pribadi virtual yang ditentukan ke VPC yang ditentukan.

```
aws ec2 attach-vpn-gateway \
```

```
--vpn-gateway-id vgw-9a4cacf3 \  
--vpc-id vpc-a01106c2
```

Output:

```
{  
  "VpcAttachment": {  
    "State": "attaching",  
    "VpcId": "vpc-a01106c2"  
  }  
}
```

- Untuk detail API, lihat [AttachVpnGateway](#) di Referensi AWS CLI Perintah.

## authorize-client-vpn-ingress

Contoh kode berikut menunjukkan cara menggunakan `authorize-client-vpn-ingress`.

### AWS CLI

Untuk menambahkan aturan otorisasi untuk titik akhir Client VPN

`authorize-client-vpn-ingress` Contoh berikut menambahkan aturan otorisasi ingress yang memungkinkan semua klien untuk mengakses internet (`0.0.0.0/0`)

```
aws ec2 authorize-client-vpn-ingress \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --target-network-cidr 0.0.0.0/0 \  
  --authorize-all-groups
```

Output:

```
{  
  "Status": {  
    "Code": "authorizing"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Aturan Otorisasi](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [AuthorizeClientVpnIngress](#) di Referensi AWS CLI Perintah.

## authorize-security-group-egress

Contoh kode berikut menunjukkan cara menggunakan `authorize-security-group-egress`.

### AWS CLI

Untuk menambahkan aturan yang memungkinkan lalu lintas keluar ke rentang alamat tertentu

Perintah contoh ini menambahkan aturan yang memberikan akses ke rentang alamat yang ditentukan pada port TCP 80.

Perintah (Linux):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges='[{"CidrIp=10.0.0.0/16}]'
```

Perintah (Windows):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{"CidrIp=10.0.0.0/16}]
```

Untuk menambahkan aturan yang memungkinkan lalu lintas keluar ke grup keamanan tertentu

Perintah contoh ini menambahkan aturan yang memberikan akses ke grup keamanan yang ditentukan pada port TCP 80.

Perintah (Linux):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,UserIdGroupPairs='[{"GroupId=sg-4b51a32f}]'
```

Perintah (Windows):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,UserIdGroupPairs=[{"GroupId=sg-4b51a32f}]
```

- Untuk detail API, lihat [AuthorizeSecurityGroupEgress](#) di Referensi AWS CLI Perintah.

## authorize-security-group-ingress

Contoh kode berikut menunjukkan cara menggunakan `authorize-security-group-ingress`.

## AWS CLI

Contoh 1: Untuk menambahkan aturan yang mengizinkan lalu lintas SSH masuk

Contoh `authorize-security-group-ingress` berikut menambahkan aturan yang mengizinkan lalu lintas masuk pada TCP port 22 (SSH).

```
aws ec2 authorize-security-group-ingress \  
  --group-id sg-1234567890abcdef0 \  
  --protocol tcp \  
  --port 22 \  
  --cidr 203.0.113.0/24
```

Output:

```
{  
  "Return": true,  
  "SecurityGroupRules": [  
    {  
      "SecurityGroupRuleId": "sgr-01afa97ef3e1bedfc",  
      "GroupId": "sg-1234567890abcdef0",  
      "GroupOwnerId": "123456789012",  
      "IsEgress": false,  
      "IpProtocol": "tcp",  
      "FromPort": 22,  
      "ToPort": 22,  
      "CidrIpv4": "203.0.113.0/24"  
    }  
  ]  
}
```

Contoh 2: Untuk menambahkan aturan yang mengizinkan lalu lintas HTTP masuk dari grup keamanan lain

Contoh `authorize-security-group-ingress` berikut menambahkan aturan yang mengizinkan akses masuk pada TCP port 80 dari grup keamanan sumber `sg-1a2b3c4d`. Grup sumber harus berada di VPC atau VPC peer yang sama (memerlukan koneksi peering VPC). Lalu lintas masuk diizinkan berdasarkan alamat IP privat dari instans yang dikaitkan dengan grup keamanan sumber (bukan alamat IP publik atau alamat IP Elastis).

```
aws ec2 authorize-security-group-ingress \  
  --group-id sg-1a2b3c4d \  
  --group-id sg-1a2b3c4d \  
  --protocol tcp \  
  --port 80 \  
  --source-group sg-1a2b3c4d
```

```
--group-id sg-1234567890abcdef0 \  
--protocol tcp \  
--port 80 \  
--source-group sg-1a2b3c4d
```

**Output:**

```
{  
  "Return": true,  
  "SecurityGroupRules": [  
    {  
      "SecurityGroupRuleId": "sgr-01f4be99110f638a7",  
      "GroupId": "sg-1234567890abcdef0",  
      "GroupOwnerId": "123456789012",  
      "IsEgress": false,  
      "IpProtocol": "tcp",  
      "FromPort": 80,  
      "ToPort": 80,  
      "ReferencedGroupInfo": {  
        "GroupId": "sg-1a2b3c4d",  
        "UserId": "123456789012"  
      }  
    }  
  ]  
}
```

**Contoh 3: Untuk menambahkan banyak aturan dalam panggilan yang sama**

Contoh `authorize-security-group-ingress` berikut menggunakan parameter `ip-permissions` untuk menambahkan dua aturan masuk, satu yang mengizinkan akses masuk pada TCP port 3389 (RDP) dan satu yang mengizinkan ping/ICMP.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions  
IpProtocol =tcp, =3389, FromPort =3389, = "[{=172.31.0.0/16}]" =icmp, =-1, =-1, =  
"[{=172.31.0.0/16}ToPort]" IpRanges CidrIp IpProtocol FromPort ToPort IpRanges CidrIp
```

**Output:**

```
{  
  "Return": true,  
  "SecurityGroupRules": [  
    {  
      "SecurityGroupRuleId": "sgr-01f4be99110f638a7",  
      "GroupId": "sg-1234567890abcdef0",  
      "GroupOwnerId": "123456789012",  
      "IsEgress": false,  
      "IpProtocol": "tcp",  
      "FromPort": 3389,  
      "ToPort": 3389,  
      "ReferencedGroupInfo": {  
        "GroupId": "sg-1a2b3c4d",  
        "UserId": "123456789012"  
      }  
    },  
    {  
      "SecurityGroupRuleId": "sgr-01f4be99110f638a8",  
      "GroupId": "sg-1234567890abcdef0",  
      "GroupOwnerId": "123456789012",  
      "IsEgress": false,  
      "IpProtocol": "icmp",  
      "FromPort": -1,  
      "ToPort": -1,  
      "ReferencedGroupInfo": {  
        "GroupId": "sg-1a2b3c4d",  
        "UserId": "123456789012"  
      }  
    }  
  ]  
}
```

```

    {
      "SecurityGroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": 3389,
      "ToPort": 3389,
      "CidrIpv4": "172.31.0.0/16"
    },
    {
      "SecurityGroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv4": "172.31.0.0/16"
    }
  ]
}

```

#### Contoh 4: Guna menambahkan aturan untuk lalu lintas ICMP

Contoh `authorize-security-group-ingress` berikut menggunakan parameter `ip-permissions` untuk menambahkan aturan masuk yang mengizinkan pesan ICMP Destination Unreachable: Fragmentation Needed and Don't Fragment was Set (Tipe 3, Kode 4) dari mana saja.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions
=icmp, =3, =4, = [{"=0.0.0.0/0}]" IpProtocol FromPort ToPort IpRanges CidrIp
```

Output:

```

{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,

```

```

        "IpProtocol": "icmp",
        "FromPort": 3,
        "ToPort": 4,
        "CidrIpv4": "0.0.0.0/0"
    }
]
}

```

### Contoh 5: Guna menambahkan aturan untuk lalu lintas IPv6

Contoh `authorize-security-group-ingress` berikut menggunakan parameter `ip-permissions` untuk menambahkan aturan masuk yang mengizinkan akses SSH (port 22) dari rentang IPv6 `2001:db8:1234:1a00::/64`.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions =tcp, =22, =22, Ipv6Ranges= "[{6=2001:db 8:1234:1 a00: IpProtocol :/64}]" FromPort ToPort CidrIpv
```

### Output:

```

{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-0455bc68b60805563",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": 22,
      "ToPort": 22,
      "CidrIpv6": "2001:db8:1234:1a00::/64"
    }
  ]
}

```

### Contoh 6: Guna menambahkan aturan untuk lalu lintas ICMPv6

Contoh `authorize-security-group-ingress` berikut menggunakan parameter `ip-permissions` untuk menambahkan aturan masuk yang mengizinkan lalu lintas ICMPv6 dari mana saja.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions
=icmpv6, Ipv6Ranges= "[{6=: /0}]" IpProtocol CidrIpv
```

Output:

```
{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sgr-04b612d9363ab6327",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "icmpv6",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv6": "::-/0"
    }
  ]
}
```

Contoh 7: Tambahkan aturan dengan deskripsi

Contoh `authorize-security-group-ingress` berikut menggunakan parameter `ip-permissions` untuk menambahkan aturan masuk yang mengizinkan lalu lintas RDP dari rentang alamat IPv4 tertentu. Aturan mencakup deskripsi untuk membantu Anda mengidentifikasinya nanti.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions
IpProtocol =tcp, =3389, =3389, = "[{=203.0.113.0/24, description='akses RDP dari kantor NY'}]"
FromPort ToPort IpRanges CidrIpv
```

Output:

```
{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sgr-0397bbcc01e974db3",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
```



```

        "IpProtocol": "tcp",
        "FromPort": 3389,
        "ToPort": 3389,
        "CidrIpv4": "203.0.113.0/24",
        "Description": "RDP access from NY office"
    }
]
}

```

Contoh 8: Untuk menambahkan aturan masuk yang menggunakan daftar prefiks

Contoh `authorize-security-group-ingress` berikut menggunakan parameter `ip-permissions` untuk menambahkan aturan masuk yang mengizinkan semua lalu lintas untuk rentang CIDR dalam daftar prefiks tertentu.

```
aws ec2 authorize-security-group-ingress --group-id sg-04a351bfe432d4e71 --ip-permissions =all,
=[{=pl-002dc3ec097de1514}] IpProtocol PrefixListIds PrefixListId
```

Output:

```

{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-09c74b32f677c6c7c",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "PrefixListId": "pl-0721453c7ac4ec009"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Grup keamanan](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [AuthorizeSecurityGroupIngress](#) di Referensi AWS CLI Perintah.

## bundle-instance

Contoh kode berikut menunjukkan cara menggunakan `bundle-instance`.

## AWS CLI

Untuk menggabungkan sebuah instance

Contoh ini menggabungkan instance `i-1234567890abcdef0` ke bucket yang disebut `bundletasks`. Sebelum Anda menentukan nilai untuk ID kunci akses Anda, tinjau dan ikuti panduan dalam Praktik Terbaik untuk Mengelola Kunci AWS Akses.

Perintah:

```
aws ec2 bundle-instance --instance-id i-1234567890abcdef0 --bucket bundletasks --
prefix winami --owner-akid AK12AJEXAMPLE --owner-sak example123example
```

Output:

```
{
  "BundleTask": {
    "UpdateTime": "2015-09-15T13:30:35.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "Storage": {
      "S3": {
        "Prefix": "winami",
        "Bucket": "bundletasks"
      }
    },
    "State": "pending",
    "StartTime": "2015-09-15T13:30:35.000Z",
    "BundleId": "bun-294e041f"
  }
}
```

- Untuk detail API, lihat [BundleInstance](#) di Referensi AWS CLI Perintah.

## cancel-bundle-task

Contoh kode berikut menunjukkan cara menggunakan `cancel-bundle-task`.

## AWS CLI

Untuk membatalkan tugas bundel

Contoh ini membatalkan tugas `bun-2a4e041c` bundel.

**Perintah:**

```
aws ec2 cancel-bundle-task --bundle-id bun-2a4e041c
```

**Output:**

```
{
  "BundleTask": {
    "UpdateTime": "2015-09-15T13:27:40.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "Storage": {
      "S3": {
        "Prefix": "winami",
        "Bucket": "bundletasks"
      }
    },
    "State": "cancelling",
    "StartTime": "2015-09-15T13:24:35.000Z",
    "BundleId": "bun-2a4e041c"
  }
}
```

- Untuk detail API, lihat [CancelBundleTask](#) di Referensi AWS CLI Perintah.

**cancel-capacity-reservation-fleets**

Contoh kode berikut menunjukkan cara menggunakan `cancel-capacity-reservation-fleets`.

**AWS CLI**

Untuk membatalkan Armada Reservasi Kapasitas

`cancel-capacity-reservation-fleets` Contoh berikut membatalkan Armada Reservasi Kapasitas yang ditentukan dan kapasitas yang dicadangkannya. Saat Anda membatalkan Armada, statusnya berubah menjadi `cancelled`, dan Armada tidak dapat lagi membuat Reservasi Kapasitas baru. Selain itu, semua Reservasi Kapasitas individu di Armada dibatalkan, dan instance yang sebelumnya berjalan dalam kapasitas cadangan terus berjalan normal dalam kapasitas bersama.

```
aws ec2 cancel-capacity-reservation-fleets \
  --capacity-reservation-fleet-ids crf-abcdef01234567890
```

**Output:**

```
{
  "SuccessfulFleetCancellations": [
    {
      "CurrentFleetState": "cancelling",
      "PreviousFleetState": "active",
      "CapacityReservationFleetId": "crf-abcdef01234567890"
    }
  ],
  "FailedFleetCancellations": []
}
```

Untuk informasi selengkapnya tentang Armada Reservasi [Kapasitas](#), lihat [Armada Reservasi Kapasitas](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CancelCapacityReservationFleets](#) di Referensi AWS CLI Perintah.

**cancel-capacity-reservation**

Contoh kode berikut menunjukkan cara menggunakan `cancel-capacity-reservation`.

**AWS CLI**

Untuk membatalkan reservasi kapasitas

`cancel-capacity-reservation` Contoh berikut membatalkan reservasi kapasitas yang ditentukan.

```
aws ec2 cancel-capacity-reservation \
  --capacity-reservation-id cr-1234abcd56EXAMPLE
```

**Output:**

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Membatalkan Reservasi Kapasitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [CancelCapacityReservation](#) di Referensi AWS CLI Perintah.

## cancel-conversion-task

Contoh kode berikut menunjukkan cara menggunakan `cancel-conversion-task`.

### AWS CLI

Untuk membatalkan konversi aktif dari sebuah instans atau volume

Contoh ini membatalkan unggahan yang terkait dengan ID tugas `import-i-fh 95npoc`. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 cancel-conversion-task --conversion-task-id import-i-fh95npoc
```

- Untuk detail API, lihat [CancelConversionTask](#) di Referensi AWS CLI Perintah.

## cancel-export-task

Contoh kode berikut menunjukkan cara menggunakan `cancel-export-task`.

### AWS CLI

Untuk membatalkan tugas ekspor aktif

Contoh ini membatalkan tugas ekspor aktif dengan ID tugas `export-i-fgelt0i7`. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 cancel-export-task --export-task-id export-i-fgelt0i7
```

- Untuk detail API, lihat [CancelExportTask](#) di Referensi AWS CLI Perintah.

## cancel-image-launch-permission

Contoh kode berikut menunjukkan cara menggunakan `cancel-image-launch-permission`.

## AWS CLI

Untuk membatalkan AMI yang dibagikan dengan akun Amazon Web Services

`cancel-image-launch-permission` Contoh berikut menghapus akun Anda dari izin peluncuran AMI yang ditentukan.

```
aws ec2 cancel-image-launch-permission \  
  --image-id ami-0123456789example \  
  --region us-east-1
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya, lihat [Batalkan agar AMI dibagikan dengan akun Amazon Web Services](#) Anda di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CancelImageLaunchPermission](#) di Referensi AWS CLI Perintah.

## `cancel-import-task`

Contoh kode berikut menunjukkan cara menggunakan `cancel-import-task`.

## AWS CLI

Untuk membatalkan tugas impor

`cancel-import-task` Contoh berikut membatalkan tugas gambar impor yang ditentukan.

```
aws ec2 cancel-import-task \  
  --import-task-id import-ami-1234567890abcdef0
```

Output:

```
{  
  "ImportTaskId": "import-ami-1234567890abcdef0",  
  "PreviousState": "active",  
  "State": "deleting"
```

```
}

```

- Untuk detail API, lihat [CancelImportTask](#) di Referensi AWS CLI Perintah.

## cancel-reserved-instances-listing

Contoh kode berikut menunjukkan cara menggunakan `cancel-reserved-instances-listing`.

### AWS CLI

Untuk membatalkan daftar Instans Cadangan

`cancel-reserved-instances-listing` Contoh berikut membatalkan daftar Instans Cadangan yang ditentukan.

```
aws ec2 cancel-reserved-instances-listing \
  --reserved-instances-listing-id 5ec28771-05ff-4b9b-aa31-9e57dexample

```

- Untuk detail API, lihat [CancelReservedInstancesListing](#) di Referensi AWS CLI Perintah.

## cancel-spot-fleet-requests

Contoh kode berikut menunjukkan cara menggunakan `cancel-spot-fleet-requests`.

### AWS CLI

Contoh 1: Untuk membatalkan permintaan armada Spot dan menghentikan instans terkait

`cancel-spot-fleet-requests` Contoh berikut membatalkan permintaan Armada Spot dan mengakhiri Instans Sesuai Permintaan dan Instans Spot terkait.

```
aws ec2 cancel-spot-fleet-requests \
  --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
  --terminate-instances

```

Output:

```
{
  "SuccessfulFleetRequests": [
    {

```

```

        "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
        "CurrentSpotFleetRequestState": "cancelled_terminating",
        "PreviousSpotFleetRequestState": "active"
    }
],
"UnsuccessfulFleetRequests": []
}

```

Untuk informasi selengkapnya, lihat [Membatalkan permintaan Armada Spot](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 2: Untuk membatalkan permintaan armada Spot tanpa menghentikan instance terkait `cancel-spot-fleet-requests` Contoh berikut membatalkan permintaan Armada Spot tanpa menghentikan Instans Sesuai Permintaan dan Instans Spot terkait.

```

aws ec2 cancel-spot-fleet-requests \
  --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
  --no-terminate-instances

```

Output:

```

{
  "SuccessfulFleetRequests": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
      "CurrentSpotFleetRequestState": "cancelled_running",
      "PreviousSpotFleetRequestState": "active"
    }
  ],
  "UnsuccessfulFleetRequests": []
}

```

Untuk informasi selengkapnya, lihat [Membatalkan permintaan Armada Spot](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [CancelSpotFleetRequests](#) di Referensi AWS CLI Perintah.

## **cancel-spot-instance-requests**

Contoh kode berikut menunjukkan cara menggunakan `cancel-spot-instance-requests`.



## AWS CLI

Untuk membatalkan permintaan Instans Spot

Perintah contoh ini membatalkan permintaan Instans Spot.

Perintah:

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

Output:

```
{
  "CancelledSpotInstanceRequests": [
    {
      "State": "cancelled",
      "SpotInstanceRequestId": "sir-08b93456"
    }
  ]
}
```

- Untuk detail API, lihat [CancelSpotInstanceRequests](#) di Referensi AWS CLI Perintah.

## confirm-product-instance

Contoh kode berikut menunjukkan cara menggunakan `confirm-product-instance`.

### AWS CLI

Untuk mengonfirmasi contoh produk

Contoh ini menentukan apakah kode produk yang ditentukan dikaitkan dengan contoh yang ditentukan.

Perintah:

```
aws ec2 confirm-product-instance --product-code 774F4FF8 --instance-id
i-1234567890abcdef0
```

Output:

```
{
```

```
"OwnerId": "123456789012"
}
```

- Untuk detail API, lihat [ConfirmProductInstance](#) di Referensi AWS CLI Perintah.

## copy-fpga-image

Contoh kode berikut menunjukkan cara menggunakan `copy-fpga-image`.

### AWS CLI

Untuk menyalin gambar Amazon FPGA

Contoh ini menyalin AFI yang ditentukan dari `us-east-1` wilayah ke wilayah saat ini (`eu-west-1`).

Perintah:

```
aws ec2 copy-fpga-image --name copy-afi --source-fpga-image-id afi-0d123e123bfc85abc
--source-region us-east-1 --region eu-west-1
```

Output:

```
{
  "FpgaImageId": "afi-06b12350a123fbabc"
}
```

- Untuk detail API, lihat [CopyFpgaImage](#) di Referensi AWS CLI Perintah.

## copy-image

Contoh kode berikut menunjukkan cara menggunakan `copy-image`.

### AWS CLI

Contoh 1: Untuk menyalin AMI ke Wilayah lain

`copy-image` Contoh perintah berikut menyalin AMI yang ditentukan dari `us-west-2` Region ke `us-east-1` Region dan menambahkan deskripsi singkat.

```
aws ec2 copy-image \
```

```
--region us-east-1 \  
--name ami-name \  
--source-region us-west-2 \  
--source-image-id ami-066877671789bd71b \  
--description "This is my copied image."
```

Output:

```
{  
  "ImageId": "ami-0123456789abcdefg"  
}
```

Untuk informasi selengkapnya, lihat [Menyalin AMI](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menyalin AMI ke Wilayah lain dan mengenkripsi snapshot dukungan

`copy-image` Perintah berikut menyalin AMI yang ditentukan dari `us-west-2` Wilayah ke Wilayah saat ini dan mengenkripsi snapshot dukungan menggunakan kunci KMS yang ditentukan.

```
aws ec2 copy-image \  
  --source-region us-west-2 \  
  --name ami-name \  
  --source-image-id ami-066877671789bd71b \  
  --encrypted \  
  --kms-key-id alias/my-kms-key
```

Output:

```
{  
  "ImageId": "ami-0123456789abcdefg"  
}
```

Untuk informasi selengkapnya, lihat [Menyalin AMI](#) di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk menyertakan tag AMI yang ditentukan pengguna saat menyalin AMI

`copy-image` Perintah berikut menggunakan `--copy-image-tags` parameter untuk menyalin tag AMI yang ditentukan pengguna saat menyalin AMI.

```
aws ec2 copy-image \  
  --region us-east-1 \  
  --copy-image-tags
```

```
--name ami-name \  
--source-region us-west-2 \  
--source-image-id ami-066877671789bd71b \  
--description "This is my copied image."  
--copy-image-tags
```

Output:

```
{  
  "ImageId": "ami-0123456789abcdefg"  
}
```

Untuk informasi selengkapnya, lihat [Menyalin AMI](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CopyImage](#) di Referensi AWS CLI Perintah.

## copy-snapshot

Contoh kode berikut menunjukkan cara menggunakan `copy-snapshot`.

### AWS CLI

Contoh 1: Untuk menyalin snapshot ke Wilayah lain

`copy-snapshot` Contoh perintah berikut menyalin snapshot yang ditentukan dari `us-west-2` Region ke `us-east-1` Region dan menambahkan deskripsi singkat.

```
aws ec2 copy-snapshot \  
  --region us-east-1 \  
  --source-region us-west-2 \  
  --source-snapshot-id snap-066877671789bd71b \  
  --description "This is my copied snapshot."
```

Output:

```
{  
  "SnapshotId": "snap-066877671789bd71b"  
}
```

Untuk informasi selengkapnya, lihat [Menyalin snapshot Amazon EBS](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menyalin snapshot yang tidak terenkripsi dan mengenkripsi snapshot baru

copy-snapshot Perintah berikut menyalin snapshot tak terenkripsi yang ditentukan dari us-west-2 Wilayah ke Wilayah saat ini dan mengenkripsi snapshot baru menggunakan kunci KMS yang ditentukan.

```
aws ec2 copy-snapshot \  
  --source-region us-west-2 \  
  --source-snapshot-id snap-066877671789bd71b \  
  --encrypted \  
  --kms-key-id alias/my-kms-key
```

Output:

```
{  
  "SnapshotId": "snap-066877671789bd71b"  
}
```

Untuk informasi selengkapnya, lihat [Menyalin snapshot Amazon EBS](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CopySnapshot](#) di Referensi AWS CLI Perintah.

## create-capacity-reservation-fleet

Contoh kode berikut menunjukkan cara menggunakan `create-capacity-reservation-fleet`.

### AWS CLI

Untuk membuat Armada Reservasi Kapasitas

`create-capacity-reservation-fleet` Contoh berikut membuat Armada Reservasi Kapasitas untuk jenis instans yang ditentukan dalam permintaan, hingga total kapasitas target yang ditentukan. Jumlah instans kapasitas yang dipesan Armada Reservasi Kapasitas tergantung pada total kapasitas target dan bobot tipe instans yang Anda tentukan dalam permintaan. Tentukan jenis instance yang akan digunakan dan prioritas untuk masing-masing jenis instans yang ditunjuk.

```
aws ec2 create-capacity-reservation-fleet \  
  --total-target-capacity 24 \  
  --allocation-strategy prioritized \  
  --instance-type t3.xlarge \  
  --availability-zone us-east-1a
```

```
--instance-match-criteria open \  
--tenancy default \  
--end-date 2022-12-31T23:59:59.000Z \  
--instance-type-specifications file://instanceTypeSpecification.json
```

Isi dari `instanceTypeSpecification.json`:

```
[  
  {  
    "InstanceType": "m5.xlarge",  
    "InstancePlatform": "Linux/UNIX",  
    "Weight": 3.0,  
    "AvailabilityZone": "us-east-1a",  
    "EbsOptimized": true,  
    "Priority" : 1  
  }  
]
```

Output:

```
{  
  "Status": "submitted",  
  "TotalFulfilledCapacity": 0.0,  
  "CapacityReservationFleetId": "crf-abcdef01234567890",  
  "TotalTargetCapacity": 24  
}
```

Untuk informasi selengkapnya tentang Armada Reservasi [Kapasitas](#), lihat [Armada Reservasi Kapasitas](#) di Panduan Pengguna Amazon EC2.

Untuk informasi selengkapnya tentang bobot tipe instans dan total kapasitas target, lihat [Bobot tipe instans](#) dan [Total kapasitas target](#) di Panduan Pengguna Amazon EC2.

Untuk informasi selengkapnya tentang menetapkan prioritas untuk jenis instans tertentu, lihat [Strategi alokasi](#) dan [prioritas tipe Instans](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CreateCapacityReservationFleet](#) di Referensi AWS CLI Perintah.

## **create-capacity-reservation**

Contoh kode berikut menunjukkan cara menggunakan `create-capacity-reservation`.

## AWS CLI

### Contoh 1: Untuk membuat Reservasi Kapasitas

`create-capacity-reservation` Contoh berikut membuat reservasi kapasitas di `eu-west-1a` Availability Zone, di mana Anda dapat meluncurkan tiga `t2.medium` instance yang menjalankan sistem operasi Linux/Unix. Secara default, reservasi kapasitas dibuat dengan kriteria pencocokan instans terbuka dan tidak ada dukungan untuk penyimpanan sementara, dan tetap aktif hingga Anda membatalkannya secara manual.

```
aws ec2 create-capacity-reservation \
  --availability-zone eu-west-1a \
  --instance-type t2.medium \
  --instance-platform Linux/UNIX \
  --instance-count 3
```

### Output:

```
{
  "CapacityReservation": {
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
    "EndDateType": "unlimited",
    "AvailabilityZone": "eu-west-1a",
    "InstanceMatchCriteria": "open",
    "EphemeralStorage": false,
    "CreateDate": "2019-08-16T09:27:35.000Z",
    "AvailableInstanceCount": 3,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 3,
    "State": "active",
    "Tenancy": "default",
    "EbsOptimized": false,
    "InstanceType": "t2.medium"
  }
}
```

Contoh 2: Untuk membuat Reservasi Kapasitas yang secara otomatis berakhir pada tanggal/waktu tertentu

`create-capacity-reservation` Contoh berikut membuat reservasi kapasitas di `eu-west-1a` Availability Zone, di mana Anda dapat meluncurkan tiga `m5.large` instance yang menjalankan

sistem operasi Linux/Unix. Reservasi kapasitas ini secara otomatis berakhir pada 31/08/2019 pukul 23:59:59.

```
aws ec2 create-capacity-reservation \  
  --availability-zone eu-west-1a \  
  --instance-type m5.large \  
  --instance-platform Linux/UNIX \  
  --instance-count 3 \  
  --end-date-type limited \  
  --end-date 2019-08-31T23:59:59Z
```

Output:

```
{  
  "CapacityReservation": {  
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",  
    "EndDateType": "limited",  
    "AvailabilityZone": "eu-west-1a",  
    "EndDate": "2019-08-31T23:59:59.000Z",  
    "InstanceMatchCriteria": "open",  
    "EphemeralStorage": false,  
    "CreateDate": "2019-08-16T10:15:53.000Z",  
    "AvailableInstanceCount": 3,  
    "InstancePlatform": "Linux/UNIX",  
    "TotalInstanceCount": 3,  
    "State": "active",  
    "Tenancy": "default",  
    "EbsOptimized": false,  
    "InstanceType": "m5.large"  
  }  
}
```

Contoh 3: Untuk membuat Reservasi Kapasitas yang hanya menerima peluncuran instans yang ditargetkan

`create-capacity-reservation` Contoh berikut membuat reservasi kapasitas yang hanya menerima peluncuran instans yang ditargetkan.

```
aws ec2 create-capacity-reservation \  
  --availability-zone eu-west-1a \  
  --instance-type m5.large \  
  --instance-platform Linux/UNIX \  
  --instance-count 3
```



```
--instance-count 3 \  
--instance-match-criteria targeted
```

Output:

```
{  
  "CapacityReservation": {  
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",  
    "EndDateType": "unlimited",  
    "AvailabilityZone": "eu-west-1a",  
    "InstanceMatchCriteria": "targeted",  
    "EphemeralStorage": false,  
    "CreateDate": "2019-08-16T10:21:57.000Z",  
    "AvailableInstanceCount": 3,  
    "InstancePlatform": "Linux/UNIX",  
    "TotalInstanceCount": 3,  
    "State": "active",  
    "Tenancy": "default",  
    "EbsOptimized": false,  
    "InstanceType": "m5.large"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Reservasi Kapasitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [CreateCapacityReservation](#) di Referensi AWS CLI Perintah.

## **create-carrier-gateway**

Contoh kode berikut menunjukkan cara menggunakan `create-carrier-gateway`.

### AWS CLI

Untuk membuat gateway operator

`create-carrier-gateway` Contoh berikut membuat gateway operator untuk VPC yang ditentukan.

```
aws ec2 create-carrier-gateway \  
--vpc-id vpc-0c529aEXAMPLE1111
```

**Output:**

```
{
  "CarrierGateway": {
    "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
    "VpcId": "vpc-0c529aEXAMPLE1111",
    "State": "pending",
    "OwnerId": "123456789012"
  }
}
```

Untuk informasi selengkapnya, lihat [Gateway operator](#) di Panduan Pengguna Wavelength AWS .

- Untuk detail API, lihat [CreateCarrierGateway](#) di Referensi AWS CLI Perintah.

**create-client-vpn-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `create-client-vpn-endpoint`.

**AWS CLI**

Untuk membuat titik akhir Client VPN

`create-client-vpn-endpoint` Contoh berikut membuat titik akhir Client VPN yang menggunakan otentikasi timbal balik dan menentukan nilai untuk blok CIDR klien.

```
aws ec2 create-client-vpn-endpoint \
  --client-cidr-block "172.31.0.0/16" \
  --server-certificate-arn arn:aws:acm:ap-south-1:123456789012:certificate/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
  --authentication-options Type=certificate-
authentication,MutualAuthentication={ClientRootCertificateChainArn=arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE} \
  --connection-log-options Enabled=false
```

**Output:**

```
{
  "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
  "Status": {
    "Code": "pending-associate"
  }
}
```

```
  },  
  "DnsName": "cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-  
south-1.amazonaws.com"  
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir Client VPN](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [CreateClientVpnEndpoint](#) di Referensi AWS CLI Perintah.

## create-client-vpn-route

Contoh kode berikut menunjukkan cara menggunakan `create-client-vpn-route`.

### AWS CLI

Untuk membuat rute untuk titik akhir Client VPN

`create-client-vpn-route` Contoh berikut menambahkan rute ke internet (`0.0.0.0/0`) untuk subnet yang ditentukan dari titik akhir Client VPN.

```
aws ec2 create-client-vpn-route \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --destination-cidr-block 0.0.0.0/0 \  
  --target-vpc-subnet-id subnet-0123456789abcabca
```

Output:

```
{  
  "Status": {  
    "Code": "creating"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Rute](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [CreateClientVpnRoute](#) di Referensi AWS CLI Perintah.

## create-coip-cidr

Contoh kode berikut menunjukkan cara menggunakan `create-coip-cidr`.

## AWS CLI

Untuk membuat berbagai alamat IP (CoIP) milik pelanggan

`create-coip-cidr` Contoh berikut menciptakan rentang tertentu alamat CoIP di kolam CoIP tertentu.

```
aws ec2 create-coip-cidr \  
  --cidr 15.0.0.0/24 \  
  --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Output:

```
{  
  "CoipCidr": {  
    "Cidr": "15.0.0.0/24",  
    "CoipPoolId": "ipv4pool-coip-1234567890abcdefg",  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Alamat IP milik pelanggan](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [CreateCoipCidr](#) di Referensi AWS CLI Perintah.

## `create-coip-pool`

Contoh kode berikut menunjukkan cara menggunakan `create-coip-pool`.

## AWS CLI

Untuk membuat kumpulan alamat IP (CoIP) milik pelanggan

`create-coip-pool` Contoh berikut membuat kumpulan CoIP untuk alamat CoIP dalam tabel rute gateway lokal yang ditentukan.

```
aws ec2 create-coip-pool \  
  --local-gateway-route-table-id lgw-rtb-abcdefg1234567890
```

Output:

```
{
  "CoipPool": {
    "PoolId": "ipv4pool-coip-1234567890abcdefg",
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
    "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-coip-1234567890abcdefg"
  }
}
```

Untuk informasi selengkapnya, lihat [Alamat IP milik pelanggan](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [CreateCoipPool](#) di Referensi AWS CLI Perintah.

## create-customer-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-customer-gateway`.

### AWS CLI

Untuk membuat gateway pelanggan

Contoh ini membuat gateway pelanggan dengan alamat IP yang ditentukan untuk antarmuka luarnya.

Perintah:

```
aws ec2 create-customer-gateway --type ipsec.1 --public-ip 12.1.2.3 --bgp-asn 65534
```

Output:

```
{
  "CustomerGateway": {
    "CustomerGatewayId": "cgw-0e11f167",
    "IpAddress": "12.1.2.3",
    "State": "available",
    "Type": "ipsec.1",
    "BgpAsn": "65534"
  }
}
```

- Untuk detail API, lihat [CreateCustomerGateway](#) di Referensi AWS CLI Perintah.

## create-default-subnet

Contoh kode berikut menunjukkan cara menggunakan `create-default-subnet`.

### AWS CLI

Untuk membuat subnet default

Contoh ini membuat subnet default di Availability Zone `us-east-2a`.

Perintah:

```
aws ec2 create-default-subnet --availability-zone us-east-2a
```

```
{
  "Subnet": {
    "AvailabilityZone": "us-east-2a",
    "Tags": [],
    "AvailableIpAddressCount": 4091,
    "DefaultForAz": true,
    "Ipv6CidrBlockAssociationSet": [],
    "VpcId": "vpc-1a2b3c4d",
    "State": "available",
    "MapPublicIpOnLaunch": true,
    "SubnetId": "subnet-1122aabb",
    "CidrBlock": "172.31.32.0/20",
    "AssignIpv6AddressOnCreation": false
  }
}
```

- Untuk detail API, lihat [CreateDefaultSubnet](#) di Referensi AWS CLI Perintah.

## create-default-vpc

Contoh kode berikut menunjukkan cara menggunakan `create-default-vpc`.

### AWS CLI

Untuk membuat VPC default

Contoh ini menciptakan VPC default.

Perintah:

```
aws ec2 create-default-vpc
```

Output:

```
{
  "Vpc": {
    "VpcId": "vpc-8eaae5ea",
    "InstanceTenancy": "default",
    "Tags": [],
    "Ipv6CidrBlockAssociationSet": [],
    "State": "pending",
    "DhcpOptionsId": "dopt-af0c32c6",
    "CidrBlock": "172.31.0.0/16",
    "IsDefault": true
  }
}
```

- Untuk detail API, lihat [CreateDefaultVpc](#) di Referensi AWS CLI Perintah.

## create-dhcp-options

Contoh kode berikut menunjukkan cara menggunakan `create-dhcp-options`.

### AWS CLI

Untuk membuat satu set opsi DHCP

`create-dhcp-options` Contoh berikut membuat satu set opsi DHCP yang menentukan nama domain, server nama domain, dan jenis node NetBIOS.

```
aws ec2 create-dhcp-options \
  --dhcp-configuration \
    "Key=domain-name-servers,Values=10.2.5.1,10.2.5.2" \
    "Key=domain-name,Values=example.com" \
    "Key=netbios-node-type,Values=2"
```

Output:

```
{
  "DhcpOptions": {
```

```
"DhcpConfigurations": [
  {
    "Key": "domain-name",
    "Values": [
      {
        "Value": "example.com"
      }
    ]
  },
  {
    "Key": "domain-name-servers",
    "Values": [
      {
        "Value": "10.2.5.1"
      },
      {
        "Value": "10.2.5.2"
      }
    ]
  },
  {
    "Key": "netbios-node-type",
    "Values": [
      {
        "Value": "2"
      }
    ]
  }
],
"DhcpOptionsId": "dopt-06d52773eff4c55f3"
}
```

- Untuk detail API, lihat [CreateDhcpOptions](#) di Referensi AWS CLI Perintah.

## **create-egress-only-internet-gateway**

Contoh kode berikut menunjukkan cara menggunakan `create-egress-only-internet-gateway`.

### AWS CLI

Untuk membuat gateway Internet khusus egres



Contoh ini membuat gateway Internet khusus egress untuk VPC yang ditentukan.

Perintah:

```
aws ec2 create-egress-only-internet-gateway --vpc-id vpc-0c62a468
```

Output:

```
{
  "EgressOnlyInternetGateway": {
    "EgressOnlyInternetGatewayId": "eigw-015e0e244e24dfe8a",
    "Attachments": [
      {
        "State": "attached",
        "VpcId": "vpc-0c62a468"
      }
    ]
  }
}
```

- Untuk detail API, lihat [CreateEgressOnlyInternetGateway](#) di Referensi AWS CLI Perintah.

## create-fleet

Contoh kode berikut menunjukkan cara menggunakan `create-fleet`.

### AWS CLI

Contoh 1: Untuk membuat Armada EC2 yang meluncurkan Instans Spot sebagai model pembelian default

`create-fleet` Contoh berikut membuat Armada EC2 menggunakan parameter minimum yang diperlukan untuk meluncurkan armada: template peluncuran, kapasitas target, dan model pembelian default. Templat peluncuran diidentifikasi dengan ID templat dan nomor versi peluncurannya. Kapasitas target untuk armada adalah 2 instance, dan model pembelian default adalah `spot`, yang menghasilkan armada meluncurkan 2 Instans Spot.

Saat Anda membuat EC2 Fleet, gunakan file JSON untuk menentukan informasi tentang instans yang akan diluncurkan.

```
aws ec2 create-fleet \
```

```
--cli-input-json file://file_name.json
```

Isi file\_name.json:

```
{
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
      }
    }
  ],
  "TargetCapacitySpecification": {
    "TotalTargetCapacity": 2,
    "DefaultTargetCapacityType": "spot"
  }
}
```

Output:

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Contoh 2: Untuk membuat Armada EC2 yang meluncurkan Instans Sesuai Permintaan sebagai model pembelian default

`create-fleet` Contoh berikut membuat Armada EC2 menggunakan parameter minimum yang diperlukan untuk meluncurkan armada: template peluncuran, kapasitas target, dan model pembelian default. Templat peluncuran diidentifikasi dengan ID templat dan nomor versi peluncurannya. Kapasitas target untuk armada adalah 2 instance, dan model pembelian default adalah on-demand, yang menghasilkan armada meluncurkan 2 Instans On-Demand.

Saat Anda membuat EC2 Fleet, gunakan file JSON untuk menentukan informasi tentang instans yang akan diluncurkan.

```
aws ec2 create-fleet \
  --cli-input-json file://file_name.json
```

Isi file\_name.json:

```
{
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
      }
    }
  ],
  "TargetCapacitySpecification": {
    "TotalTargetCapacity": 2,
    "DefaultTargetCapacityType": "on-demand"
  }
}
```

Output:

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Contoh 3: Untuk membuat Armada EC2 yang meluncurkan Instans Sesuai Permintaan sebagai kapasitas utama

`create-fleet` Contoh berikut membuat Armada EC2 yang menentukan total kapasitas target 2 instans untuk armada, dan kapasitas target 1 Instans On-Demand. Model pembelian default adalah `spot`. Armada meluncurkan 1 Instans Sesuai Permintaan seperti yang ditentukan, tetapi perlu meluncurkan satu instance lagi untuk memenuhi total kapasitas target. Model pembelian untuk selisihnya dihitung sebagai `TotalTargetCapacity - OnDemandTargetCapacity = DefaultTargetCapacityType`, yang menghasilkan armada meluncurkan 1 Instans Spot.

Saat Anda membuat EC2 Fleet, gunakan file JSON untuk menentukan informasi tentang instans yang akan diluncurkan.

```
aws ec2 create-fleet \
  --cli-input-json file:///file_name.json
```

Isi `file_name.json`:

```
{
```

```
"LaunchTemplateConfigs": [  
  {  
    "LaunchTemplateSpecification": {  
      "LaunchTemplateId": "lt-0e8c754449b27161c",  
      "Version": "1"  
    }  
  },  
],  
"TargetCapacitySpecification": {  
  "TotalTargetCapacity": 2,  
  "OnDemandTargetCapacity": 1,  
  "DefaultTargetCapacityType": "spot"  
}  
}
```

#### Output:

```
{  
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"  
}
```

Contoh 4: Untuk membuat Armada EC2 yang meluncurkan Instans Spot menggunakan strategi alokasi harga terendah

Jika strategi alokasi untuk Instans Spot tidak ditentukan, strategi alokasi default, yaitu `lowest-price`, akan digunakan. `create-fleet` Contoh berikut membuat Armada EC2 menggunakan strategi `lowest-price` alokasi. Tiga spesifikasi peluncuran, yang menimpa templat peluncuran, memiliki tipe instans berbeda tetapi kapasitas dan subnet berbobot sama. Total kapasitas target adalah 2 instance dan model pembelian default adalah `spot`. Armada EC2 meluncurkan 2 Instans Spot menggunakan tipe instans spesifikasi peluncuran dengan harga terendah.

Saat Anda membuat EC2 Fleet, gunakan file JSON untuk menentukan informasi tentang instans yang akan diluncurkan.

```
aws ec2 create-fleet \  
  --cli-input-json file:///file_name.jsonContents of file_name.json::  
  
{  
  "LaunchTemplateConfigs": [  
    {  
      "LaunchTemplateSpecification": {  
        "LaunchTemplateId": "lt-0e8c754449b27161c",
```

```
    "Version": "1"
  },
  "Overrides": [
    {
      "InstanceType": "c4.large",
      "WeightedCapacity": 1,
      "SubnetId": "subnet-a4f6c5d3"
    },
    {
      "InstanceType": "c3.large",
      "WeightedCapacity": 1,
      "SubnetId": "subnet-a4f6c5d3"
    },
    {
      "InstanceType": "c5.large",
      "WeightedCapacity": 1,
      "SubnetId": "subnet-a4f6c5d3"
    }
  ]
}
],
"TargetCapacitySpecification": {
  "TotalTargetCapacity": 2,
  "DefaultTargetCapacityType": "spot"
}
}
```

Output:

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

- Untuk detail API, lihat [CreateFleet](#) di Referensi AWS CLI Perintah.

## create-flow-logs

Contoh kode berikut menunjukkan cara menggunakan `create-flow-logs`.

### AWS CLI

Contoh 1: Untuk membuat log aliran

`create-flow-logs` Contoh berikut membuat log aliran yang menangkap semua lalu lintas ditolak untuk antarmuka jaringan yang ditentukan. Log aliran dikirim ke grup log di CloudWatch Log menggunakan izin dalam peran IAM yang ditentukan.

```
aws ec2 create-flow-logs \
  --resource-type NetworkInterface \
  --resource-ids eni-11223344556677889 \
  --traffic-type REJECT \
  --log-group-name my-flow-logs \
  --deliver-logs-permission-arn arn:aws:iam::123456789101:role/publishFlowLogs
```

Output:

```
{
  "ClientToken": "so0eNA2uSHUNlHI0S2cJ305GuIX1CezaRdGtexample",
  "FlowLogIds": [
    "fl-12345678901234567"
  ],
  "Unsuccessful": []
}
```

Untuk informasi selengkapnya, lihat [Log Alur VPC](#) di Panduan Pengguna Amazon VPC.

Contoh 2: Untuk membuat log alur dengan format kustom

`create-flow-logs` Contoh berikut membuat log alur yang menangkap semua lalu lintas untuk VPC yang ditentukan dan mengirimkan log aliran ke bucket Amazon S3. Parameter `--log-format` menentukan format kustom untuk catatan log alur. Untuk menjalankan perintah ini di Windows, ubah tanda kutip tunggal (') menjadi tanda kutip ganda (").

```
aws ec2 create-flow-logs \
  --resource-type VPC \
  --resource-ids vpc-00112233344556677 \
  --traffic-type ALL \
  --log-destination-type s3 \
  --log-destination arn:aws:s3:::flow-log-bucket/my-custom-flow-logs/ \
  --log-format '${version} ${vpc-id} ${subnet-id} ${instance-id} ${srcaddr}
  ${dstaddr} ${srcport} ${dstport} ${protocol} ${tcp-flags} ${type} ${pkt-srcaddr}
  ${pkt-dstaddr}'
```

Untuk informasi selengkapnya, lihat [Log Alur VPC](#) di Panduan Pengguna Amazon VPC.

Contoh 3: Untuk membuat log aliran dengan interval agregasi maksimum satu menit

`create-flow-logs` Contoh berikut membuat log alur yang menangkap semua lalu lintas untuk VPC yang ditentukan dan mengirimkan log aliran ke bucket Amazon S3. `--max-aggregation-interval` Parameter menentukan interval agregasi maksimum 60 detik (1 menit).

```
aws ec2 create-flow-logs \  
  --resource-type VPC \  
  --resource-ids vpc-00112233344556677 \  
  --traffic-type ALL \  
  --log-destination-type s3 \  
  --log-destination arn:aws:s3:::flow-log-bucket/my-custom-flow-logs/ \  
  --max-aggregation-interval 60
```

Untuk informasi selengkapnya, lihat [Log Alur VPC](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [CreateFlowLogs](#) di Referensi AWS CLI Perintah.

## `create-fpga-image`

Contoh kode berikut menunjukkan cara menggunakan `create-fpga-image`.

### AWS CLI

Untuk membuat gambar Amazon FPGA

Contoh ini membuat AFI dari tarball yang ditentukan dalam ember yang ditentukan.

Perintah:

```
aws ec2 create-fpga-image --name my-afi --description test-afi --input-storage-  
location Bucket=my-fpga-bucket,Key=dcp/17_12_22-103226.Developer_CL.tar --logs-  
storage-location Bucket=my-fpga-bucket,Key=logs
```

Output:

```
{  
  "FpgaImageId": "afi-0d123e123bfc85abc",  
  "FpgaImageGlobalId": "agfi-123cb27b5e84a0abc"  
}
```

- Untuk detail API, lihat [CreateFpgaImage](#) di Referensi AWS CLI Perintah.

## create-image

Contoh kode berikut menunjukkan cara menggunakan `create-image`.

### AWS CLI

Contoh 1: Untuk membuat AMI dari instans yang didukung Amazon EBS

`create-image` Contoh berikut membuat AMI dari instance yang ditentukan.

```
aws ec2 create-image \  
  --instance-id i-1234567890abcdef0 \  
  --name "My server" \  
  --description "An AMI for my server"
```

Output:

```
{  
  "ImageId": "ami-abcdef01234567890"  
}
```

Untuk informasi selengkapnya tentang menentukan pemetaan perangkat blokir untuk AMI Anda, lihat [Menentukan pemetaan perangkat blokir untuk AMI di Panduan Pengguna Amazon EC2](#).

Contoh 2: Untuk membuat AMI dari instans yang didukung Amazon EBS tanpa reboot

`create-image` Contoh berikut membuat AMI dan menetapkan parameter `--no-reboot`, sehingga instance tidak di-reboot sebelum gambar dibuat.

```
aws ec2 create-image \  
  --instance-id i-1234567890abcdef0 \  
  --name "My server" \  
  --no-reboot
```

Output:

```
{  
  "ImageId": "ami-abcdef01234567890"  
}
```

Untuk informasi selengkapnya tentang menentukan pemetaan perangkat blokir untuk AMI Anda, lihat [Menentukan pemetaan perangkat blokir untuk AMI di Panduan Pengguna Amazon EC2](#).



### Contoh 3: Untuk menandai AMI dan snapshot pada pembuatan

`create-image` Contoh berikut membuat AMI, dan menandai AMI dan snapshot dengan tag yang sama `cost-center=cc123`

```
aws ec2 create-image \  
  --instance-id i-1234567890abcdef0 \  
  --name "My server" \  
  --tag-specifications "ResourceType=image,Tags=[{Key=cost-center,Value=cc123}]" \  
  "ResourceType=snapshot,Tags=[{Key=cost-center,Value=cc123}]"
```

Output:

```
{  
  "ImageId": "ami-abcdef01234567890"  
}
```

Untuk informasi selengkapnya tentang menandai sumber daya Anda saat pembuatan, lihat [Menambahkan tag pada pembuatan sumber daya](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CreateImage](#) di Referensi AWS CLI Perintah.

### **create-instance-connect-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `create-instance-connect-endpoint`.

#### AWS CLI

Untuk membuat Instance Connect Endpoint EC2

`create-instance-connect-endpoint` Contoh berikut membuat EC2 Instance Connect Endpoint di subnet yang ditentukan.

```
aws ec2 create-instance-connect-endpoint \  
  --region us-east-1 \  
  --subnet-id subnet-0123456789example
```

Output:

```
{
```

```
"VpcId": "vpc-0123abcd",
  "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
  "AvailabilityZone": "us-east-1a",
  "NetworkInterfaceIds": [
    "eni-0123abcd"
  ],
  "PreserveClientIp": true,
  "Tags": [],
  "FipsDnsName": "eice-0123456789example.0123abcd.fips.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
  "StateMessage": "",
  "State": "create-complete",
  "DnsName": "eice-0123456789example.0123abcd.ec2-instance-connect-endpoint.us-
east-1.amazonaws.com",
  "SubnetId": "subnet-0123abcd",
  "OwnerId": "111111111111",
  "SecurityGroupIds": [
    "sg-0123abcd"
  ],
  "InstanceConnectEndpointId": "eice-0123456789example",
  "CreatedAt": "2023-04-07T15:43:53.000Z"
}
```

Untuk informasi selengkapnya, lihat [Membuat Titik Akhir Connect Instans EC2](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CreateInstanceConnectEndpoint](#) di Referensi AWS CLI Perintah.

## **create-instance-event-window**

Contoh kode berikut menunjukkan cara menggunakan `create-instance-event-window`.

### AWS CLI

Contoh 1: Untuk membuat jendela acara dengan rentang waktu

`create-instance-event-window` Contoh berikut membuat jendela acara dengan rentang waktu. Anda juga tidak dapat menentukan parameter `cron-expression`.

```
aws ec2 create-instance-event-window \
  --region us-east-1 \
  --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8 \
```

```
--tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
--name myEventWindowName
```

## Output:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "TimeRanges": [
      {
        "StartWeekDay": "monday",
        "StartHour": 2,
        "EndWeekDay": "wednesday",
        "EndHour": 8
      }
    ],
    "Name": "myEventWindowName",
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}
```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 2: Untuk membuat jendela acara dengan ekspresi cron

`create-instance-event-window` Contoh berikut membuat jendela acara dengan ekspresi cron. Anda juga tidak dapat menentukan parameter `time-range`.

```
aws ec2 create-instance-event-window \
--region us-east-1 \
--cron-expression "* 21-23 * * 2,3" \
--tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
--name myEventWindowName
```

**Output:**

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}
```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CreateInstanceEventWindow](#) di Referensi AWS CLI Perintah.

**create-instance-export-task**

Contoh kode berikut menunjukkan cara menggunakan `create-instance-export-task`.

**AWS CLI**

Untuk mengekspor sebuah instance

Perintah contoh ini membuat tugas untuk mengekspor instance `i-1234567890abcdef0` ke bucket Amazon S3 `myexportbucket`.

Perintah:

```
aws ec2 create-instance-export-task --description "RHEL5 instance" --instance-id i-1234567890abcdef0 --target-environment vmware --export-to-s3-task DiskImageFormat=vmdk,ContainerFormat=ova,S3Bucket=myexportbucket,S3Prefix=RHEL5
```

**Output:**

```
{
```

```

"ExportTask": {
  "State": "active",
  "InstanceExportDetails": {
    "InstanceId": "i-1234567890abcdef0",
    "TargetEnvironment": "vmware"
  },
  "ExportToS3Task": {
    "S3Bucket": "myexportbucket",
    "S3Key": "RHEL5export-i-fh8sjjsq.ova",
    "DiskImageFormat": "vmdk",
    "ContainerFormat": "ova"
  },
  "Description": "RHEL5 instance",
  "ExportTaskId": "export-i-fh8sjjsq"
}
}

```

- Untuk detail API, lihat [CreateInstanceExportTask](#) di Referensi AWS CLI Perintah.

## create-internet-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-internet-gateway`.

### AWS CLI

Untuk membuat gateway internet

`create-internet-gateway` Contoh berikut membuat gateway internet dengan `tagName=my-igw`.

```

aws ec2 create-internet-gateway \
  --tag-specifications ResourceType=internet-gateway,Tags=[{Key=Name,Value=my-igw}]

```

Output:

```

{
  "InternetGateway": {
    "Attachments": [],
    "InternetGatewayId": "igw-0d0fb496b3994d755",
    "OwnerId": "123456789012",
    "Tags": [

```

```

    {
      "Key": "Name",
      "Value": "my-igw"
    }
  ]
}

```

Untuk informasi lebih lanjut, lihat [Gateway internet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [CreateInternetGateway](#) di Referensi AWS CLI Perintah.

## create-ipam-pool

Contoh kode berikut menunjukkan cara menggunakan `create-ipam-pool`.

### AWS CLI

Untuk membuat kolam IPAM

`create-ipam-pool` Contoh berikut membuat kolam IPAM.

(Linux):

```

aws ec2 create-ipam-pool \
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
  --address-family ipv4 \
  --auto-import \
  --allocation-min-netmask-length 16 \
  --allocation-max-netmask-length 26 \
  --allocation-default-netmask-length 24 \
  --allocation-resource-tags "Key=Environment,Value=Preprod" \
  --tag-specifications 'ResourceType=ipam-pool,Tags=[{Key=Name,Value="Preprod pool"}]'
```

(Windows):

```

aws ec2 create-ipam-pool ^
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
  --address-family ipv4 ^
  --auto-import ^
  --allocation-min-netmask-length 16 ^
  --allocation-max-netmask-length 26 ^
```

```
--allocation-default-netmask-length 24 ^
--allocation-resource-tags "Key=Environment,Value=Preprod" ^
--tag-specifications ResourceType=ipam-pool,Tags=[{Key=Name,Value="Preprod
pool"}]
```

### Output:

```
{
  "IpamPool": {
    "OwnerId": "123456789012",
    "IpamPoolId": "ipam-pool-0533048da7d823723",
    "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0533048da7d823723",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
    "IpamScopeType": "private",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "Locale": "None",
    "PoolDepth": 1,
    "State": "create-in-progress",
    "AutoImport": true,
    "AddressFamily": "ipv4",
    "AllocationMinNetmaskLength": 16,
    "AllocationMaxNetmaskLength": 26,
    "AllocationDefaultNetmaskLength": 24,
    "AllocationResourceTags": [
      {
        "Key": "Environment",
        "Value": "Preprod"
      }
    ],
    "Tags": [
      {
        "Key": "Name",
        "Value": "Preprod pool"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Merencanakan penyediaan alamat IP di Panduan Pengguna Amazon VPC IPAM](#).

- Untuk detail API, lihat [CreateIpamPool](#) di Referensi AWS CLI Perintah.

## **create-ipam-resource-discovery**

Contoh kode berikut menunjukkan cara menggunakan `create-ipam-resource-discovery`.

### AWS CLI

Untuk membuat penemuan sumber daya

Dalam contoh ini, Anda adalah admin IPAM yang didelegasikan yang ingin membuat dan berbagi penemuan sumber daya dengan admin IPAM di AWS Organisasi lain sehingga admin di organisasi lain dapat mengelola dan memantau alamat IP sumber daya di organisasi Anda.

### Penting

Contoh ini mencakup `--operating-regions` opsi `--region` dan opsi karena, meskipun opsional, mereka harus dikonfigurasi dengan cara tertentu agar berhasil mengintegrasikan penemuan sumber daya dengan IPAM. \* `--operating-regions` harus cocok dengan Wilayah tempat Anda memiliki sumber daya yang ingin ditemukan IPAM. Jika ada Wilayah di mana Anda tidak ingin IPAM mengelola alamat IP (misalnya untuk alasan kepatuhan), jangan sertakan. \* `--region` harus cocok dengan Wilayah asal IPAM yang ingin Anda kaitkan. Anda harus membuat penemuan sumber daya di Wilayah yang sama tempat IPAM dibuat. Misalnya, jika IPAM yang Anda kaitkan dengan dibuat di `us-east-1`, sertakan `--region us-east-1` dalam request. Both `--region` dan `--operating-regions` opsi default ke Wilayah tempat Anda menjalankan perintah jika Anda tidak menentukannya.

Dalam contoh ini, Wilayah operasi IPAM yang kami integrasikan dengan `includeus-west-1,us-west-2`, dan `ap-south-1` Saat kami membuat penemuan sumber daya, kami ingin IPAM menemukan alamat IP sumber daya di `us-west-1` dan `us-west-2` tetapi tidak `ap-south-1`. Jadi kami hanya memasukkan `--operating-regions RegionName='us-west-1' RegionName='us-west-2'` dalam permintaan.

`create-ipam-resource-discovery` Contoh berikut menciptakan penemuan sumber daya IPAM.

```
aws ec2 create-ipam-resource-discovery \
  --description 'Example-resource-discovery' \
  --tag-specifications 'ResourceType=ipam-resource-discovery,Tags=[{Key=cost-center,Value=cc123}]' \
```



```
--operating-regions RegionName='us-west-1' RegionName='us-west-2' \  
--region us-east-1
```

### Output:

```
{  
  "IpamResourceDiscovery":{  
    "OwnerId": "149977607591",  
    "IpamResourceDiscoveryId": "ipam-res-disco-0257046d8aa78b8bc",  
    "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-  
discovery/ipam-res-disco-0257046d8aa78b8bc",  
    "IpamResourceDiscoveryRegion": "us-east-1",  
    "Description": "'Example-resource-discovery'",  
    "OperatingRegions":[  
      {"RegionName": "us-west-1"},  
      {"RegionName": "us-west-2"},  
      {"RegionName": "us-east-1"}  
    ],  
    "IsDefault": false,  
    "State": "create-in-progress",  
    "Tags": [  
      {  
        "Key": "cost-center",  
        "Value": "cc123"  
      }  
    ]  
  }  
}
```

Setelah Anda membuat penemuan sumber daya, Anda mungkin ingin membagikannya dengan admin delegasi IPAM lain, yang dapat Anda lakukan. [create-resource-share](#) Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [CreateIpamResourceDiscovery](#) di Referensi AWS CLI Perintah.

## create-ipam-scope

Contoh kode berikut menunjukkan cara menggunakan `create-ipam-scope`.

### AWS CLI

Untuk membuat cakupan IPAM

`create-ipam-scope` Contoh berikut menciptakan lingkup IPAM.

(Linux):

```
aws ec2 create-ipam-scope \  
  --ipam-id ipam-08440e7a3acde3908 \  
  --description "Example description" \  
  --tag-specifications 'ResourceType=ipam-scope,Tags=[{Key=Name,Value="Example  
name value"}]'
```

(Windows):

```
aws ec2 create-ipam-scope ^  
  --ipam-id ipam-08440e7a3acde3908 ^  
  --description "Example description" ^  
  --tag-specifications ResourceType=ipam-scope,Tags=[{Key=Name,Value="Example name  
value"}]
```

Output:

```
{  
  "IpamScope": {  
    "OwnerId": "123456789012",  
    "IpamScopeId": "ipam-scope-01c1ebab2b63bd7e4",  
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-  
scope-01c1ebab2b63bd7e4",  
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",  
    "IpamRegion": "us-east-1",  
    "IpamScopeType": "private",  
    "IsDefault": false,  
    "Description": "Example description",  
    "PoolCount": 0,  
    "State": "create-in-progress",  
    "Tags": [  
      {  
        "Key": "Name",  
        "Value": "Example name value"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat cakupan tambahan](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [CreateIpamScope](#) di Referensi AWS CLI Perintah.

## create-ipam

Contoh kode berikut menunjukkan cara menggunakan `create-ipam`.

### AWS CLI

Untuk membuat IPAM

`create-ipam` Contoh berikut menciptakan IPAM.

(Linux):

```
aws ec2 create-ipam \  
  --description "Example description" \  
  --operating-regions "RegionName=us-east-2" "RegionName=us-west-1" \  
  --tag-specifications 'ResourceType=ipam,Tags=[{Key=Name,Value=ExampleIPAM}]'
```

(Windows):

```
aws ec2 create-ipam ^  
  --description "Example description" ^  
  --operating-regions "RegionName=us-east-2" "RegionName=us-west-1" ^  
  --tag-specifications ResourceType=ipam,Tags=[{Key=Name,Value=ExampleIPAM}]
```

Output:

```
{  
  "Ipam": {  
    "OwnerId": "123456789012",  
    "IpamId": "ipam-036486dfa6af58ee0",  
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-036486dfa6af58ee0",  
    "IpamRegion": "us-east-1",  
    "PublicDefaultScopeId": "ipam-scope-071b8042b0195c183",  
    "PrivateDefaultScopeId": "ipam-scope-0807405dece705a30",  
    "ScopeCount": 2,  
    "OperatingRegions": [  
      {  
        "RegionName": "us-east-2"      }  
    ]  
  }  
}
```

```
    },
    {
      "RegionName": "us-west-1"
    },
    {
      "RegionName": "us-east-1"
    }
  ],
  "State": "create-in-progress",
  "Tags": [
    {
      "Key": "Name",
      "Value": "ExampleIPAM"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat IPAM](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [Createlpam](#) di Referensi AWS CLI Perintah.

## create-key-pair

Contoh kode berikut menunjukkan cara menggunakan `create-key-pair`.

### AWS CLI

Untuk membuat pasangan kunci

Contoh ini membuat pasangan kunci bernama `MyKeyPair`.

Perintah:

```
aws ec2 create-key-pair --key-name MyKeyPair
```

Output-nya adalah versi ASCII dari kunci privat dan sidik jari kunci. Anda perlu menyimpan kunci ke file.

Untuk informasi selengkapnya, lihat [Menggunakan Pasangan Kunci](#) di Panduan Pengguna Antarmuka Baris Perintah AWS .

- Untuk detail API, lihat [CreateKeyPair](#) di Referensi AWS CLI Perintah.

## create-launch-template-version

Contoh kode berikut menunjukkan cara menggunakan `create-launch-template-version`.

### AWS CLI

Untuk membuat versi template peluncuran

Contoh ini membuat versi template peluncuran baru berdasarkan versi 1 dari template peluncuran dan menentukan ID AMI yang berbeda.

Perintah:

```
aws ec2 create-launch-template-version --launch-template-id lt-0abcd290751193123
--version-description WebVersion2 --source-version 1 --launch-template-data
'{"ImageId":"ami-c998b6b2"}'
```

Output:

```
{
  "LaunchTemplateVersion": {
    "VersionDescription": "WebVersion2",
    "LaunchTemplateId": "lt-0abcd290751193123",
    "LaunchTemplateName": "WebServers",
    "VersionNumber": 2,
    "CreatedBy": "arn:aws:iam::123456789012:root",
    "LaunchTemplateData": {
      "ImageId": "ami-c998b6b2",
      "InstanceType": "t2.micro",
      "NetworkInterfaces": [
        {
          "Ipv6Addresses": [
            {
              "Ipv6Address": "2001:db8:1234:1a00::123"
            }
          ],
          "DeviceIndex": 0,
          "SubnetId": "subnet-7b16de0c",
          "AssociatePublicIpAddress": true
        }
      ]
    },
    "DefaultVersion": false,
    "CreateTime": "2017-12-01T13:35:46.000Z"
  }
}
```

```
}
}
```

- Untuk detail API, lihat [CreateLaunchTemplateVersion](#) di Referensi AWS CLI Perintah.

## create-launch-template

Contoh kode berikut menunjukkan cara menggunakan `create-launch-template`.

### AWS CLI

Contoh 1: Untuk membuat templat peluncuran

Contoh `create-launch-template` berikut membuat templat peluncuran yang menentukan subnet untuk meluncurkan instans, menetapkan alamat IP publik dan alamat IPv6 ke instans, lalu membuat tanda untuk instans.

```
aws ec2 create-launch-template \
  --launch-template-name TemplateForWebServer \
  --version-description WebVersion1 \
  --launch-template-data '{"NetworkInterfaces":
[{"AssociatePublicIpAddress":true,"DeviceIndex":0,"Ipv6AddressCount":1,"SubnetId":"subnet-7b
[{"ResourceType":"instance","Tags":[{"Key":"purpose","Value":"webserver"}]}]}'
```

Output:

```
{
  "LaunchTemplate": {
    "LatestVersionNumber": 1,
    "LaunchTemplateId": "lt-01238c059e3466abc",
    "LaunchTemplateName": "TemplateForWebServer",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
    "CreateTime": "2019-01-27T09:13:24.000Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Meluncurkan Instans dari Templat Peluncuran](#) di Panduan Pengguna Amazon Elastic Compute Cloud. Untuk informasi tentang mengutip parameter berformat JSON, lihat [Mengutip String di Panduan Pengguna Antarmuka Baris Perintah AWS](#).

Contoh 2: Guna membuat templat peluncuran untuk Amazon EC2 Auto Scaling

Contoh `create-launch-template` berikut membuat templat peluncuran dengan banyak tanda dan pemetaan perangkat blok untuk menentukan volume EBS tambahan saat instans diluncurkan. Tentukan nilai untuk Groups yang sesuai dengan grup keamanan untuk VPC yang akan menjadi tujuan peluncuran instans oleh grup Auto Scaling Anda. Tentukan VPC dan subnet sebagai properti grup Auto Scaling.

```
aws ec2 create-launch-template \
  --launch-template-name TemplateForAutoScaling \
  --version-description AutoScalingVersion1 \
  --launch-template-data '{"NetworkInterfaces":
[{"DeviceIndex":0,"AssociatePublicIpAddress":true,"Groups":
[{"sg-7c227019,sg-903004f8"},"DeleteOnTermination":true}], "ImageId":"ami-
b42209de","InstanceType":"m4.large","TagSpecifications":
[{"ResourceType":"instance","Tags":[{"Key":"environment","Value":"production"},
{"Key":"purpose","Value":"webserver"}]},{"ResourceType":"volume","Tags":
[{"Key":"environment","Value":"production"}, {"Key":"cost-
center","Value":"cc123"}]}],"BlockDeviceMappings":[{"DeviceName":"/dev/sda1","Ebs":
{"VolumeSize":100}}]}' --region us-east-1
```

Output:

```
{
  "LaunchTemplate": {
    "LatestVersionNumber": 1,
    "LaunchTemplateId": "lt-0123c79c33a54e0abc",
    "LaunchTemplateName": "TemplateForAutoScaling",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
    "CreateTime": "2019-04-30T18:16:06.000Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Templat Peluncuran untuk Grup Auto Scaling](#) di [Panduan Pengguna Amazon EC2 Auto Scaling](#). Untuk informasi tentang mengutip parameter berformat JSON, lihat [Mengutip String di Panduan Pengguna Antarmuka Baris Perintah AWS](#).

Contoh 3: Untuk membuat templat peluncuran yang menentukan enkripsi volume EBS

Contoh `create-launch-template` berikut membuat templat peluncuran yang menyertakan volume EBS terenkripsi yang dibuat dari snapshot yang tidak terenkripsi. Contoh tersebut juga menandai volume selama pembuatan. Jika enkripsi secara default dinonaktifkan, Anda

harus menentukan opsi "Encrypted" seperti yang ditunjukkan pada contoh berikut. Jika menggunakan opsi "KmsKeyId" untuk menentukan CMK yang dikelola pelanggan, Anda juga harus menentukan opsi "Encrypted" bahkan jika enkripsi secara default diaktifkan.

```
aws ec2 create-launch-template \  
  --launch-template-name TemplateForEncryption \  
  --launch-template-data file://config.json
```

Isi dari config.json:

```
{  
  "BlockDeviceMappings":[  
    {  
      "DeviceName":"/dev/sda1",  
      "Ebs":{  
        "VolumeType":"gp2",  
        "DeleteOnTermination":true,  
        "SnapshotId":"snap-066877671789bd71b",  
        "Encrypted":true,  
        "KmsKeyId":"arn:aws:kms:us-east-1:012345678910:key/abcd1234-  
a123-456a-a12b-a123b4cd56ef"  
      }  
    }  
  ],  
  "ImageId":"ami-00068cd7555f543d5",  
  "InstanceType":"c5.large",  
  "TagSpecifications":[  
    {  
      "ResourceType":"volume",  
      "Tags":[  
        {  
          "Key":"encrypted",  
          "Value":"yes"  
        }  
      ]  
    }  
  ]  
}
```

Output:

```
{
```



```

"LaunchTemplate": {
  "LatestVersionNumber": 1,
  "LaunchTemplateId": "lt-0d5bd51bcf8530abc",
  "LaunchTemplateName": "TemplateForEncryption",
  "DefaultVersionNumber": 1,
  "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
  "CreateTime": "2020-01-07T19:08:36.000Z"
}
}

```

Untuk informasi selengkapnya, lihat Memulihkan Volume Amazon EBS dari Snapshot dan Enkripsi secara Default di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [CreateLaunchTemplate](#) di Referensi AWS CLI Perintah.

## create-local-gateway-route-table-virtual-interface-group-association

Contoh kode berikut menunjukkan cara menggunakan `create-local-gateway-route-table-virtual-interface-group-association`.

### AWS CLI

Untuk mengaitkan tabel rute gateway lokal dengan grup antarmuka virtual (VIF)

`create-local-gateway-route-table-virtual-interface-group-association` Contoh berikut menciptakan hubungan antara tabel rute gateway lokal yang ditentukan dan grup VIF.

```

aws ec2 create-local-gateway-route-table-virtual-interface-group-association \
  --local-gateway-route-table-id lgw-rtb-exampleidabcd1234 \
  --local-gateway-virtual-interface-group-id lgw-vif-grp-exampleid0123abcd

```

### Output:

```

{
  "LocalGatewayRouteTableVirtualInterfaceGroupAssociation": {
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-
assoc-exampleid12345678",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-exampleid0123abcd",
    "LocalGatewayId": "lgw-exampleid11223344",
    "LocalGatewayRouteTableId": "lgw-rtb-exampleidabcd1234",

```

```

    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-exampleidabcd1234",
    "OwnerId": "111122223333",
    "State": "pending",
    "Tags": []
  }
}

```

Untuk informasi selengkapnya, lihat [Asosiasi grup VIF](#) di Panduan AWS Pengguna Outposts.

- Untuk detail API, lihat [CreateLocalGatewayRouteTableVirtualInterfaceGroupAssociation](#) di Referensi AWS CLI Perintah.

### **create-local-gateway-route-table-vpc-association**

Contoh kode berikut menunjukkan cara menggunakan `create-local-gateway-route-table-vpc-association`.

#### AWS CLI

Untuk mengaitkan VPC dengan tabel rute

`create-local-gateway-route-table-vpc-association` Contoh berikut mengaitkan VPC tertentu dengan tabel rute gateway lokal yang ditentukan.

```

aws ec2 create-local-gateway-route-table-vpc-association \
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE \
  --vpc-id vpc-07ef66ac71EXAMPLE

```

Output:

```

{
  "LocalGatewayRouteTableVpcAssociation": {
    "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-0ee765bcc8EXAMPLE",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
    "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
    "VpcId": "vpc-07ef66ac71EXAMPLE",
    "State": "associated"
  }
}

```

- Untuk detail API, lihat [CreateLocalGatewayRouteTableVpcAssociation](#) di Referensi AWS CLI Perintah.

## create-local-gateway-route-table

Contoh kode berikut menunjukkan cara menggunakan `create-local-gateway-route-table`.

### AWS CLI

Untuk membuat tabel rute gateway lokal

`create-local-gateway-route-table` Contoh berikut membuat tabel rute gateway lokal dengan modus routing VPC langsung.

```
aws ec2 create-local-gateway-route-table \  
  --local-gateway-id lgw-1a2b3c4d5e6f7g8h9 \  
  --mode direct-vpc-routing
```

Output:

```
{  
  "LocalGatewayRouteTable": {  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",  
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-gateway-route-table/lgw-rtb-abcdefg1234567890",  
    "LocalGatewayId": "lgw-1a2b3c4d5e6f7g8h9",  
    "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/op-021345abcdef67890",  
    "OwnerId": "111122223333",  
    "State": "pending",  
    "Tags": [],  
    "Mode": "direct-vpc-routing"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway lokal](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [CreateLocalGatewayRouteTable](#) di Referensi AWS CLI Perintah.

## create-local-gateway-route

Contoh kode berikut menunjukkan cara menggunakan `create-local-gateway-route`.

### AWS CLI

Untuk membuat rute statis untuk tabel rute gateway lokal

`create-local-gateway-route` Contoh berikut membuat rute yang ditentukan dalam tabel rute gateway lokal yang ditentukan.

```
aws ec2 create-local-gateway-route \
  --destination-cidr-block 0.0.0.0/0 \
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE
```

Output:

```
{
  "Route": {
    "DestinationCidrBlock": "0.0.0.0/0",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
    "Type": "static",
    "State": "deleted",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE"
  }
}
```

- Untuk detail API, lihat [CreateLocalGatewayRoute](#) di Referensi AWS CLI Perintah.

## create-managed-prefix-list

Contoh kode berikut menunjukkan cara menggunakan `create-managed-prefix-list`.

### AWS CLI

Untuk membuat daftar awalan

`create-managed-prefix-list` Contoh berikut membuat daftar awalan IPv4 dengan maksimal 10 entri, dan membuat 2 entri dalam daftar awalan.

```
aws ec2 create-managed-prefix-list \
```

```
--address-family IPv4 \  
--max-entries 10 \  
--entries Cidr=10.0.0.0/16,Description=vpc-a Cidr=10.2.0.0/16,Description=vpc-b \  
\  
--prefix-list-name vpc-cidrs
```

### Output:

```
{  
  "PrefixList": {  
    "PrefixListId": "pl-0123456abcabcabc1",  
    "AddressFamily": "IPv4",  
    "State": "create-in-progress",  
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/  
pl-0123456abcabcabc1",  
    "PrefixListName": "vpc-cidrs",  
    "MaxEntries": 10,  
    "Version": 1,  
    "Tags": [],  
    "OwnerId": "123456789012"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Daftar awalan terkelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [CreateManagedPrefixList](#) di Referensi AWS CLI Perintah.

## create-nat-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-nat-gateway`.

### AWS CLI

Contoh 1: Untuk membuat gateway NAT publik

`create-nat-gateway` Contoh berikut membuat gateway NAT publik di subnet yang ditentukan dan mengaitkan alamat IP Elastis dengan ID alokasi yang ditentukan. Saat Anda membuat gateway NAT publik, Anda harus mengaitkan alamat IP Elastis.

```
aws ec2 create-nat-gateway \  
  --subnet-id subnet-0250c25a1fEXAMPLE \  
  --allocation-id eipalloc-09ad461b0dEXAMPLE
```

**Output:**

```
{
  "NatGateway": {
    "CreateTime": "2021-12-01T22:22:38.000Z",
    "NatGatewayAddresses": [
      {
        "AllocationId": "eipalloc-09ad461b0dEXAMPLE"
      }
    ],
    "NatGatewayId": "nat-0c61bf8a12EXAMPLE",
    "State": "pending",
    "SubnetId": "subnet-0250c25a1fEXAMPLE",
    "VpcId": "vpc-0a60eb65b4EXAMPLE",
    "ConnectivityType": "public"
  }
}
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.

**Contoh 2: Untuk membuat gateway NAT pribadi**

`create-nat-gateway` Contoh berikut membuat gateway NAT pribadi di subnet yang ditentukan. Gateway NAT pribadi tidak memiliki alamat IP Elastis terkait.

```
aws ec2 create-nat-gateway \
  --subnet-id subnet-0250c25a1fEXAMPLE \
  --connectivity-type private
```

**Output:**

```
{
  "NatGateway": {
    "CreateTime": "2021-12-01T22:26:00.000Z",
    "NatGatewayAddresses": [
      {}
    ],
    "NatGatewayId": "nat-011b568379EXAMPLE",
    "State": "pending",
    "SubnetId": "subnet-0250c25a1fEXAMPLE",
    "VpcId": "vpc-0a60eb65b4EXAMPLE",
    "ConnectivityType": "private"
  }
}
```

```
}
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [CreateNatGateway](#) di Referensi AWS CLI Perintah.

## **create-network-acl-entry**

Contoh kode berikut menunjukkan cara menggunakan `create-network-acl-entry`.

### AWS CLI

Untuk membuat entri ACL jaringan

Contoh ini membuat entri untuk ACL jaringan tertentu. Aturan ini memungkinkan masuknya lalu lintas dari alamat IPv4 (0.0.0.0/0) pada port UDP 53 (DNS) ke subnet terkait. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 create-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100 --protocol udp --port-range From=53,To=53 --cidr-block 0.0.0.0/0 --rule-action allow
```

Contoh ini membuat aturan untuk ACL jaringan tertentu yang memungkinkan lalu lintas masuk dari alamat IPv6 (:: /0) pada port TCP 80 (HTTP).

Perintah:

```
aws ec2 create-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 120 --protocol tcp --port-range From=80,To=80 --ipv6-cidr-block ::/0 --rule-action allow
```

- Untuk detail API, lihat [CreateNetworkAcEntry](#) di Referensi AWS CLI Perintah.

## **create-network-acl**

Contoh kode berikut menunjukkan cara menggunakan `create-network-acl`.

### AWS CLI

Untuk membuat jaringan ACL

Contoh ini membuat ACL jaringan untuk VPC tertentu.

Perintah:

```
aws ec2 create-network-acl --vpc-id vpc-a01106c2
```

Output:

```
{
  "NetworkAcl": {
    "Associations": [],
    "NetworkAclId": "acl-5fb85d36",
    "VpcId": "vpc-a01106c2",
    "Tags": [],
    "Entries": [
      {
        "CidrBlock": "0.0.0.0/0",
        "RuleNumber": 32767,
        "Protocol": "-1",
        "Egress": true,
        "RuleAction": "deny"
      },
      {
        "CidrBlock": "0.0.0.0/0",
        "RuleNumber": 32767,
        "Protocol": "-1",
        "Egress": false,
        "RuleAction": "deny"
      }
    ],
    "IsDefault": false
  }
}
```

- Untuk detail API, lihat [CreateNetworkAcl](#) di Referensi AWS CLI Perintah.

## **create-network-insights-access-scope**

Contoh kode berikut menunjukkan cara menggunakan `create-network-insights-access-scope`.



## AWS CLI

Untuk membuat Network Access Scope

`create-network-insights-access-scope` Contoh berikut membuat Network Access Scope.

```
aws ec2 create-network-insights-access-scope \  
  --cli-input-json file://access-scope-file.json
```

Isi dari `access-scope-file.json`:

```
{  
  "MatchPaths": [  
    {  
      "Source": {  
        "ResourceStatement": {  
          "Resources": [  
            "vpc-abcd12e3"  
          ]  
        }  
      }  
    }  
  ],  
  "ExcludePaths": [  
    {  
      "Source": {  
        "ResourceStatement": {  
          "ResourceTypes": [  
            "AWS::EC2::InternetGateway"  
          ]  
        }  
      }  
    }  
  ]  
}
```

Output:

```
{  
  "NetworkInsightsAccessScope": {  
    "NetworkInsightsAccessScopeId": "nis-123456789abc01234",  
    "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-  
east-1:123456789012:network-insights-access-scope/nis-123456789abc01234",
```

```

    "CreateDate": "2022-01-25T19:20:28.796000+00:00",
    "UpdatedDate": "2022-01-25T19:20:28.797000+00:00"
  },
  "NetworkInsightsAccessScopeContent": {
    "NetworkInsightsAccessScopeId": "nis-123456789abc01234",
    "MatchPaths": [
      {
        "Source": {
          "ResourceStatement": {
            "Resources": [
              "vpc-abcd12e3"
            ]
          }
        }
      }
    ],
    "ExcludePaths": [
      {
        "Source": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::InternetGateway"
            ]
          }
        }
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [CreateNetworkInsightsAccessScope](#) di Referensi AWS CLI Perintah.

## create-network-insights-path

Contoh kode berikut menunjukkan cara menggunakan `create-network-insights-path`.

### AWS CLI

Untuk membuat jalur

`create-network-insights-path` Contoh berikut menciptakan jalur. Sumbernya adalah gateway internet yang ditentukan dan tujuannya adalah instance EC2 yang ditentukan. Untuk menentukan apakah tujuan dapat dijangkau menggunakan protokol dan port yang ditentukan, analisis jalur menggunakan perintah. `start-network-insights-analysis`

```
aws ec2 create-network-insights-path \
  --source igw-0797cccdc9d73b0e5 \
  --destination i-0495d385ad28331c7 \
  --destination-port 22 \
  --protocol TCP
```

Output:

```
{
  "NetworkInsightsPaths": {
    "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
    "NetworkInsightsPathArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-path/nip-0b26f224f1d131fa8",
    "CreateDate": "2021-01-20T22:43:46.933Z",
    "Source": "igw-0797cccdc9d73b0e5",
    "Destination": "i-0495d385ad28331c7",
    "Protocol": "tcp"
  }
}
```

Untuk informasi selengkapnya, lihat [Memulai menggunakan AWS CLI](#) di Panduan Reachability Analyzer.

- Untuk detail API, lihat [CreateNetworkInsightsPath](#) di Referensi AWS CLI Perintah.

## **create-network-interface-permission**

Contoh kode berikut menunjukkan cara menggunakan `create-network-interface-permission`.

AWS CLI

Untuk membuat izin antarmuka jaringan

Contoh ini memberikan izin ke akun 123456789012 untuk melampirkan antarmuka jaringan `eni-1a2b3c4d` ke sebuah instance.

**Perintah:**

```
aws ec2 create-network-interface-permission --network-interface-id eni-1a2b3c4d --aws-account-id 123456789012 --permission INSTANCE-ATTACH
```

**Output:**

```
{
  "InterfacePermission": {
    "PermissionState": {
      "State": "GRANTED"
    },
    "NetworkInterfacePermissionId": "eni-perm-06fd19020ede149ea",
    "NetworkInterfaceId": "eni-1a2b3c4d",
    "Permission": "INSTANCE-ATTACH",
    "AwsAccountId": "123456789012"
  }
}
```

- Untuk detail API, lihat [CreateNetworkInterfacePermission](#) di Referensi AWS CLI Perintah.

**create-network-interface**

Contoh kode berikut menunjukkan cara menggunakan `create-network-interface`.

**AWS CLI**

Contoh 1: Untuk menentukan alamat IPv4 untuk antarmuka jaringan

`create-network-interface` Contoh berikut membuat antarmuka jaringan untuk subnet tertentu dengan alamat IPv4 primer yang ditentukan.

```
aws ec2 create-network-interface \
  --subnet-id subnet-00a24d0d67acf6333 \
  --description "my network interface" \
  --groups sg-09dfba7ed20cda78b \
  --private-ip-address 10.0.8.17
```

**Output:**

```
{
```

```

"NetworkInterface": {
  "AvailabilityZone": "us-west-2a",
  "Description": "my network interface",
  "Groups": [
    {
      "GroupName": "my-security-group",
      "GroupId": "sg-09dfba7ed20cda78b"
    }
  ],
  "InterfaceType": "interface",
  "Ipv6Addresses": [],
  "MacAddress": "06:6a:0f:9a:49:37",
  "NetworkInterfaceId": "eni-0492b355f0cf3b3f8",
  "OwnerId": "123456789012",
  "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
  "PrivateIpAddress": "10.0.8.17",
  "PrivateIpAddresses": [
    {
      "Primary": true,
      "PrivateDnsName": "ip-10-0-8-17.us-west-2.compute.internal",
      "PrivateIpAddress": "10.0.8.17"
    }
  ],
  "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
  "RequesterManaged": false,
  "SourceDestCheck": true,
  "Status": "pending",
  "SubnetId": "subnet-00a24d0d67acf6333",
  "TagSet": [],
  "VpcId": "vpc-02723a0feeeb9d57b"
}
}

```

Contoh 2: Untuk membuat antarmuka jaringan dengan alamat IPv4 dan alamat IPv6

`create-network-interface` Contoh berikut membuat antarmuka jaringan untuk subnet yang ditentukan dengan alamat IPv4 dan alamat IPv6 yang dipilih oleh Amazon EC2.

```

aws ec2 create-network-interface \
  --subnet-id subnet-00a24d0d67acf6333 \
  --description "my dual stack network interface" \
  --ipv6-address-count 1 \
  --groups sg-09dfba7ed20cda78b

```

## Output:

```
{
  "NetworkInterface": {
    "AvailabilityZone": "us-west-2a",
    "Description": "my dual stack network interface",
    "Groups": [
      {
        "GroupName": "my-security-group",
        "GroupId": "sg-09dfba7ed20cda78b"
      }
    ],
    "InterfaceType": "interface",
    "Ipv6Addresses": [
      {
        "Ipv6Address": "2600:1f13:cfe:3650:a1dc:237c:393a:4ba7",
        "IsPrimaryIpv6": false
      }
    ],
    "MacAddress": "06:b8:68:d2:b2:2d",
    "NetworkInterfaceId": "eni-05da417453f9a84bf",
    "OwnerId": "123456789012",
    "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
    "PrivateIpAddress": "10.0.8.18",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.18"
      }
    ],
    "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
    "RequesterManaged": false,
    "SourceDestCheck": true,
    "Status": "pending",
    "SubnetId": "subnet-00a24d0d67acf6333",
    "TagSet": [],
    "VpcId": "vpc-02723a0feeeb9d57b",
    "Ipv6Address": "2600:1f13:cfe:3650:a1dc:237c:393a:4ba7"
  }
}
```

Contoh 3: Untuk membuat antarmuka jaringan dengan opsi konfigurasi pelacakan koneksi

`create-network-interface` Contoh berikut membuat antarmuka jaringan dan mengkonfigurasi batas waktu pelacakan koneksi idle.

```
aws ec2 create-network-interface \  
  --subnet-id subnet-00a24d0d67acf6333 \  
  --groups sg-02e57dbcfe0331c1b \  
  --connection-tracking-specification TcpEstablishedTimeout=86400,UdpTimeout=60
```

Output:

```
{  
  "NetworkInterface": {  
    "AvailabilityZone": "us-west-2a",  
    "ConnectionTrackingConfiguration": {  
      "TcpEstablishedTimeout": 86400,  
      "UdpTimeout": 60  
    },  
    "Description": "",  
    "Groups": [  
      {  
        "GroupName": "my-security-group",  
        "GroupId": "sg-02e57dbcfe0331c1b"  
      }  
    ],  
    "InterfaceType": "interface",  
    "Ipv6Addresses": [],  
    "MacAddress": "06:4c:53:de:6d:91",  
    "NetworkInterfaceId": "eni-0c133586e08903d0b",  
    "OwnerId": "123456789012",  
    "PrivateDnsName": "ip-10-0-8-94.us-west-2.compute.internal",  
    "PrivateIpAddress": "10.0.8.94",  
    "PrivateIpAddresses": [  
      {  
        "Primary": true,  
        "PrivateDnsName": "ip-10-0-8-94.us-west-2.compute.internal",  
        "PrivateIpAddress": "10.0.8.94"  
      }  
    ],  
    "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",  
    "RequesterManaged": false,  
    "SourceDestCheck": true,  
    "Status": "pending",  
    "SubnetId": "subnet-00a24d0d67acf6333",
```

```

    "TagSet": [],
    "VpcId": "vpc-02723a0feeeb9d57b"
  }
}

```

#### Contoh 4: Untuk membuat Adaptor Kain Elastis

`create-network-interface` Contoh berikut menciptakan EFA.

```

aws ec2 create-network-interface \
  --interface-type efa \
  --subnet-id subnet-00a24d0d67acf6333 \
  --description "my efa" \
  --groups sg-02e57dbcf0331c1b

```

#### Output:

```

{
  "NetworkInterface": {
    "AvailabilityZone": "us-west-2a",
    "Description": "my efa",
    "Groups": [
      {
        "GroupName": "my-efa-sg",
        "GroupId": "sg-02e57dbcf0331c1b"
      }
    ],
    "InterfaceType": "efa",
    "Ipv6Addresses": [],
    "MacAddress": "06:d7:a4:f7:4d:57",
    "NetworkInterfaceId": "eni-034acc2885e862b65",
    "OwnerId": "123456789012",
    "PrivateDnsName": "ip-10-0-8-180.us-west-2.compute.internal",
    "PrivateIpAddress": "10.0.8.180",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateDnsName": "ip-10-0-8-180.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.180"
      }
    ],
    "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
    "RequesterManaged": false,

```



```
    "SourceDestCheck": true,  
    "Status": "pending",  
    "SubnetId": "subnet-00a24d0d67acf6333",  
    "TagSet": [],  
    "VpcId": "vpc-02723a0feeeb9d57b"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Antarmuka jaringan elastis](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CreateNetworkInterface](#) di Referensi AWS CLI Perintah.

## create-placement-group

Contoh kode berikut menunjukkan cara menggunakan `create-placement-group`.

### AWS CLI

Untuk membuat grup penempatan

Perintah contoh ini membuat grup penempatan dengan nama yang ditentukan.

Perintah:

```
aws ec2 create-placement-group --group-name my-cluster --strategy cluster
```

Untuk membuat grup penempatan partisi

Perintah contoh ini membuat grup penempatan partisi bernama `HDFS-Group-A` dengan lima partisi.

Perintah:

```
aws ec2 create-placement-group --group-name HDFS-Group-A --strategy partition --  
partition-count 5
```

- Untuk detail API, lihat [CreatePlacementGroup](#) di Referensi AWS CLI Perintah.

## create-replace-root-volume-task

Contoh kode berikut menunjukkan cara menggunakan `create-replace-root-volume-task`.

## AWS CLI

Contoh 1: Untuk mengembalikan volume root ke status peluncuran awalnya

`create-replace-root-volume-task` Contoh berikut mengembalikan volume root instance `i-0123456789abcdefa` ke status peluncuran awalnya.

```
aws ec2 create-replace-root-volume-task \  
  --instance-id i-0123456789abcdefa
```

Output:

```
{  
  "ReplaceRootVolumeTask":  
  {  
    "InstanceId": "i-0123456789abcdefa",  
    "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",  
    "TaskState": "pending",  
    "StartTime": "2022-03-14T15:06:38Z",  
    "Tags": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengganti volume root](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

Contoh 2: Untuk mengembalikan volume root ke snapshot tertentu

`create-replace-root-volume-task` Contoh berikut mengembalikan volume root instance `i-0123456789abcdefa` ke snapshot `snap-0abcdef1234567890`.

```
aws ec2 create-replace-root-volume-task \  
  --instance-id i-0123456789abcdefa \  
  --snapshot-id snap-0abcdef1234567890
```

Output:

```
{  
  "ReplaceRootVolumeTask":  
  {  
    "InstanceId": "i-0123456789abcdefa",
```

```

    "ReplaceRootVolumeTaskId": "replacevol-0555566667777abcd",
    "TaskState": "pending",
    "StartTime": "2022-03-14T15:16:28Z",
    "Tags": []
  }
}

```

Untuk informasi selengkapnya, lihat [Mengganti volume root](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [CreateReplaceRootVolumeTask](#) di Referensi AWS CLI Perintah.

## create-reserved-instances-listing

Contoh kode berikut menunjukkan cara menggunakan `create-reserved-instances-listing`.

### AWS CLI

Untuk mencantumkan Instans Cadangan di Marketplace Instans Cadangan

`create-reserved-instances-listing` Contoh berikut membuat daftar untuk Instans Cadangan yang ditentukan di Marketplace Instans Cadangan.

```

aws ec2 create-reserved-instances-listing \
  --reserved-instances-id 5ec28771-05ff-4b9b-aa31-9e57dexample \
  --instance-count 3 \
  --price-schedules CurrencyCode=USD,Price=25.50 \
  --client-token 550e8400-e29b-41d4-a716-446655440000

```

- Untuk detail API, lihat [CreateReservedInstancesListing](#) di Referensi AWS CLI Perintah.

## create-restore-image-task

Contoh kode berikut menunjukkan cara menggunakan `create-restore-image-task`.

### AWS CLI

Untuk memulihkan AMI dari bucket S3

`create-restore-image-task` Contoh berikut mengembalikan AMI dari bucket S3. Gunakan nilai untuk `S3objectKey` dan `Bucket` dari `describe-store-image-tasks` output,

tentukan kunci objek AMI dan nama bucket S3 tempat AMI disalin, dan tentukan nama untuk AMI yang dipulihkan. Nama harus unik untuk AMI di Wilayah untuk akun ini. AMI yang dipulihkan akan menerima ID AMI baru.

```
aws ec2 create-restore-image-task \  
  --object-key ami-1234567890abcdef0.bin \  
  --bucket my-ami-bucket \  
  --name "New AMI Name"
```

Output:

```
{  
  "ImageId": "ami-0eab20fe36f83e1a8"  
}
```

Untuk informasi selengkapnya tentang menyimpan dan memulihkan AMI menggunakan S3, lihat Menyimpan dan memulihkan AMI menggunakan S3 < <https://docs.aws.amazon.com/AWS EC2/latest//.html>> di Panduan Pengguna Amazon EC2UserGuide. `ami-store-restore`

- Untuk detail API, lihat [CreateRestoreImageTask](#) di Referensi AWS CLI Perintah.

## **create-route-table**

Contoh kode berikut menunjukkan cara menggunakan `create-route-table`.

### AWS CLI

Untuk membuat tabel rute

Contoh ini membuat tabel rute untuk VPC tertentu.

Perintah:

```
aws ec2 create-route-table --vpc-id vpc-a01106c2
```

Output:

```
{  
  "RouteTable": {  
    "Associations": [],  
    "RouteTableId": "rtb-22574640",
```

```
    "VpcId": "vpc-a01106c2",
    "PropagatingVgws": [],
    "Tags": [],
    "Routes": [
      {
        "GatewayId": "local",
        "DestinationCidrBlock": "10.0.0.0/16",
        "State": "active"
      }
    ]
  }
}
```

- Untuk detail API, lihat [CreateRouteTable](#) di Referensi AWS CLI Perintah.

## create-route

Contoh kode berikut menunjukkan cara menggunakan `create-route`.

### AWS CLI

Untuk membuat rute

Contoh ini membuat rute untuk tabel rute yang ditentukan. Rute cocok dengan semua lalu lintas IPv4 (`0.0.0.0/0`) dan merutekannya ke gateway Internet yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 create-route --route-table-id rtb-22574640 --destination-cidr-block
0.0.0.0/0 --gateway-id igw-c0a643a9
```

Perintah contoh ini membuat rute dalam tabel rute `rtb-g8ff4ea2`. Rute ini cocok dengan lalu lintas untuk blok IPv4 CIDR `10.0.0.0/16` dan merutekannya ke koneksi peering VPC, `pcx-111aaa22`. Rute ini memungkinkan lalu lintas diarahkan ke VPC rekan dalam koneksi peering VPC. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 create-route --route-table-id rtb-g8ff4ea2 --destination-cidr-block
10.0.0.0/16 --vpc-peering-connection-id pcx-1a2b3c4d
```

Contoh ini membuat rute dalam tabel rute yang ditentukan yang cocok dengan semua lalu lintas IPv6 (:::/0) dan merutekannya ke gateway Internet khusus egress yang ditentukan.

Perintah:

```
aws ec2 create-route --route-table-id rtb-dce620b8 --destination-ipv6-cidr-block ::/0 --egress-only-internet-gateway-id eigw-01eadbd45ecd7943f
```

- Untuk detail API, lihat [CreateRoute](#) di Referensi AWS CLI Perintah.

## create-security-group

Contoh kode berikut menunjukkan cara menggunakan `create-security-group`.

### AWS CLI

Guna membuat grup keamanan untuk EC2-Classic

Contoh ini membuat grup keamanan bernama `MySecurityGroup`.

Perintah:

```
aws ec2 create-security-group --group-name MySecurityGroup --description "My security group"
```

Output:

```
{
  "GroupId": "sg-903004f8"
}
```

Guna membuat grup keamanan untuk EC2-VPC

Contoh ini membuat grup keamanan bernama `MySecurityGroup` untuk VPC tertentu.

Perintah:

```
aws ec2 create-security-group --group-name MySecurityGroup --description "My security group" --vpc-id vpc-1a2b3c4d
```

Output:

```
{
  "GroupId": "sg-903004f8"
}
```

Untuk informasi selengkapnya, lihat Menggunakan Grup Keamanan di Panduan Pengguna Antarmuka Baris Perintah AWS .

- Untuk detail API, lihat [CreateSecurityGroup](#) di Referensi AWS CLI Perintah.

## create-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-snapshot`.

### AWS CLI

Untuk membuat snapshot

Perintah contoh ini membuat snapshot volume dengan ID volume `vol-1234567890abcdef0` dan deskripsi singkat untuk mengidentifikasi snapshot.

Perintah:

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description "This is my
root volume snapshot"
```

Output:

```
{
  "Description": "This is my root volume snapshot",
  "Tags": [],
  "Encrypted": false,
  "VolumeId": "vol-1234567890abcdef0",
  "State": "pending",
  "VolumeSize": 8,
  "StartTime": "2018-02-28T21:06:01.000Z",
  "Progress": "",
  "OwnerId": "012345678910",
  "SnapshotId": "snap-066877671789bd71b"
}
```

Untuk membuat snapshot dengan tag

Perintah contoh ini membuat snapshot dan menerapkan dua tag: `purpose=prod` dan `costcenter=123`.

Perintah:

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description 'Prod backup' --tag-specifications 'ResourceType=snapshot,Tags=[{Key=purpose,Value=prod},{Key=costcenter,Value=123}]'
```

Output:

```
{
  "Description": "Prod backup",
  "Tags": [
    {
      "Value": "prod",
      "Key": "purpose"
    },
    {
      "Value": "123",
      "Key": "costcenter"
    }
  ],
  "Encrypted": false,
  "VolumeId": "vol-1234567890abcdef0",
  "State": "pending",
  "VolumeSize": 8,
  "StartTime": "2018-02-28T21:06:06.000Z",
  "Progress": "",
  "OwnerId": "012345678910",
  "SnapshotId": "snap-09ed24a70bc19bbe4"
}
```

- Untuk detail API, lihat [CreateSnapshot](#) di Referensi AWS CLI Perintah.

## create-snapshots

Contoh kode berikut menunjukkan cara menggunakan `create-snapshots`.

### AWS CLI

Contoh 1: Untuk membuat snapshot multi-volume



`create-snapshots` Contoh berikut membuat snapshot dari semua volume yang dilampirkan ke instance yang ditentukan.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0 \  
  --description "This is snapshot of a volume from my-instance"
```

Output:

```
{  
  "Snapshots": [  
    {  
      "Description": "This is a snapshot of a volume from my-instance",  
      "Tags": [],  
      "Encrypted": false,  
      "VolumeId": "vol-0a01d2d5a34697479",  
      "State": "pending",  
      "VolumeSize": 16,  
      "StartTime": "2019-08-05T16:58:19.000Z",  
      "Progress": "",  
      "OwnerId": "123456789012",  
      "SnapshotId": "snap-07f30e3909aa0045e"  
    },  
    {  
      "Description": "This is a snapshot of a volume from my-instance",  
      "Tags": [],  
      "Encrypted": false,  
      "VolumeId": "vol-02d0d4947008cb1a2",  
      "State": "pending",  
      "VolumeSize": 20,  
      "StartTime": "2019-08-05T16:58:19.000Z",  
      "Progress": "",  
      "OwnerId": "123456789012",  
      "SnapshotId": "snap-0ec20b602264aad48"  
    },  
    ...  
  ]  
}
```

Contoh 2: Untuk membuat snapshot multi-volume dengan tag dari volume sumber

`create-snapshots` Contoh berikut membuat snapshot dari semua volume yang dilampirkan ke instance yang ditentukan dan menyalin tag dari setiap volume ke snapshot yang sesuai.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0 \  
  --copy-tags-from-source volume \  
  --description "This is snapshot of a volume from my-instance"
```

Output:

```
{  
  "Snapshots": [  
    {  
      "Description": "This is a snapshot of a volume from my-instance",  
      "Tags": [  
        {  
          "Key": "Name",  
          "Value": "my-volume"  
        }  
      ],  
      "Encrypted": false,  
      "VolumeId": "vol-02d0d4947008cb1a2",  
      "State": "pending",  
      "VolumeSize": 20,  
      "StartTime": "2019-08-05T16:53:04.000Z",  
      "Progress": "",  
      "OwnerId": "123456789012",  
      "SnapshotId": "snap-053bfaeb821a458dd"  
    }  
    ...  
  ]  
}
```

Contoh 3: Untuk membuat snapshot multi-volume yang tidak termasuk volume root

`create-snapshots` Contoh berikut membuat snapshot dari semua volume yang dilampirkan ke instance tertentu kecuali untuk volume root.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0,ExcludeBootVolume=true
```

Lihat contoh 1 untuk output sampel.

#### Contoh 4: Untuk membuat snapshot multi-volume dan menambahkan tag

`create-snapshots` Contoh berikut membuat snapshot dari semua volume yang dilampirkan ke instance yang ditentukan dan menambahkan dua tag ke setiap snapshot.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0 \  
  --tag-specifications 'ResourceType=snapshot,Tags=[{Key=Name,Value=backup},  
{Key=costcenter,Value=123}]'
```

Lihat contoh 1 untuk output sampel.

- Untuk detail API, lihat [CreateSnapshots](#) di Referensi AWS CLI Perintah.

#### **create-spot-datafeed-subscription**

Contoh kode berikut menunjukkan cara menggunakan `create-spot-datafeed-subscription`.

#### AWS CLI

Untuk membuat umpan data Instance Spot

`create-spot-datafeed-subscription` Contoh berikut membuat umpan data Spot Instance.

```
aws ec2 create-spot-datafeed-subscription \  
  --bucket my-bucket \  
  --prefix spot-data-feed
```

Output:

```
{  
  "SpotDatafeedSubscription": {  
    "Bucket": "my-bucket",  
    "OwnerId": "123456789012",  
    "Prefix": "spot-data-feed",  
    "State": "Active"  
  }  
}
```

Umpan data disimpan di bucket Amazon S3 yang Anda tentukan. Nama file untuk umpan data ini memiliki format berikut.

```
my-bucket.s3.amazonaws.com/spot-data-feed/123456789012.YYYY-MM-DD-HH.n.abcd1234.gz
```

Untuk informasi selengkapnya, lihat [umpan data Instans Spot](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [CreateSpotDatafeedSubscription](#) di Referensi AWS CLI Perintah.

## **create-store-image-task**

Contoh kode berikut menunjukkan cara menggunakan `create-store-image-task`.

### AWS CLI

Untuk menyimpan AMI dalam ember S3

`create-store-image-task` Contoh berikut menyimpan AMI dalam bucket S3. Tentukan ID AMI dan nama bucket S3 untuk menyimpan AMI.

```
aws ec2 create-store-image-task \  
  --image-id ami-1234567890abcdef0 \  
  --bucket my-ami-bucket
```

Output:

```
{  
  "objectKey": "ami-1234567890abcdef0.bin"  
}
```

Untuk informasi selengkapnya, lihat [Menyimpan dan memulihkan AMI menggunakan S3](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [CreateStoreImageTask](#) di Referensi AWS CLI Perintah.

## **create-subnet-cidr-reservation**

Contoh kode berikut menunjukkan cara menggunakan `create-subnet-cidr-reservation`.

### AWS CLI

Untuk membuat reservasi CIDR subnet

`create-subnet-cidr-reservation` Contoh berikut membuat reservasi CIDR subnet untuk subnet dan rentang CIDR yang ditentukan.

```
aws ec2 create-subnet-cidr-reservation \  
  --subnet-id subnet-03c51e2eEXAMPLE \  
  --reservation-type prefix \  
  --cidr 10.1.0.20/26
```

Output:

```
{  
  "SubnetCidrReservation": {  
    "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",  
    "SubnetId": "subnet-03c51e2e6cEXAMPLE",  
    "Cidr": "10.1.0.16/28",  
    "ReservationType": "prefix",  
    "OwnerId": "123456789012"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Reservasi CIDR Subnet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [CreateSubnetCidrReservation](#) di Referensi AWS CLI Perintah.

## **create-subnet**

Contoh kode berikut menunjukkan cara menggunakan `create-subnet`.

### AWS CLI

Contoh 1: Untuk membuat subnet dengan blok CIDR IPv4 saja

Contoh `create-subnet` berikut membuat subnet di VPC tertentu dengan blok CIDR IPv4 tertentu.

```
aws ec2 create-subnet \  
  --vpc-id vpc-081ec835f3EXAMPLE \  
  --cidr-block 10.0.0.0/24 \  
  --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv4-only-  
  subnet}]
```

Output:

```
{
  "Subnet": {
    "AvailabilityZone": "us-west-2a",
    "AvailabilityZoneId": "usw2-az2",
    "AvailableIpAddressCount": 251,
    "CidrBlock": "10.0.0.0/24",
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false,
    "State": "available",
    "SubnetId": "subnet-0e99b93155EXAMPLE",
    "VpcId": "vpc-081ec835f3EXAMPLE",
    "OwnerId": "123456789012",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [],
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-ipv4-only-subnet"
      }
    ],
    "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0e99b93155EXAMPLE"
  }
}
```

Contoh 2: Untuk membuat subnet dengan blok CIDR IPv4 dan IPv6

Contoh `create-subnet` berikut membuat subnet di VPC tertentu dengan blok CIDR IPv4 dan IPv6 tertentu.

```
aws ec2 create-subnet \
  --vpc-id vpc-081ec835f3EXAMPLE \
  --cidr-block 10.0.0.0/24 \
  --ipv6-cidr-block 2600:1f16:cfe:3660::/64 \
  --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv4-ipv6-
subnet}]
```

Output:

```
{
  "Subnet": {
    "AvailabilityZone": "us-west-2a",
```

```

    "AvailabilityZoneId": "usw2-az2",
    "AvailableIpAddressCount": 251,
    "CidrBlock": "10.0.0.0/24",
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false,
    "State": "available",
    "SubnetId": "subnet-0736441d38EXAMPLE",
    "VpcId": "vpc-081ec835f3EXAMPLE",
    "OwnerId": "123456789012",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [
      {
        "AssociationId": "subnet-cidr-assoc-06c5f904499fcc623",
        "Ipv6CidrBlock": "2600:1f13:cfe:3660::/64",
        "Ipv6CidrBlockState": {
          "State": "associating"
        }
      }
    ],
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-ipv4-ipv6-subnet"
      }
    ],
    "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0736441d38EXAMPLE"
  }
}

```

Contoh 3: Untuk membuat subnet dengan blok CIDR IPv6 saja

Contoh `create-subnet` berikut membuat subnet di VPC tertentu dengan blok CIDR IPv6 tertentu.

```

aws ec2 create-subnet \
  --vpc-id vpc-081ec835f3EXAMPLE \
  --ipv6-native \
  --ipv6-cidr-block 2600:1f16:115:200::/64 \
  --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv6-only-
subnet}]

```

Output:

```
{
  "Subnet": {
    "AvailabilityZone": "us-west-2a",
    "AvailabilityZoneId": "usw2-az2",
    "AvailableIpAddressCount": 0,
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false,
    "State": "available",
    "SubnetId": "subnet-03f720e7deEXAMPLE",
    "VpcId": "vpc-081ec835f3EXAMPLE",
    "OwnerId": "123456789012",
    "AssignIpv6AddressOnCreation": true,
    "Ipv6CidrBlockAssociationSet": [
      {
        "AssociationId": "subnet-cidr-assoc-01ef639edde556709",
        "Ipv6CidrBlock": "2600:1f13:cfe:3660::/64",
        "Ipv6CidrBlockState": {
          "State": "associating"
        }
      }
    ],
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-ipv6-only-subnet"
      }
    ],
    "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-03f720e7deEXAMPLE"
  }
}
```

Untuk informasi selengkapnya, lihat [VPC dan subnet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [CreateSubnet](#) di Referensi AWS CLI Perintah.

## create-tags

Contoh kode berikut menunjukkan cara menggunakan `create-tags`.

### AWS CLI

Contoh 1: Untuk menambahkan tag ke sumber daya



Contoh `create-tags` berikut menambahkan tanda `Stack=production` ke gambar tertentu, atau menimpa tanda yang sudah ada untuk AMI di mana kunci tandanya adalah `Stack`.

```
aws ec2 create-tags \  
  --resources ami-1234567890abcdef0 \  
  --tags Key=Stack,Value=production
```

Untuk informasi selengkapnya, lihat [Ini adalah judul topik](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 2: Untuk menambahkan tag ke beberapa sumber daya

Contoh `create-tags` berikut menambahkan (atau menimpa) dua tanda untuk AMI dan instans. Salah satu tanda memiliki kunci (`webserver`) tetapi tidak memiliki nilai (nilai diatur ke string kosong). Tanda lainnya memiliki kunci (`stack`) dan nilai (`Production`).

```
aws ec2 create-tags \  
  --resources ami-1a2b3c4d i-1234567890abcdef0 \  
  --tags Key=webserver,Value=   Key=stack,Value=Production
```

Untuk informasi selengkapnya, lihat [Ini adalah judul topik](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 3: Untuk menambahkan tag yang berisi karakter khusus

Contoh `create-tags` berikut menambahkan tanda `[Group]=test` untuk instans. Tanda kurung siku (`[` dan `]`) adalah karakter khusus dan harus di-escape. Contoh berikut juga menggunakan karakter lanjutan baris yang sesuai untuk setiap lingkungan.

Jika Anda menggunakan Windows, kurung elemen yang memiliki karakter khusus dengan petik ganda (`"`), lalu di depan setiap karakter bertanda petik ganda, tambahkan garis miring terbalik (`\`) sebagai berikut:

```
aws ec2 create-tags ^  
  --resources i-1234567890abcdef0 ^  
  --tags Key=\"[Group]\",Value=test
```

Jika Anda menggunakan Windows PowerShell, kelilingi elemen nilai yang memiliki karakter khusus dengan tanda kutip ganda (`"`), mendahului setiap karakter kutipan ganda dengan garis

miring terbalik (\), dan kemudian kelilingi seluruh kunci dan struktur nilai dengan tanda kutip tunggal (') sebagai berikut:

```
aws ec2 create-tags `
  --resources i-1234567890abcdef0 `
  --tags 'Key=\"[Group]\",Value=test'
```

Jika Anda menggunakan Linux atau OS X, apit elemen yang memiliki karakter khusus dengan petik ganda ("), lalu apit keseluruhan kunci dan struktur nilai dengan tanda petik tunggal (') sebagai berikut:

```
aws ec2 create-tags \
  --resources i-1234567890abcdef0 \
  --tags 'Key="[Group]",Value=test'
```

Untuk informasi selengkapnya, lihat [Ini adalah judul topik](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [CreateTags](#) di Referensi AWS CLI Perintah.

## **create-traffic-mirror-filter-rule**

Contoh kode berikut menunjukkan cara menggunakan `create-traffic-mirror-filter-rule`.

### AWS CLI

Untuk membuat aturan filter untuk lalu lintas TCP yang masuk

`create-traffic-mirror-filter-rule` Contoh berikut membuat aturan yang dapat Anda gunakan untuk mencerminkan semua lalu lintas TCP yang masuk. Sebelum Anda menjalankan perintah ini, gunakan `create-traffic-mirror-filter` untuk membuat filter Traffic Mirror.

```
aws ec2 create-traffic-mirror-filter-rule \
  --description "TCP Rule" \
  --destination-cidr-block 0.0.0.0/0 \
  --protocol 6 \
  --rule-action accept \
  --rule-number 1 \
  --source-cidr-block 0.0.0.0/0 \
  --traffic-direction ingress \
  --traffic-mirror-filter-id tmf-04812ff784b25ae67
```

**Output:**

```
{
  "TrafficMirrorFilterRule": {
    "DestinationCidrBlock": "0.0.0.0/0",
    "TrafficMirrorFilterId": "tmf-04812ff784b25ae67",
    "TrafficMirrorFilterRuleId": "tmfr-02d20d996673f3732",
    "SourceCidrBlock": "0.0.0.0/0",
    "TrafficDirection": "ingress",
    "Description": "TCP Rule",
    "RuleNumber": 1,
    "RuleAction": "accept",
    "Protocol": 6
  },
  "ClientToken": "4752b573-40a6-4eac-a8a4-a72058761219"
}
```

Untuk informasi selengkapnya, lihat [Membuat Filter Cermin Lalu Lintas](#) di Panduan Pencerminan AWS Lalu Lintas.

- Untuk detail API, lihat [CreateTrafficMirrorFilterRule](#) di Referensi AWS CLI Perintah.

**create-traffic-mirror-filter**

Contoh kode berikut menunjukkan cara menggunakan `create-traffic-mirror-filter`.

**AWS CLI**

Untuk membuat Filter Cermin Lalu Lintas

`create-traffic-mirror-filter` Contoh berikut membuat filter Traffic Mirror. Setelah Anda membuat filter, gunakan `create-traffic-mirror-filter-rule` untuk menambahkan aturan ke filter.

```
aws ec2 create-traffic-mirror-filter \
  --description "TCP Filter"
```

**Output:**

```
{
  "ClientToken": "28908518-100b-4987-8233-8c744EXAMPLE",
  "TrafficMirrorFilter": {
```

```

    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
    "Description": "TCP Filter",
    "EgressFilterRules": [],
    "IngressFilterRules": [],
    "Tags": [],
    "NetworkServices": []
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat Filter Cermin Lalu Lintas](#) di Panduan Pencermian AWS Lalu Lintas.

- Untuk detail API, lihat [CreateTrafficMirrorFilter](#) di Referensi AWS CLI Perintah.

## create-traffic-mirror-session

Contoh kode berikut menunjukkan cara menggunakan `create-traffic-mirror-session`.

### AWS CLI

Untuk membuat Sesi Cermin Lalu Lintas

`create-traffic-mirror-session` Perintah berikut membuat sesi cermin lalu lintas untuk sumber yang ditentukan dan target untuk 25 byte paket.

```

aws ec2 create-traffic-mirror-session \
  --description "example session" \
  --traffic-mirror-target-id tmt-07f75d8feeEXAMPLE \
  --network-interface-id eni-070203f901EXAMPLE \
  --session-number 1 \
  --packet-length 25 \
  --traffic-mirror-filter-id tmf-04812ff784EXAMPLE

```

Output:

```

{
  "TrafficMirrorSession": {
    "TrafficMirrorSessionId": "tms-08a33b1214EXAMPLE",
    "TrafficMirrorTargetId": "tmt-07f75d8feeEXAMPLE",
    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
    "NetworkInterfaceId": "eni-070203f901EXAMPLE",
    "OwnerId": "111122223333",
  }
}

```

```

    "PacketLength": 25,
    "SessionNumber": 1,
    "VirtualNetworkId": 7159709,
    "Description": "example session",
    "Tags": [],
  },
  "ClientToken": "5236cffc-ee13-4a32-bb5b-388d9da09d96"
}

```

Untuk informasi selengkapnya, lihat [Membuat Sesi Cermin Lalu Lintas](#) di Panduan Pencerminan AWS Lalu Lintas.

- Untuk detail API, lihat [CreateTrafficMirrorSession](#) di Referensi AWS CLI Perintah.

## create-traffic-mirror-target

Contoh kode berikut menunjukkan cara menggunakan `create-traffic-mirror-target`.

### AWS CLI

Untuk membuat target Network Load Balancer Traffic Mirror

`create-traffic-mirror-target` Contoh berikut membuat target Network Load Balancer Traffic Mirror.

```

aws ec2 create-traffic-mirror-target \
  --description "Example Network Load Balancer Target" \
  --network-load-balancer-arn arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/net/NLB/7cdec873EXAMPLE

```

Output:

```

{
  "TrafficMirrorTarget": {
    "Type": "network-load-balancer",
    "Tags": [],
    "Description": "Example Network Load Balancer Target",
    "OwnerId": "111122223333",
    "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:724145273726:loadbalancer/net/NLB/7cdec873EXAMPLE",
    "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE"
  },
}

```

```
"ClientToken": "d5c090f5-8a0f-49c7-8281-72c796a21f72"
}
```

Untuk membuat target Traffic Mirror jaringan

`create-traffic-mirror-target` Contoh berikut membuat antarmuka jaringan target Traffic Mirror.

```
aws ec2 create-traffic-mirror-target --description "Target antarmuka jaringan" -- Eni-
eni-01f6f631EExample network-interface-id
```

Output:

```
{
  "ClientToken": "5289a345-0358-4e62-93d5-47ef3061d65e",
  "TrafficMirrorTarget": {
    "Description": "Network interface target",
    "NetworkInterfaceId": "eni-01f6f631eEXAMPLE",
    "TrafficMirrorTargetId": "tmt-02dcdb2abEXAMPLE",
    "OwnerId": "111122223333",
    "Type": "network-interface",
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Target Cermin Lalu Lintas](#) di Panduan Pencermian AWS Lalu Lintas.

- Untuk detail API, lihat [CreateTrafficMirrorTarget](#) di Referensi AWS CLI Perintah.

## **create-transit-gateway-connect-peer**

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-connect-peer`.

AWS CLI

Untuk membuat rekan Transit Gateway Connect

`create-transit-gateway-connect-peer` Contoh berikut membuat Connect peer.

```
aws ec2 create-transit-gateway-connect-peer \
```

```
--transit-gateway-attachment-id tgw-attach-0f0927767cEXAMPLE \  
--peer-address 172.31.1.11 \  
--inside-cidr-blocks 169.254.6.0/29
```

### Output:

```
{  
  "TransitGatewayConnectPeer": {  
    "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",  
    "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",  
    "State": "pending",  
    "CreationTime": "2021-10-13T03:35:17.000Z",  
    "ConnectPeerConfiguration": {  
      "TransitGatewayAddress": "10.0.0.234",  
      "PeerAddress": "172.31.1.11",  
      "InsideCidrBlocks": [  
        "169.254.6.0/29"  
      ],  
      "Protocol": "gre",  
      "BgpConfigurations": [  
        {  
          "TransitGatewayAsn": 64512,  
          "PeerAsn": 64512,  
          "TransitGatewayAddress": "169.254.6.2",  
          "PeerAddress": "169.254.6.1",  
          "BgpStatus": "down"  
        },  
        {  
          "TransitGatewayAsn": 64512,  
          "PeerAsn": 64512,  
          "TransitGatewayAddress": "169.254.6.3",  
          "PeerAddress": "169.254.6.1",  
          "BgpStatus": "down"  
        }  
      ]  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Lampiran Transit gateway Connect dan rekan Transit Gateway Connect](#) di Panduan Transit Gateway.

- Untuk detail API, lihat [CreateTransitGatewayConnectPeer](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-connect

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-connect`.

### AWS CLI

Untuk membuat lampiran Connect gateway transit

`create-transit-gateway-connect` Contoh berikut membuat lampiran Connect, dengan protokol "gre", untuk lampiran yang ditentukan.

```
aws ec2 create-transit-gateway-connect \
  --transport-transit-gateway-attachment-id tgw-attach-0a89069f57EXAMPLE \
  --options "Protocol=gre"
```

Output:

```
{
  "TransitGatewayConnect": {
    "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",
    "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "State": "pending",
    "CreationTime": "2021-03-09T19:59:17+00:00",
    "Options": {
      "Protocol": "gre"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Lampiran Transit gateway Connect dan rekan Transit Gateway Connect](#) di Panduan Transit Gateway.

- Untuk detail API, lihat [CreateTransitGatewayConnect](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-multicast-domain

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-multicast-domain`.

### AWS CLI

Contoh 1: Untuk membuat domain multicast IGMP



`create-transit-gateway-multicast-domain` Contoh berikut membuat domain multicast untuk gateway transit yang ditentukan. Dengan sumber statis dinonaktifkan, setiap instance dalam subnet yang terkait dengan domain multicast dapat mengirim lalu lintas multicast. Jika setidaknya satu anggota menggunakan protokol IGMP, Anda harus mengaktifkan dukungan IGMPv2.

```
aws ec2 create-transit-gateway-multicast-domain \  
  --transit-gateway-id tgw-0bf0bffefaEXAMPLE \  
  --options StaticSourcesSupport=disable,Igmpv2Support=enable
```

Output:

```
{  
  "TransitGatewayMulticastDomain": {  
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c9e29e2a7EXAMPLE",  
    "TransitGatewayId": "tgw-0bf0bffefaEXAMPLE",  
    "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-  
west-2:123456789012:transit-gateway-multicast-domain/tgw-mcast-  
domain-0c9e29e2a7EXAMPLE",  
    "OwnerId": "123456789012",  
    "Options": {  
      "Igmpv2Support": "enable",  
      "StaticSourcesSupport": "disable",  
      "AutoAcceptSharedAssociations": "disable"  
    },  
    "State": "pending",  
    "CreationTime": "2021-09-29T22:17:13.000Z"  
  }  
}
```

Contoh 2: Untuk membuat domain multicast statis

`create-transit-gateway-multicast-domain` Contoh berikut membuat domain multicast untuk gateway transit yang ditentukan. Dengan sumber statis diaktifkan, Anda harus menambahkan sumber secara statis.

```
aws ec2 create-transit-gateway-multicast-domain \  
  --transit-gateway-id tgw-0bf0bffefaEXAMPLE \  
  --options StaticSourcesSupport=enable,Igmpv2Support=disable
```

Output:

```
{
  "TransitGatewayMulticastDomain": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-000fb24d04EXAMPLE",
    "TransitGatewayId": "tgw-0bf0bfffefaEXAMPLE",
    "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-
west-2:123456789012:transit-gateway-multicast-domain/tgw-mcast-
domain-000fb24d04EXAMPLE",
    "OwnerId": "123456789012",
    "Options": {
      "Icmpv2Support": "disable",
      "StaticSourcesSupport": "enable",
      "AutoAcceptSharedAssociations": "disable"
    },
    "State": "pending",
    "CreationTime": "2021-09-29T22:20:19.000Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Transit Gateways.

- Untuk detail API, lihat [CreateTransitGatewayMulticastDomain](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-peering-attachment

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-peering-attachment`.

### AWS CLI

Untuk membuat lampiran peering gateway transit

`create-transit-gateway-peering-attachment` Contoh berikut membuat permintaan lampiran peering antara dua gateway transit yang ditentukan.

```
aws ec2 create-transit-gateway-peering-attachment \
  --transit-gateway-id tgw-123abc05e04123abc \
  --peer-transit-gateway-id tgw-11223344aabbcc112 \
  --peer-account-id 123456789012 \
  --peer-region us-east-2
```

Output:

```
{
  "TransitGatewayPeeringAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
    "RequesterTgwInfo": {
      "TransitGatewayId": "tgw-123abc05e04123abc",
      "OwnerId": "123456789012",
      "Region": "us-west-2"
    },
    "AcceptorTgwInfo": {
      "TransitGatewayId": "tgw-11223344aabbcc112",
      "OwnerId": "123456789012",
      "Region": "us-east-2"
    },
    "State": "initiatingRequest",
    "CreationTime": "2019-12-09T11:38:05.000Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Lampiran Pengintip Transit Gateway](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [CreateTransitGatewayPeeringAttachment](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-policy-table

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-policy-table`.

### AWS CLI

Untuk membuat tabel kebijakan gateway transit

`create-transit-gateway-policy-table` Contoh berikut membuat tabel kebijakan gateway transit untuk gateway transit yang ditentukan.

```
aws ec2 create-transit-gateway-policy-table \
  --transit-gateway-id tgw-067f8505c18f0bd6e
```

Output:

```
{
```

```

    "TransitGatewayPolicyTable": {
      "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
      "TransitGatewayId": "tgw-067f8505c18f0bd6e",
      "State": "pending",
      "CreationTime": "2023-11-28T16:36:43+00:00"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Tabel kebijakan gateway transit](#) di Panduan Pengguna Gateway Transit.

- Untuk detail API, lihat [CreateTransitGatewayPolicyTable](#) di Referensi AWS CLI Perintah.

### create-transit-gateway-prefix-list-reference

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-prefix-list-reference`.

#### AWS CLI

Untuk membuat referensi ke daftar awalan

`create-transit-gateway-prefix-list-reference` Contoh berikut membuat referensi ke daftar awalan yang ditentukan dalam tabel rute gateway transit yang ditentukan.

```

aws ec2 create-transit-gateway-prefix-list-reference \
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
  --prefix-list-id pl-1111112222222333 \
  --transit-gateway-attachment-id tgw-attach-aaaaaabbbbb11111

```

Output:

```

{
  "TransitGatewayPrefixListReference": {
    "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
    "PrefixListId": "pl-1111112222222333",
    "PrefixListOwnerId": "123456789012",
    "State": "pending",
    "Blackhole": false,
    "TransitGatewayAttachment": {
      "TransitGatewayAttachmentId": "tgw-attach-aaaaaabbbbb11111",
      "ResourceType": "vpc",

```

```

    "ResourceId": "vpc-112233445566aabbcc"
  }
}
}

```

Untuk informasi selengkapnya, lihat [Referensi daftar awalan](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [CreateTransitGatewayPrefixListReferenced](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-route-table

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-route-table`.

### AWS CLI

Untuk membuat Tabel Rute Transit Gateway

`create-transit-gateway-route-table` Contoh berikut membuat tabel rute untuk gateway transit yang ditentukan.

```

aws ec2 create-transit-gateway-route-table \
  --transit-gateway-id tgw-0262a0e521EXAMPLE

```

Output:

```

{
  "TransitGatewayRouteTable": {
    "TransitGatewayRouteTableId": "tgw-rtb-0960981be7EXAMPLE",
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "State": "pending",
    "DefaultAssociationRouteTable": false,
    "DefaultPropagationRouteTable": false,
    "CreationTime": "2019-07-10T19:01:46.000Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat tabel rute gateway transit](#) di Panduan Gateway Transit.

- Untuk detail API, lihat [CreateTransitGatewayRouteTable](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-route

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-route`.

### AWS CLI

Untuk membuat rute gateway transit

`create-transit-gateway-route` Contoh berikut membuat rute, dengan tujuan yang ditentukan, untuk tabel rute yang ditentukan.

```
aws ec2 create-transit-gateway-route \
  --destination-cidr-block 10.0.2.0/24 \
  --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE \
  --transit-gateway-attachment-id tgw-attach-0b5968d3b6EXAMPLE
```

Output:

```
{
  "Route": {
    "DestinationCidrBlock": "10.0.2.0/24",
    "TransitGatewayAttachments": [
      {
        "ResourceId": "vpc-0065acced4EXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
        "ResourceType": "vpc"
      }
    ],
    "Type": "static",
    "State": "active"
  }
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [CreateTransitGatewayRoute](#) di Referensi AWS CLI Perintah.

## create-transit-gateway-vpc-attachment

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway-vpc-attachment`.

## AWS CLI

Contoh 1: Untuk mengaitkan gateway transit dengan VPC

`create-transit-gateway-vpc-attachment` Contoh berikut membuat lampiran gateway transit ke VPC yang ditentukan.

```
aws ec2 create-transit-gateway-vpc-attachment \  
  --transit-gateway-id tgw-0262a0e521EXAMPLE \  
  --vpc-id vpc-07e8ffd50f49335df \  
  --subnet-id subnet-0752213d59EXAMPLE
```

Output:

```
{  
  "TransitGatewayVpcAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",  
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",  
    "VpcId": "vpc-07e8ffd50fEXAMPLE",  
    "VpcOwnerId": "111122223333",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-0752213d59EXAMPLE"  
    ],  
    "CreationTime": "2019-07-10T17:33:46.000Z",  
    "Options": {  
      "DnsSupport": "enable",  
      "Ipv6Support": "disable"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat lampiran gateway transit ke VPC](#) di Panduan Gateway Transit.

Contoh 2: Untuk mengaitkan gateway transit dengan beberapa subnet dalam VPC

`create-transit-gateway-vpc-attachment` Contoh berikut membuat lampiran gateway transit ke VPC dan subnet yang ditentukan.

```
aws ec2 create-transit-gateway-vpc-attachment \  
  --transit-gateway-id tgw-0262a0e521EXAMPLE \  
  --vpc-id vpc-07e8ffd50f49335df \  
  --subnet-ids subnet-0752213d59EXAMPLE subnet-0752213d59EXAMPLE
```

```
--transit-gateway-id tgw-02f776b1a7EXAMPLE \  
--vpc-id vpc-3EXAMPLE \  
--subnet-ids "subnet-dEXAMPLE" "subnet-6EXAMPLE"
```

### Output:

```
{  
  "TransitGatewayVpcAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-0e141e0bebEXAMPLE",  
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",  
    "VpcId": "vpc-3EXAMPLE",  
    "VpcOwnerId": "111122223333",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-6EXAMPLE",  
      "subnet-dEXAMPLE"  
    ],  
    "CreationTime": "2019-12-17T20:07:52.000Z",  
    "Options": {  
      "DnsSupport": "enable",  
      "Ipv6Support": "disable"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat lampiran gateway transit ke VPC](#) di Panduan Gateway Transit.

- Untuk detail API, lihat [CreateTransitGatewayVpcAttachment](#) di Referensi AWS CLI Perintah.

## create-transit-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-transit-gateway`.

### AWS CLI

Untuk membuat gateway transit

`create-transit-gateway` Contoh berikut membuat gateway transit.

```
aws ec2 create-transit-gateway \  
--description MyTGW \  

```



```
--options
```

```
AmazonSideAsn=64516,AutoAcceptSharedAttachments=enable,DefaultRouteTableAssociation=enable,
```

Output:

```
{
  "TransitGateway": {
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-gateway/
tgw-0262a0e521EXAMPLE",
    "State": "pending",
    "OwnerId": "111122223333",
    "Description": "MyTGW",
    "CreationTime": "2019-07-10T14:02:12.000Z",
    "Options": {
      "AmazonSideAsn": 64516,
      "AutoAcceptSharedAttachments": "enable",
      "DefaultRouteTableAssociation": "enable",
      "AssociationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
      "DefaultRouteTablePropagation": "enable",
      "PropagationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
      "VpnEcmpSupport": "enable",
      "DnsSupport": "enable"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [CreateTransitGateway](#) di Referensi AWS CLI Perintah.

## create-verified-access-endpoint

Contoh kode berikut menunjukkan cara menggunakan `create-verified-access-endpoint`.

### AWS CLI

Untuk membuat titik akhir Akses Terverifikasi

`create-verified-access-endpoint` Contoh berikut membuat titik akhir Akses Terverifikasi untuk grup Akses Terverifikasi tertentu. Antarmuka jaringan dan grup keamanan yang ditentukan harus termasuk dalam VPC yang sama.

```
aws ec2 create-verified-access-endpoint \  
  --verified-access-group-id vgr-0dbe967baf14b7235 \  
  --endpoint-type network-interface \  
  --attachment-type vpc \  
  --domain-certificate-arn arn:aws:acm:us-east-2:123456789012:certificate/  
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE \  
  --application-domain example.com \  
  --endpoint-domain-prefix my-ava-app \  
  --security-group-ids sg-004915970c4c8f13a \  
  --network-interface-options  
NetworkInterfaceId=eni-0aec70418c8d87a0f,Protocol=https,Port=443 \  
  --tag-specifications ResourceType=verified-access-  
endpoint,Tags=[{Key=Name,Value=my-va-endpoint}]
```

### Output:

```
{  
  "VerifiedAccessEndpoint": {  
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",  
    "VerifiedAccessGroupId": "vgr-0dbe967baf14b7235",  
    "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",  
    "ApplicationDomain": "example.com",  
    "EndpointType": "network-interface",  
    "AttachmentType": "vpc",  
    "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/  
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",  
    "EndpointDomain": "my-ava-  
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-  
east-2.amazonaws.com",  
    "SecurityGroupIds": [  
      "sg-004915970c4c8f13a"  
    ],  
    "NetworkInterfaceOptions": {  
      "NetworkInterfaceId": "eni-0aec70418c8d87a0f",  
      "Protocol": "https",  
      "Port": 443  
    },  
    "Status": {  
      "Code": "pending"  
    },  
    "Description": "",  
    "CreationTime": "2023-08-25T20:54:43",  
    "LastUpdatedTime": "2023-08-25T20:54:43",
```

```

    "Tags": [
      {
        "Key": "Name",
        "Value": "my-va-endpoint"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Titik akhir Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [CreateVerifiedAccessEndpoint](#) di Referensi AWS CLI Perintah.

## create-verified-access-group

Contoh kode berikut menunjukkan cara menggunakan `create-verified-access-group`.

### AWS CLI

Untuk membuat grup Akses Terverifikasi

`create-verified-access-group` Contoh berikut membuat grup Akses Terverifikasi untuk instance Akses Terverifikasi yang ditentukan.

```

aws ec2 create-verified-access-group \
  --verified-access-instance-id vai-0ce000c0b7643abea \
  --tag-specifications ResourceType=verified-access-
group,Tags=[{Key=Name,Value=my-va-group}]

```

Output:

```

{
  "VerifiedAccessGroup": {
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "",
    "Owner": "123456789012",
    "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
    "CreationTime": "2023-08-25T19:55:19",
    "LastUpdatedTime": "2023-08-25T19:55:19",
  }
}

```

```
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-va-group"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Grup Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [CreateVerifiedAccessGroup](#) di Referensi AWS CLI Perintah.

## create-verified-access-instance

Contoh kode berikut menunjukkan cara menggunakan `create-verified-access-instance`.

### AWS CLI

Untuk membuat instance Akses Terverifikasi

`create-verified-access-instance` Contoh berikut membuat instance Akses Terverifikasi dengan tag Nama.

```
aws ec2 create-verified-access-instance \
  --tag-specifications ResourceType=verified-access-
instance,Tags=[{Key=Name,Value=my-va-instance}]
```

Output:

```
{
  "VerifiedAccessInstance": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "",
    "VerifiedAccessTrustProviders": [],
    "CreationTime": "2023-08-25T18:27:56",
    "LastUpdatedTime": "2023-08-25T18:27:56",
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-va-instance"
      }
    ]
  }
}
```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Instans Akses Terverifikasi](#) di Panduan Pengguna Akses AWS Terverifikasi.

- Untuk detail API, lihat [CreateVerifiedAccessInstance](#) di Referensi AWS CLI Perintah.

## create-verified-access-trust-provider

Contoh kode berikut menunjukkan cara menggunakan `create-verified-access-trust-provider`.

### AWS CLI

Untuk membuat penyedia kepercayaan Akses Terverifikasi

`create-verified-access-trust-provider` Contoh berikut menyiapkan penyedia kepercayaan Akses Terverifikasi menggunakan Pusat AWS Identitas.

```

aws ec2 create-verified-access-trust-provider \
  --trust-provider-type user \
  --user-trust-provider-type iam-identity-center \
  --policy-reference-name idc \
  --tag-specifications ResourceType=verified-access-trust-
provider,Tags=[{Key=Name,Value=my-va-trust-provider}]

```

Output:

```

{
  "VerifiedAccessTrustProvider": {
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
    "Description": "",
    "TrustProviderType": "user",
    "UserTrustProviderType": "iam-identity-center",
    "PolicyReferenceName": "idc",
    "CreationTime": "2023-08-25T18:40:36",
    "LastUpdatedTime": "2023-08-25T18:40:36",
    "Tags": [
      {

```

```
        "Key": "Name",
        "Value": "my-va-trust-provider"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Penyedia kepercayaan untuk Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [CreateVerifiedAccessTrustProvider](#) di Referensi AWS CLI Perintah.

## create-volume

Contoh kode berikut menunjukkan cara menggunakan `create-volume`.

### AWS CLI

Untuk membuat volume General Purpose SSD (gp2) kosong

`create-volume` Contoh berikut membuat volume 80 GiB General Purpose SSD (gp2) di Availability Zone yang ditentukan. Perhatikan bahwa Wilayah saat ini harus `us-east-1`, atau Anda dapat menambahkan `--region` parameter untuk menentukan Wilayah untuk perintah.

```
aws ec2 create-volume \
  --volume-type gp2 \
  --size 80 \
  --availability-zone us-east-1a
```

Output:

```
{
  "AvailabilityZone": "us-east-1a",
  "Tags": [],
  "Encrypted": false,
  "VolumeType": "gp2",
  "VolumeId": "vol-1234567890abcdef0",
  "State": "creating",
  "Iops": 240,
  "SnapshotId": "",
  "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
  "Size": 80
}
```

```
}
```

Jika Anda tidak menentukan jenis volume, tipe volume default adalah gp2.

```
aws ec2 create-volume \  
  --size 80 \  
  --availability-zone us-east-1a
```

Contoh 2: Untuk membuat volume IOPS SSD (io1) yang Disediakan dari snapshot

Contoh berikut membuat volume Provisioned IOPS SSD (io1) dengan 1000 IOPS yang disediakan di Availability Zone yang ditentukan menggunakan snapshot yang ditentukan.

```
aws ec2 create-volume \  
  --volume-type io1 \  
  --iops 1000 \  
  --snapshot-id snap-066877671789bd71b \  
  --availability-zone us-east-1a
```

Output:

```
{  
  "AvailabilityZone": "us-east-1a",  
  "Tags": [],  
  "Encrypted": false,  
  "VolumeType": "io1",  
  "VolumeId": "vol-1234567890abcdef0",  
  "State": "creating",  
  "Iops": 1000,  
  "SnapshotId": "snap-066877671789bd71b",  
  "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",  
  "Size": 500  
}
```

Contoh 3: Untuk membuat volume terenkripsi

Contoh berikut membuat volume terenkripsi menggunakan CMK default untuk enkripsi EBS. Jika enkripsi secara default dinonaktifkan, Anda harus menentukan `--encrypted` parameter sebagai berikut.

```
aws ec2 create-volume \  
  --size 80 \  
  --encrypted \  
  --availability-zone us-east-1a
```

Output:

```
{  
  "AvailabilityZone": "us-east-1a",  
  "Tags": [],  
  "Encrypted": true,  
  "VolumeType": "gp2",  
  "VolumeId": "vol-1234567890abcdef0",  
  "State": "creating",  
  "Iops": 240,  
  "SnapshotId": "",  
  "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",  
  "Size": 80  
}
```

Jika enkripsi secara default diaktifkan, perintah contoh berikut membuat volume terenkripsi, bahkan tanpa parameter. `--encrypted`

```
aws ec2 create-volume \  
  --size 80 \  
  --availability-zone us-east-1a
```

Jika Anda menggunakan `--kms-key-id` parameter untuk menentukan CMK yang dikelola pelanggan, Anda harus menentukan `--encrypted` parameter bahkan jika enkripsi secara default diaktifkan.

```
aws ec2 create-volume \  
  --volume-type gp2 \  
  --size 80 \  
  --encrypted \  
  --kms-key-id 0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE \  
  --availability-zone us-east-1a
```

Contoh 4: Untuk membuat volume dengan tag

`create-volume` Contoh berikut menciptakan volume dan menambahkan dua tag.



```
aws ec2 create-volume \  
  --availability-zone us-east-1a \  
  --volume-type gp2 \  
  --size 80 \  
  --tag-specifications 'ResourceType=volume,Tags=[{Key=purpose,Value=production},  
{Key=cost-center,Value=cc123}]'
```

- Untuk detail API, lihat [CreateVolume](#) di Referensi AWS CLI Perintah.

## create-vpc-endpoint-connection-notification

Contoh kode berikut menunjukkan cara menggunakan `create-vpc-endpoint-connection-notification`.

### AWS CLI

Untuk membuat notifikasi koneksi endpoint

Contoh ini membuat pemberitahuan untuk layanan titik akhir tertentu yang memberi tahu Anda ketika titik akhir antarmuka telah terhubung ke layanan Anda dan ketika titik akhir telah diterima untuk layanan Anda.

Perintah:

```
aws ec2 create-vpc-endpoint-connection-notification --connection-notification-arn  
arn:aws:sns:us-east-2:123456789012:VpceNotification --connection-events Connect  
Accept --service-id vpce-svc-1237881c0d25a3abc
```

Output:

```
{  
  "ConnectionNotification": {  
    "ConnectionNotificationState": "Enabled",  
    "ConnectionNotificationType": "Topic",  
    "ServiceId": "vpce-svc-1237881c0d25a3abc",  
    "ConnectionEvents": [  
      "Accept",  
      "Connect"  
    ],  
    "ConnectionNotificationId": "vpce-nfn-008776de7e03f5abc",
```

```

    "ConnectionNotificationArn": "arn:aws:sns:us-
east-2:123456789012:VpceNotification"
  }
}

```

- Untuk detail API, lihat [CreateVpcEndpointConnectionNotification](#) di Referensi AWS CLI Perintah.

## create-vpc-endpoint-service-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-vpc-endpoint-service-configuration`.

### AWS CLI

Contoh 1: Untuk membuat konfigurasi layanan endpoint untuk titik akhir antarmuka

`create-vpc-endpoint-service-configuration` Contoh berikut membuat konfigurasi layanan titik akhir VPC menggunakan Network Load Balancer. `nlb-vpce` Contoh ini juga menentukan bahwa permintaan untuk terhubung ke layanan melalui titik akhir antarmuka harus diterima.

```

aws ec2 create-vpc-endpoint-service-configuration \
  --network-load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/net/nlb-vpce/e94221227f1ba532 \
  --acceptance-required

```

### Output:

```

{
  "ServiceConfiguration": {
    "ServiceType": [
      {
        "ServiceType": "Interface"
      }
    ],
    "NetworkLoadBalancerArns": [
      "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/net/
nlb-vpce/e94221227f1ba532"
    ],
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-03d5ebb7d9579a2b3",
    "ServiceState": "Available",
  }
}

```

```

    "ServiceId": "vpce-svc-03d5ebb7d9579a2b3",
    "AcceptanceRequired": true,
    "AvailabilityZones": [
        "us-east-1d"
    ],
    "BaseEndpointDnsNames": [
        "vpce-svc-03d5ebb7d9579a2b3.us-east-1.vpce.amazonaws.com"
    ]
}
}

```

Contoh 2: Untuk membuat konfigurasi layanan endpoint untuk titik akhir Load Balancer Gateway

`create-vpc-endpoint-service-configuration` Contoh berikut membuat konfigurasi layanan titik akhir VPC menggunakan Load Balancer Gateway. GWLBService Permintaan untuk terhubung ke layanan melalui titik akhir Load Balancer Gateway diterima secara otomatis.

```

aws ec2 create-vpc-endpoint-service-configuration \
  --gateway-load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/gwy/GWLBService/123123123123abcc \
  --no-acceptance-required

```

Output:

```

{
  "ServiceConfiguration": {
    "ServiceType": [
      {
        "ServiceType": "GatewayLoadBalancer"
      }
    ],
    "ServiceId": "vpce-svc-123123a1c43abc123",
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123",
    "ServiceState": "Available",
    "AvailabilityZones": [
        "us-east-1d"
    ],
    "AcceptanceRequired": false,
    "ManagesVpcEndpoints": false,
    "GatewayLoadBalancerArns": [
        "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/gwy/
GWLBService/123123123123abcc"
    ]
  }
}

```

```

    ]
  }
}

```

Untuk informasi selengkapnya, lihat [layanan titik akhir VPC di Panduan Pengguna Amazon VPC](#).

- Untuk detail API, lihat [CreateVpcEndpointServiceConfiguration](#) di Referensi AWS CLI Perintah.

## create-vpc-endpoint

Contoh kode berikut menunjukkan cara menggunakan `create-vpc-endpoint`.

### AWS CLI

Contoh 1: Untuk membuat titik akhir gateway

`create-vpc-endpoint` Contoh berikut membuat titik akhir VPC gateway antara VPC dan `vpc-1a2b3c4d` Amazon S3 di wilayah tersebut, dan mengaitkan tabel rute `us-east-1` dengan titik akhir. `rtb-11aa22bb`

```

aws ec2 create-vpc-endpoint \
  --vpc-id vpc-1a2b3c4d \
  --service-name com.amazonaws.us-east-1.s3 \
  --route-table-ids rtb-11aa22bb

```

Output:

```

{
  "VpcEndpoint": {
    "PolicyDocument": "{\"Version\":\"2008-10-17\",\"Statement\": [{\"Sid\":\"\", \"Effect\":\"Allow\", \"Principal\": \"*\", \"Action\": \"*\", \"Resource\": \"*\"}]}",
    "VpcId": "vpc-1a2b3c4d",
    "State": "available",
    "ServiceName": "com.amazonaws.us-east-1.s3",
    "RouteTableIds": [
      "rtb-11aa22bb"
    ],
    "VpcEndpointId": "vpc-1a2b3c4d",
    "CreationTimestamp": "2015-05-15T09:40:50Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat titik akhir gateway](#) di AWS PrivateLink Panduan.

Contoh 2: Untuk membuat titik akhir antarmuka

create-vpc-endpoint Contoh berikut membuat antarmuka VPC endpoint antara VPC dan vpc-1a2b3c4d Amazon S3 di wilayah tersebut. us-east-1 Perintah membuat titik akhir di subnet subnet-1a2b3c4d, mengaitkannya dengan grup keamanan sg-1a2b3c4d, dan menambahkan tag dengan kunci "Layanan" dan Nilai "S3".

```
aws ec2 create-vpc-endpoint \  
  --vpc-id vpc-1a2b3c4d \  
  --vpc-endpoint-type Interface \  
  --service-name com.amazonaws.us-east-1.s3 \  
  --subnet-ids subnet-7b16de0c \  
  --security-group-id sg-1a2b3c4d \  
  --tag-specifications ResourceType=vpc-endpoint,Tags=[{Key=service,Value=S3}]
```

Output:

```
{  
  "VpcEndpoint": {  
    "VpcEndpointId": "vpce-1a2b3c4d5e6f1a2b3",  
    "VpcEndpointType": "Interface",  
    "VpcId": "vpc-1a2b3c4d",  
    "ServiceName": "com.amazonaws.us-east-1.s3",  
    "State": "pending",  
    "RouteTableIds": [],  
    "SubnetIds": [  
      "subnet-1a2b3c4d"  
    ],  
    "Groups": [  
      {  
        "GroupId": "sg-1a2b3c4d",  
        "GroupName": "default"  
      }  
    ],  
    "PrivateDnsEnabled": false,  
    "RequesterManaged": false,  
    "NetworkInterfaceIds": [  
      "eni-0b16f0581c8ac6877"  
    ],  
    "DnsEntries": [  
      {
```

```

        "DnsName": "*.vpce-1a2b3c4d5e6f1a2b3-9hnenorg.s3.us-
east-1.vpce.amazonaws.com",
        "HostedZoneId": "Z7HUB22UULQXV"
    },
    {
        "DnsName": "*.vpce-1a2b3c4d5e6f1a2b3-9hnenorg-us-east-1c.s3.us-
east-1.vpce.amazonaws.com",
        "HostedZoneId": "Z7HUB22UULQXV"
    }
],
"CreationTimestamp": "2021-03-05T14:46:16.030000+00:00",
"Tags": [
    {
        "Key": "service",
        "Value": "S3"
    }
],
"OwnerId": "123456789012"
}
}

```

Untuk informasi selengkapnya, lihat [Membuat titik akhir antarmuka](#) di Panduan Pengguna untuk AWS PrivateLink.

### Contoh 3: Untuk membuat titik akhir Load Balancer Gateway

`create-vpc-endpoint` Contoh berikut membuat titik akhir Load Balancer Gateway antara VPC `vpc-111122223333aabbcc` dan layanan yang dikonfigurasi menggunakan Load Balancer Gateway.

```

aws ec2 create-vpc-endpoint \
  --service-name com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123 \
  --vpc-endpoint-type GatewayLoadBalancer \
  --vpc-id vpc-111122223333aabbcc \
  --subnet-ids subnet-0011aabbcc2233445

```

### Output:

```

{
  "VpcEndpoint": {
    "VpcEndpointId": "vpce-aabbaabbaabbaabba",
    "VpcEndpointType": "GatewayLoadBalancer",

```

```
"VpcId": "vpc-111122223333aabb",
"ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123",
"State": "pending",
"SubnetIds": [
  "subnet-0011aabbcc2233445"
],
"RequesterManaged": false,
"NetworkInterfaceIds": [
  "eni-01010120203030405"
],
"CreationTimestamp": "2020-11-11T08:06:03.522Z",
"OwnerId": "123456789012"
}
}
```

Untuk informasi selengkapnya, lihat [titik akhir Load Balancer Gateway](#) di Panduan Pengguna untuk AWS PrivateLink

- Untuk detail API, lihat [CreateVpcEndpoint](#) di Referensi AWS CLI Perintah.

## create-vpc-peering-connection

Contoh kode berikut menunjukkan cara menggunakan `create-vpc-peering-connection`.

### AWS CLI

Untuk membuat koneksi peering VPC antara VPC Anda

Contoh ini meminta koneksi peering antara VPC Anda `vpc-1a2b3c4d` dan `vpc-11122233`.

Perintah:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id
vpc-11122233
```

Output:

```
{
  "VpcPeeringConnection": {
    "Status": {
      "Message": "Initiating Request to 444455556666",
      "Code": "initiating-request"
    }
  }
}
```

```
    },
    "Tags": [],
    "RequesterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-1a2b3c4d",
      "CidrBlock": "10.0.0.0/28"
    },
    "VpcPeeringConnectionId": "pcx-111aaa111",
    "ExpirationTime": "2014-04-02T16:13:36.000Z",
    "AccepterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-11122233"
    }
  }
}
```

Untuk membuat koneksi peering VPC dengan VPC di akun lain

Contoh ini meminta koneksi peering antara VPC Anda (vpc-1a2b3c4d), dan VPC (vpc-11122233) yang dimiliki akun 123456789012. AWS

Perintah:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id
vpc-11122233 --peer-owner-id 123456789012
```

Untuk membuat koneksi peering VPC dengan VPC di wilayah yang berbeda

Contoh ini meminta koneksi peering antara VPC Anda di wilayah saat ini (vpc-1a2b3c4d), dan VPC (vpc-11122233) di akun Anda di wilayah tersebut. us-west-2

Perintah:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id
vpc-11122233 --peer-region us-west-2
```

Contoh ini meminta koneksi peering antara VPC Anda di wilayah saat ini (vpc-1a2b3c4d), dan VPC (vpc-11122233) yang memiliki akun 123456789012 yang ada di wilayah tersebut. AWS us-west-2

Perintah:



```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id
vpc-11122233 --peer-owner-id 123456789012 --peer-region us-west-2
```

- Untuk detail API, lihat [CreateVpcPeeringConnection](#) di Referensi AWS CLI Perintah.

## create-vpc

Contoh kode berikut menunjukkan cara menggunakan `create-vpc`.

### AWS CLI

Contoh 1: Untuk membuat VPC

Contoh `create-vpc` berikut membuat VPC dengan blok CIDR IPv4 dan tanda Nama tertentu.

```
aws ec2 create-vpc \
  --cidr-block 10.0.0.0/16 \
  --tag-specification ResourceType=vpc,Tags=[{Key=Name,Value=MyVpc}]
```

Output:

```
{
  "Vpc": {
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-5EXAMPLE",
    "State": "pending",
    "VpcId": "vpc-0a60eb65b4EXAMPLE",
    "OwnerId": "123456789012",
    "InstanceTenancy": "default",
    "Ipv6CidrBlockAssociationSet": [],
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-07501b79ecEXAMPLE",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ],
    "IsDefault": false,
    "Tags": [
      {
```

```

        "Key": "Name",
        "Value": MyVpc"
    }
  ]
}

```

Contoh 2: Untuk membuat VPC dengan penghunian khusus

Contoh `create-vpc` berikut membuat VPC dengan blok CIDR IPv4 dan penghunian khusus tertentu.

```

aws ec2 create-vpc \
  --cidr-block 10.0.0.0/16 \
  --instance-tenancy dedicated

```

Output:

```

{
  "Vpc": {
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-19edf471",
    "State": "pending",
    "VpcId": "vpc-0a53287fa4EXAMPLE",
    "OwnerId": "111122223333",
    "InstanceTenancy": "dedicated",
    "Ipv6CidrBlockAssociationSet": [],
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-00b24cc1c2EXAMPLE",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ],
    "IsDefault": false
  }
}

```

Contoh 3: Untuk membuat VPC dengan blok CIDR IPv6

Contoh `create-vpc` berikut membuat VPC dengan blok CIDR IPv6 yang disediakan Amazon.

```
aws ec2 create-vpc \  
  --cidr-block 10.0.0.0/16 \  
  --amazon-provided-ipv6-cidr-block
```

### Output:

```
{  
  "Vpc": {  
    "CidrBlock": "10.0.0.0/16",  
    "DhcpOptionsId": "dopt-dEXAMPLE",  
    "State": "pending",  
    "VpcId": "vpc-0fc5e3406bEXAMPLE",  
    "OwnerId": "123456789012",  
    "InstanceTenancy": "default",  
    "Ipv6CidrBlockAssociationSet": [  
      {  
        "AssociationId": "vpc-cidr-assoc-068432c60bEXAMPLE",  
        "Ipv6CidrBlock": "",  
        "Ipv6CidrBlockState": {  
          "State": "associating"  
        },  
        "Ipv6Pool": "Amazon",  
        "NetworkBorderGroup": "us-west-2"  
      }  
    ],  
    "CidrBlockAssociationSet": [  
      {  
        "AssociationId": "vpc-cidr-assoc-0669f8f9f5EXAMPLE",  
        "CidrBlock": "10.0.0.0/16",  
        "CidrBlockState": {  
          "State": "associated"  
        }  
      }  
    ],  
    "IsDefault": false  
  }  
}
```

Contoh 4: Untuk membuat VPC dengan CIDR dari kolam IPAM

Contoh `create-vpc` berikut membuat VPC dengan CIDR dari kolam Manajer Alamat IP (IPAM) Amazon VPC.

## Linux dan macOS:

```
aws ec2 create-vpc \  
  --ipv4-ipam-pool-id ipam-pool-0533048da7d823723 \  
  --tag-specifications ResourceType=vpc,Tags='[{"Key=Environment,Value="Preprod"},  
{"Key=Owner,Value="Build Team"}]'
```

## Windows:

```
aws ec2 create-vpc ^  
  --ipv4-ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --tag-specifications ResourceType=vpc,Tags=[{"Key=Environment,Value="Preprod"},  
{"Key=Owner,Value="Build Team"}]
```

## Output:

```
{  
  "Vpc": {  
    "CidrBlock": "10.0.1.0/24",  
    "DhcpOptionsId": "dopt-2afccf50",  
    "State": "pending",  
    "VpcId": "vpc-010e1791024eb0af9",  
    "OwnerId": "123456789012",  
    "InstanceTenancy": "default",  
    "Ipv6CidrBlockAssociationSet": [],  
    "CidrBlockAssociationSet": [  
      {  
        "AssociationId": "vpc-cidr-assoc-0a77de1d803226d4b",  
        "CidrBlock": "10.0.1.0/24",  
        "CidrBlockState": {  
          "State": "associated"  
        }  
      }  
    ],  
    "IsDefault": false,  
    "Tags": [  
      {  
        "Key": "Environment",  
        "Value": "Preprod"  
      },  
      {  
        "Key": "Owner",
```

```

    "Value": "Build Team"
  }
]
}
}

```

Untuk informasi selengkapnya, lihat [Membuat VPC yang menggunakan CIDR kolam IPAM](#) di Panduan Pengguna IPAM Amazon VPC.

- Untuk detail API, lihat [CreateVpc](#) di Referensi AWS CLI Perintah.

## create-vpn-connection-route

Contoh kode berikut menunjukkan cara menggunakan `create-vpn-connection-route`.

### AWS CLI

Untuk membuat rute statis untuk koneksi VPN

Contoh ini membuat rute statis untuk koneksi VPN yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 create-vpn-connection-route --vpn-connection-id vpn-40f41529 --destination-cidr-block 11.12.0.0/16
```

- Untuk detail API, lihat [CreateVpnConnectionRoute](#) di Referensi AWS CLI Perintah.

## create-vpn-connection

Contoh kode berikut menunjukkan cara menggunakan `create-vpn-connection`.

### AWS CLI

Contoh 1: Untuk membuat koneksi VPN dengan perutean dinamis

`create-vpn-connection` Contoh berikut membuat koneksi VPN antara gateway pribadi virtual yang ditentukan dan gateway pelanggan yang ditentukan, dan menerapkan tag ke koneksi VPN. Outputnya mencakup informasi konfigurasi untuk perangkat gateway pelanggan Anda, dalam format XHTML.

```
aws ec2 create-vpn-connection \  
  --type ipsec.1 \  
  --customer-gateway-id cgw-001122334455aabbcc \  
  --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \  
  --tag-specification 'ResourceType=vpn-connection,Tags=[{Key=Name,Value=BGP-  
VPN}]'
```

### Output:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "...configuration information...",  
    "CustomerGatewayId": "cgw-001122334455aabbcc",  
    "Category": "VPN",  
    "State": "pending",  
    "VpnConnectionId": "vpn-123123123123abccab",  
    "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",  
    "Options": {  
      "EnableAcceleration": false,  
      "StaticRoutesOnly": false,  
      "LocalIpv4NetworkCidr": "0.0.0.0/0",  
      "RemoteIpv4NetworkCidr": "0.0.0.0/0",  
      "TunnelInsideIpVersion": "ipv4",  
      "TunnelOptions": [  
        {},  
        {}  
      ]  
    },  
    "Routes": [],  
    "Tags": [  
      {  
        "Key": "Name",  
        "Value": "BGP-VPN"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Cara kerja AWS VPN Site-to-Site di Panduan Pengguna VPN AWS Site-to-Site](#).

Contoh 2: Untuk membuat koneksi VPN dengan perutean statis

`create-vpn-connection` Contoh berikut membuat koneksi VPN antara gateway pribadi virtual yang ditentukan dan gateway pelanggan yang ditentukan. Opsi menentukan perutean statis. Outputnya mencakup informasi konfigurasi untuk perangkat gateway pelanggan Anda, dalam format XHTML.

```
aws ec2 create-vpn-connection \
  --type ipsec.1 \
  --customer-gateway-id cgw-001122334455aabbc \
  --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \
  --options "{\"StaticRoutesOnly\":true}"
```

Output:

```
{
  "VpnConnection": {
    "CustomerGatewayConfiguration": "..configuration information...",
    "CustomerGatewayId": "cgw-001122334455aabbc",
    "Category": "VPN",
    "State": "pending",
    "VpnConnectionId": "vpn-123123123123abcab",
    "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",
    "Options": {
      "EnableAcceleration": false,
      "StaticRoutesOnly": true,
      "LocalIpv4NetworkCidr": "0.0.0.0/0",
      "RemoteIpv4NetworkCidr": "0.0.0.0/0",
      "TunnelInsideIpVersion": "ipv4",
      "TunnelOptions": [
        {},
        {}
      ]
    },
    "Routes": [],
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Cara kerja AWS VPN Site-to-Site di Panduan Pengguna VPN AWS Site-to-Site](#).

Contoh 3: Untuk membuat koneksi VPN dan menentukan sendiri di dalam CIDR dan kunci yang telah dibagikan sebelumnya

`create-vpn-connection` Contoh berikut membuat koneksi VPN dan menentukan blok CIDR alamat IP bagian dalam dan kunci pra-bersama khusus untuk setiap terowongan. Nilai yang ditentukan dikembalikan dalam `CustomerGatewayConfiguration` informasi.

```
aws ec2 create-vpn-connection \  
  --type ipsec.1 \  
  --customer-gateway-id cgw-001122334455aabbc \  
  --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \  
  --options  
  TunnelOptions='[{TunnelInsideCidr=169.254.12.0/30,PreSharedKey=ExamplePreSharedKey1},  
{TunnelInsideCidr=169.254.13.0/30,PreSharedKey=ExamplePreSharedKey2}]'
```

Output:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "..configuration information..",  
    "CustomerGatewayId": "cgw-001122334455aabbc",  
    "Category": "VPN",  
    "State": "pending",  
    "VpnConnectionId": "vpn-123123123123abcab",  
    "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",  
    "Options": {  
      "EnableAcceleration": false,  
      "StaticRoutesOnly": false,  
      "LocalIpv4NetworkCidr": "0.0.0.0/0",  
      "RemoteIpv4NetworkCidr": "0.0.0.0/0",  
      "TunnelInsideIpVersion": "ipv4",  
      "TunnelOptions": [  
        {  
          "OutsideIpAddress": "203.0.113.3",  
          "TunnelInsideCidr": "169.254.12.0/30",  
          "PreSharedKey": "ExamplePreSharedKey1"  
        },  
        {  
          "OutsideIpAddress": "203.0.113.5",  
          "TunnelInsideCidr": "169.254.13.0/30",  
          "PreSharedKey": "ExamplePreSharedKey2"  
        }  
      ]  
    },  
    "Routes": [],  
    "Tags": []  
  }  
}
```



```
}  
}
```

Untuk informasi selengkapnya, lihat [Cara kerja AWS VPN Site-to-Site di Panduan Pengguna VPN AWS Site-to-Site](#).

Contoh 4: Untuk membuat koneksi VPN yang mendukung lalu lintas IPv6

`create-vpn-connection` Contoh berikut membuat koneksi VPN yang mendukung lalu lintas IPv6 antara gateway transit yang ditentukan dan gateway pelanggan yang ditentukan. Opsi terowongan untuk kedua terowongan menentukan yang AWS harus memulai negosiasi IKE.

```
aws ec2 create-vpn-connection \  
  --type ipsec.1 \  
  --transit-gateway-id tgw-12312312312312312 \  
  --customer-gateway-id cgw-001122334455aabbcc \  
  --options TunnelInsideIpVersion=ipv6,TunnelOptions=[{StartupAction=start},  
  {StartupAction=start}]
```

Output:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "..configuration information...",  
    "CustomerGatewayId": "cgw-001122334455aabbcc",  
    "Category": "VPN",  
    "State": "pending",  
    "VpnConnectionId": "vpn-111111111122222222",  
    "TransitGatewayId": "tgw-12312312312312312",  
    "Options": {  
      "EnableAcceleration": false,  
      "StaticRoutesOnly": false,  
      "LocalIpv6NetworkCidr": "::/0",  
      "RemoteIpv6NetworkCidr": "::/0",  
      "TunnelInsideIpVersion": "ipv6",  
      "TunnelOptions": [  
        {  
          "OutsideIpAddress": "203.0.113.3",  
          "StartupAction": "start"  
        },  
        {  
          "OutsideIpAddress": "203.0.113.5",
```

```
        "StartupAction": "start"
      }
    ]
  },
  "Routes": [],
  "Tags": []
}
}
```

Untuk informasi selengkapnya, lihat [Cara kerja AWS VPN Site-to-Site di Panduan Pengguna VPN AWS Site-to-Site](#).

- Untuk detail API, lihat [CreateVpnConnection](#) di Referensi AWS CLI Perintah.

## create-vpn-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-vpn-gateway`.

### AWS CLI

Untuk membuat gateway pribadi virtual

Contoh ini membuat gateway pribadi virtual.

Perintah:

```
aws ec2 create-vpn-gateway --type ipsec.1
```

Output:

```
{
  "VpnGateway": {
    "AmazonSideAsn": 64512,
    "State": "available",
    "Type": "ipsec.1",
    "VpnGatewayId": "vgw-9a4cacf3",
    "VpcAttachments": []
  }
}
```

Untuk membuat gateway pribadi virtual dengan ASN sisi Amazon tertentu

Contoh ini membuat gateway pribadi virtual dan menentukan Autonomous System Number (ASN) untuk sisi Amazon dari sesi BGP.

Perintah:

```
aws ec2 create-vpn-gateway --type ipsec.1 --amazon-side-asn 65001
```

Output:

```
{
  "VpnGateway": {
    "AmazonSideAsn": 65001,
    "State": "available",
    "Type": "ipsec.1",
    "VpnGatewayId": "vgw-9a4cacf3",
    "VpcAttachments": []
  }
}
```

- Untuk detail API, lihat [CreateVpnGateway](#) di Referensi AWS CLI Perintah.

## delete-carrier-gateway

Contoh kode berikut menunjukkan cara menggunakan `delete-carrier-gateway`.

### AWS CLI

Untuk menghapus gateway operator Anda

`delete-carrier-gateway` Contoh berikut menghapus gateway operator yang ditentukan.

```
aws ec2 delete-carrier-gateway \
  --carrier-gateway-id cagw-0465cdEXAMPLE1111
```

Output:

```
{
  "CarrierGateway": {
    "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
    "VpcId": "vpc-0c529aEXAMPLE1111",
    "State": "deleting",
    "OwnerId": "123456789012"
  }
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Gateway operator](#) di Panduan Pengguna Amazon Virtual Private Cloud.

- Untuk detail API, lihat [DeleteCarrierGateway](#) di Referensi AWS CLI Perintah.

## **delete-client-vpn-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `delete-client-vpn-endpoint`.

### AWS CLI

Untuk menghapus titik akhir Client VPN

`delete-client-vpn-endpoint` Contoh berikut menghapus titik akhir Client VPN yang ditentukan.

```
aws ec2 delete-client-vpn-endpoint \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{  
  "Status": {  
    "Code": "deleting"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir Client VPN](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DeleteClientVpnEndpoint](#) di Referensi AWS CLI Perintah.

## **delete-client-vpn-route**

Contoh kode berikut menunjukkan cara menggunakan `delete-client-vpn-route`.

### AWS CLI

Untuk menghapus rute untuk titik akhir Client VPN

`delete-client-vpn-route` Contoh berikut menghapus `0.0.0.0/0` rute untuk subnet tertentu dari titik akhir Client VPN.

```
aws ec2 delete-client-vpn-route \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --destination-cidr-block 0.0.0.0/0 \  
  --target-vpc-subnet-id subnet-0123456789abcabca
```

Output:

```
{  
  "Status": {  
    "Code": "deleting"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Rute](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DeleteClientVpnRoute](#) di Referensi AWS CLI Perintah.

## `delete-coip-cidr`

Contoh kode berikut menunjukkan cara menggunakan `delete-coip-cidr`.

### AWS CLI

Untuk menghapus berbagai alamat IP (CoIP) milik pelanggan

`delete-coip-cidr` Contoh berikut menghapus rentang alamat CoIP yang ditentukan di kolom CoIP yang ditentukan.

```
aws ec2 delete-coip-cidr \  
  --cidr 14.0.0.0/24 \  
  --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Output:

```
{  
  "CoipCidr": {  
    "Cidr": "14.0.0.0/24",  
    "CoipPoolId": "ipv4pool-coip-1234567890abcdefg",
```

```
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Alamat IP milik pelanggan](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [DeleteCoipCidr](#) di Referensi AWS CLI Perintah.

## delete-coip-pool

Contoh kode berikut menunjukkan cara menggunakan `delete-coip-pool`.

### AWS CLI

Untuk menghapus kumpulan alamat IP (CoIP) milik pelanggan

`delete-coip-pool` Contoh berikut menghapus kumpulan CoIP dari alamat CoIP.

```
aws ec2 delete-coip-pool \  
  --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Output:

```
{  
  "CoipPool": {  
    "PoolId": "ipv4pool-coip-1234567890abcdefg",  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",  
    "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-  
coip-1234567890abcdefg"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Alamat IP milik pelanggan](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [DeleteCoipPool](#) di Referensi AWS CLI Perintah.

## delete-customer-gateway

Contoh kode berikut menunjukkan cara menggunakan `delete-customer-gateway`.

## AWS CLI

Untuk menghapus gateway pelanggan

Contoh ini menghapus gateway pelanggan yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-customer-gateway --customer-gateway-id cgw-0e11f167
```

- Untuk detail API, lihat [DeleteCustomerGateway](#) di Referensi AWS CLI Perintah.

## **delete-dhcp-options**

Contoh kode berikut menunjukkan cara menggunakan `delete-dhcp-options`.

## AWS CLI

Untuk menghapus set opsi DHCP

Contoh ini menghapus set opsi DHCP yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-dhcp-options --dhcp-options-id dopt-d9070ebb
```

- Untuk detail API, lihat [DeleteDhcpOptions](#) di Referensi AWS CLI Perintah.

## **delete-egress-only-internet-gateway**

Contoh kode berikut menunjukkan cara menggunakan `delete-egress-only-internet-gateway`.

## AWS CLI

Untuk menghapus gateway Internet khusus egres

Contoh ini menghapus gateway Internet khusus egres yang ditentukan.

**Perintah:**

```
aws ec2 delete-egress-only-internet-gateway --egress-only-internet-gateway-id
eigw-01eadbd45ecd7943f
```

**Output:**

```
{
  "ReturnCode": true
}
```

- Untuk detail API, lihat [DeleteEgressOnlyInternetGateway](#) di Referensi AWS CLI Perintah.

**delete-fleets**

Contoh kode berikut menunjukkan cara menggunakan `delete-fleets`.

**AWS CLI**

Contoh 1: Untuk menghapus Armada EC2 dan menghentikan instans terkait

`delete-fleets` Contoh berikut menghapus Armada EC2 yang ditentukan dan mengakhiri Instans Sesuai Permintaan dan Instans Spot terkait.

```
aws ec2 delete-fleets \
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
  --terminate-instances
```

**Output:**

```
{
  "SuccessfulFleetDeletions": [
    {
      "CurrentFleetState": "deleted_terminating",
      "PreviousFleetState": "active",
      "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
    }
  ],
  "UnsuccessfulFleetDeletions": []
}
```



Untuk informasi selengkapnya, lihat [Menghapus Armada EC2](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 2: Untuk menghapus Armada EC2 tanpa menghentikan instans terkait

`delete-fleets` Contoh berikut menghapus Armada EC2 yang ditentukan tanpa menghentikan Instans Sesuai Permintaan dan Instans Spot terkait.

```
aws ec2 delete-fleets \  
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \  
  --no-terminate-instances
```

Output:

```
{  
  "SuccessfulFleetDeletions": [  
    {  
      "CurrentFleetState": "deleted_running",  
      "PreviousFleetState": "active",  
      "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"  
    }  
  ],  
  "UnsuccessfulFleetDeletions": []  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Armada EC2](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DeleteFleets](#) di Referensi AWS CLI Perintah.

## **delete-flow-logs**

Contoh kode berikut menunjukkan cara menggunakan `delete-flow-logs`.

AWS CLI

Untuk menghapus log aliran

`delete-flow-logs` Contoh berikut menghapus log aliran yang ditentukan.

```
aws ec2 delete-flow-logs --flow-log-id fl-11223344556677889
```

**Output:**

```
{
  "Unsuccessful": []
}
```

- Untuk detail API, lihat [DeleteFlowLogs](#) di Referensi AWS CLI Perintah.

**delete-fpga-image**

Contoh kode berikut menunjukkan cara menggunakan `delete-fpga-image`.

**AWS CLI**

Untuk menghapus gambar Amazon FPGA

Contoh ini menghapus AFI yang ditentukan.

**Perintah:**

```
aws ec2 delete-fpga-image --fpga-image-id afi-06b12350a123fbabc
```

**Output:**

```
{
  "Return": true
}
```

- Untuk detail API, lihat [DeleteFpgaImage](#) di Referensi AWS CLI Perintah.

**delete-instance-connect-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `delete-instance-connect-endpoint`.

**AWS CLI**

Untuk menghapus Instance Connect Endpoint EC2

`delete-instance-connect-endpoint` Contoh berikut menghapus EC2 Instance Connect Endpoint yang ditentukan.

```
aws ec2 delete-instance-connect-endpoint \  
  --instance-connect-endpoint-id eice-03f5e49b83924bbc7
```

Output:

```
{  
  "InstanceConnectEndpoint": {  
    "OwnerId": "111111111111",  
    "InstanceConnectEndpointId": "eice-0123456789example",  
    "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-  
connect-endpoint/eice-0123456789example",  
    "State": "delete-in-progress",  
    "StateMessage": "",  
    "NetworkInterfaceIds": [],  
    "VpcId": "vpc-0123abcd",  
    "AvailabilityZone": "us-east-1d",  
    "CreatedAt": "2023-02-07T12:05:37+00:00",  
    "SubnetId": "subnet-0123abcd"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Titik Akhir Instans Connect EC2](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DeleteInstanceConnectEndpoint](#) di Referensi AWS CLI Perintah.

## **delete-instance-event-window**

Contoh kode berikut menunjukkan cara menggunakan `delete-instance-event-window`.

### AWS CLI

Contoh 1: Untuk menghapus jendela acara

`delete-instance-event-window` Contoh berikut menghapus jendela acara.

```
aws ec2 delete-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890
```

Output:

```
{
  "InstanceEventWindowState": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "State": "deleting"
  }
}
```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 2: Untuk memaksa menghapus jendela acara

`delete-instance-event-window` Contoh gaya berikut menghapus jendela acara jika jendela acara saat ini dikaitkan dengan target.

```
aws ec2 delete-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --force-delete
```

Output:

```
{
  "InstanceEventWindowState": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "State": "deleting"
  }
}
```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DeleteInstanceEventWindow](#) di Referensi AWS CLI Perintah.

## **delete-internet-gateway**

Contoh kode berikut menunjukkan cara menggunakan `delete-internet-gateway`.

### AWS CLI

Untuk menghapus gateway internet

`delete-internet-gateway` Contoh berikut menghapus gateway internet yang ditentukan.

```
aws ec2 delete-internet-gateway \  
  --internet-gateway-id igw-0d0fb496b3EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Gateway internet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DeleteInternetGateway](#) di Referensi AWS CLI Perintah.

## `delete-ipam-pool`

Contoh kode berikut menunjukkan cara menggunakan `delete-ipam-pool`.

### AWS CLI

Untuk menghapus kolam IPAM

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM yang ingin menghapus kumpulan IPAM yang tidak lagi Anda perlukan, tetapi kumpulan tersebut memiliki CIDR yang disediakan untuk itu. Anda tidak dapat menghapus kumpulan jika memiliki CIDR yang disediakan untuk itu kecuali Anda menggunakan `--cascade` opsi, jadi Anda akan menggunakannya. `--cascade`

Untuk melengkapi permintaan ini:

Anda akan memerlukan ID kolam IPAM yang bisa Anda dapatkan [describe-ipam-pools](#). `--region` Harus menjadi Wilayah rumah IPAM.

`delete-ipam-pool` Contoh berikut menghapus kolam IPAM di akun Anda AWS .

```
aws ec2 delete-ipam-pool \  
  --ipam-pool-id ipam-pool-050c886a3ca41cd5b \  
  --cascade \  
  --region us-east-1
```

Output:

```
{  
  "IpamPool": {  
    "OwnerId": "320805250157",
```

```
"IpamPoolId": "ipam-pool-050c886a3ca41cd5b",
  "IpamPoolArn": "arn:aws:ec2::320805250157:ipam-pool/ipam-
pool-050c886a3ca41cd5b",
  "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-
scope-0a158dde35c51107b",
  "IpamScopeType": "private",
  "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
  "IpamRegion": "us-east-1",
  "Locale": "None",
  "PoolDepth": 1,
  "State": "delete-in-progress",
  "Description": "example",
  "AutoImport": false,
  "AddressFamily": "ipv4",
  "AllocationMinNetmaskLength": 0,
  "AllocationMaxNetmaskLength": 32
}
}
```

Untuk informasi selengkapnya, lihat [Menghapus kumpulan](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DeleteIpamPool](#) di Referensi AWS CLI Perintah.

## **delete-ipam-resource-discovery**

Contoh kode berikut menunjukkan cara menggunakan `delete-ipam-resource-discovery`.

### AWS CLI

Untuk menghapus penemuan sumber daya

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM yang ingin menghapus penemuan sumber daya non-default yang Anda buat untuk dibagikan dengan admin IPAM lain selama proses mengintegrasikan IPAM dengan akun di luar organisasi Anda.

Untuk melengkapi permintaan ini:

`--region` Harus Wilayah tempat Anda membuat penemuan sumber daya. Anda tidak dapat menghapus penemuan sumber daya default jika `"IsDefault": true`. Penemuan sumber daya default adalah penemuan yang dibuat secara otomatis di akun yang membuat IPAM. Untuk menghapus penemuan sumber daya default, Anda harus menghapus IPAM.

`delete-ipam-resource-discovery` Contoh berikut menghapus penemuan sumber daya.

```
aws ec2 delete-ipam-resource-discovery \  
  --ipam-resource-discovery-id ipam-res-disco-0e39761475298ee0f \  
  --region us-east-1
```

Output:

```
{  
  "IpamResourceDiscovery": {  
    "OwnerId": "149977607591",  
    "IpamResourceDiscoveryId": "ipam-res-disco-0e39761475298ee0f",  
    "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-  
discovery/ipam-res-disco-0e39761475298ee0f",  
    "IpamResourceDiscoveryRegion": "us-east-1",  
    "OperatingRegions": [  
      {  
        "RegionName": "us-east-1"  
      }  
    ],  
    "IsDefault": false,  
    "State": "delete-in-progress"  
  }  
}
```

Untuk informasi selengkapnya tentang penemuan sumber daya, lihat [Bekerja dengan penemuan sumber daya di](#) Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DeletelIpamResourceDiscovery](#) di Referensi AWS CLI Perintah.

## `delete-ipam-scope`

Contoh kode berikut menunjukkan cara menggunakan `delete-ipam-scope`.

### AWS CLI

Untuk menghapus cakupan IPAM

`delete-ipam-scope` Contoh berikut menghapus IPAM.

```
aws ec2 delete-ipam-scope \  
  --ipam-scope-id ipam-scope-0e39761475298ee0f \  
  --region us-east-1
```

```
--ipam-scope-id ipam-scope-01c1ebab2b63bd7e4
```

Output:

```
{
  "IpamScope": {
    "OwnerId": "123456789012",
    "IpamScopeId": "ipam-scope-01c1ebab2b63bd7e4",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-01c1ebab2b63bd7e4",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "IpamScopeType": "private",
    "IsDefault": false,
    "Description": "Example description",
    "PoolCount": 0,
    "State": "delete-in-progress"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus cakupan](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DeleteIpamScope](#) di Referensi AWS CLI Perintah.

## delete-ipam

Contoh kode berikut menunjukkan cara menggunakan `delete-ipam`.

AWS CLI

Untuk menghapus IPAM

`delete-ipam` Contoh berikut menghapus IPAM.

```
aws ec2 delete-ipam \
  --ipam-id ipam-036486dfa6af58ee0
```

Output:

```
{
  "Ipam": {
```



```

    "OwnerId": "123456789012",
    "IpamId": "ipam-036486dfa6af58ee0",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-036486dfa6af58ee0",
    "IpamRegion": "us-east-1",
    "PublicDefaultScopeId": "ipam-scope-071b8042b0195c183",
    "PrivateDefaultScopeId": "ipam-scope-0807405dece705a30",
    "ScopeCount": 2,
    "OperatingRegions": [
      {
        "RegionName": "us-east-1"
      },
      {
        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-west-1"
      }
    ],
    "State": "delete-in-progress"
  }
}

```

Untuk informasi selengkapnya, lihat [Menghapus IPAM](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [Deletelpam](#) di Referensi AWS CLI Perintah.

## delete-key-pair

Contoh kode berikut menunjukkan cara menggunakan `delete-key-pair`.

### AWS CLI

Untuk menghapus pasangan kunci

`delete-key-pair` Contoh berikut menghapus key pair yang ditentukan.

```
aws ec2 delete-key-pair \
  --key-name my-key-pair
```

Output:

```
{
  "Return": true,
}
```

```
"KeyPairId": "key-03c8d3aceb53b507"
}
```

Untuk informasi selengkapnya, lihat [Membuat dan menghapus pasangan kunci](#) di Panduan Pengguna Antarmuka Baris AWS Perintah.

- Untuk detail API, lihat [DeleteKeyPair](#) di Referensi AWS CLI Perintah.

## **delete-launch-template-versions**

Contoh kode berikut menunjukkan cara menggunakan `delete-launch-template-versions`.

### AWS CLI

Untuk menghapus versi template peluncuran

Contoh ini menghapus versi template peluncuran yang ditentukan.

Perintah:

```
aws ec2 delete-launch-template-versions --launch-template-id lt-0abcd290751193123 --
versions 1
```

Output:

```
{
  "UnsuccessfullyDeletedLaunchTemplateVersions": [],
  "SuccessfullyDeletedLaunchTemplateVersions": [
    {
      "LaunchTemplateName": "TestVersion",
      "VersionNumber": 1,
      "LaunchTemplateId": "lt-0abcd290751193123"
    }
  ]
}
```

- Untuk detail API, lihat [DeleteLaunchTemplateVersions](#) di Referensi AWS CLI Perintah.

## **delete-launch-template**

Contoh kode berikut menunjukkan cara menggunakan `delete-launch-template`.

## AWS CLI

Untuk menghapus templat peluncuran

Contoh ini menghapus templat peluncuran yang ditentukan.

Perintah:

```
aws ec2 delete-launch-template --launch-template-id lt-0abcd290751193123
```

Output:

```
{
  "LaunchTemplate": {
    "LatestVersionNumber": 2,
    "LaunchTemplateId": "lt-0abcd290751193123",
    "LaunchTemplateName": "TestTemplate",
    "DefaultVersionNumber": 2,
    "CreatedBy": "arn:aws:iam::123456789012:root",
    "CreateTime": "2017-11-23T16:46:25.000Z"
  }
}
```

- Untuk detail API, lihat [DeleteLaunchTemplate](#) di Referensi AWS CLI Perintah.

## delete-local-gateway-route-table-virtual-interface-group-association

Contoh kode berikut menunjukkan cara menggunakan `delete-local-gateway-route-table-virtual-interface-group-association`.

## AWS CLI

Untuk memisahkan tabel rute gateway lokal dari grup antarmuka virtual (VIF)

`delete-local-gateway-route-table-virtual-interface-group-association` Contoh berikut menghapus hubungan antara tabel rute gateway lokal yang ditentukan dan grup VIF.

```
aws ec2 delete-local-gateway-route-table-virtual-interface-group-association \
  --local-gateway-route-table-virtual-interface-group-association-id lgw-vif-grp-
  assoc-exampleid12345678
```

**Output:**

```
{
  "LocalGatewayRouteTableVirtualInterfaceGroupAssociation": {
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-
assoc-exampleid12345678",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-exampleid0123abcd",
    "LocalGatewayId": "lgw-exampleid11223344",
    "LocalGatewayRouteTableId": "lgw-rtb-exampleidabcd1234",
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-exampleidabcd1234",
    "OwnerId": "111122223333",
    "State": "disassociating",
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Asosiasi grup VIF](#) di Panduan AWS Pengguna Outposts.

- Untuk detail API, lihat [DeleteLocalGatewayRouteTableVirtualInterfaceGroupAssociation](#) di Referensi AWS CLI Perintah.

**delete-local-gateway-route-table-vpc-association**

Contoh kode berikut menunjukkan cara menggunakan `delete-local-gateway-route-table-vpc-association`.

**AWS CLI**

Untuk memisahkan tabel rute gateway lokal dari VPC

`delete-local-gateway-route-table-vpc-association` Contoh berikut menghapus hubungan antara tabel rute gateway lokal yang ditentukan dan VPC.

```
aws ec2 delete-local-gateway-route-table-vpc-association \
  --local-gateway-route-table-vpc-association-id vpc-example0123456789
```

**Output:**

```
{
  "LocalGatewayRouteTableVpcAssociation": {
    "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-abcd1234wxyz56789",
```

```

    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:555555555555:local-
gateway-route-table/lgw-rtb-abcdefg1234567890",
    "LocalGatewayId": "lgw-exampleid01234567",
    "VpcId": "vpc-example0123456789",
    "OwnerId": "555555555555",
    "State": "disassociating"
  }
}

```

Untuk informasi selengkapnya, lihat [asosiasi VPC](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [DeleteLocalGatewayRouteTableVpcAssociation](#) di Referensi AWS CLI Perintah.

## delete-local-gateway-route-table

Contoh kode berikut menunjukkan cara menggunakan `delete-local-gateway-route-table`.

### AWS CLI

Untuk menghapus tabel rute gateway lokal

`delete-local-gateway-route-table` Contoh berikut membuat tabel rute gateway lokal dengan modus routing VPC langsung.

```

aws ec2 delete-local-gateway-route-table \
  --local-gateway-route-table-id lgw-rtb-abcdefg1234567890

```

Output:

```

{
  "LocalGatewayRouteTable": {
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-abcdefg1234567890",
    "LocalGatewayId": "lgw-1a2b3c4d5e6f7g8h9",
    "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-021345abcdef67890",
    "OwnerId": "111122223333",
    "State": "deleting",
    "Tags": [],
  }
}

```

```

    "Mode": "direct-vpc-routing"
  }
}

```

Untuk informasi selengkapnya, lihat [Tabel rute gateway lokal](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [DeleteLocalGatewayRouteTable](#) di Referensi AWS CLI Perintah.

## delete-local-gateway-route

Contoh kode berikut menunjukkan cara menggunakan `delete-local-gateway-route`.

### AWS CLI

Untuk menghapus rute dari tabel rute gateway lokal

`delete-local-gateway-route` Contoh berikut menghapus rute yang ditentukan dari tabel rute gateway lokal yang ditentukan.

```

aws ec2 delete-local-gateway-route \
  --destination-cidr-block 0.0.0.0/0 \
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE

```

Output:

```

{
  "Route": {
    "DestinationCidrBlock": "0.0.0.0/0",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
    "Type": "static",
    "State": "deleted",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7EXAMPLE"
  }
}

```

- Untuk detail API, lihat [DeleteLocalGatewayRoute](#) di Referensi AWS CLI Perintah.

## delete-managed-prefix-list

Contoh kode berikut menunjukkan cara menggunakan `delete-managed-prefix-list`.

## AWS CLI

Untuk menghapus daftar awalan

`delete-managed-prefix-list` Contoh berikut menghapus daftar awalan yang ditentukan.

```
aws ec2 delete-managed-prefix-list \
  --prefix-list-id pl-0123456abcabcabc1
```

Output:

```
{
  "PrefixList": {
    "PrefixListId": "pl-0123456abcabcabc1",
    "AddressFamily": "IPv4",
    "State": "delete-in-progress",
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
    "PrefixListName": "test",
    "MaxEntries": 10,
    "Version": 1,
    "OwnerId": "123456789012"
  }
}
```

Untuk informasi selengkapnya, lihat [Daftar awalan terkelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DeleteManagedPrefixList](#) di Referensi AWS CLI Perintah.

## `delete-nat-gateway`

Contoh kode berikut menunjukkan cara menggunakan `delete-nat-gateway`.

## AWS CLI

Untuk menghapus gateway NAT

Contoh ini menghapus gateway NAT. `nat-04ae55e711cec5680`

Perintah:

```
aws ec2 delete-nat-gateway --nat-gateway-id nat-04ae55e711cec5680
```

**Output:**

```
{
  "NatGatewayId": "nat-04ae55e711cec5680"
}
```

- Untuk detail API, lihat [DeleteNatGateway](#) di Referensi AWS CLI Perintah.

**delete-network-acl-entry**

Contoh kode berikut menunjukkan cara menggunakan `delete-network-acl-entry`.

**AWS CLI**

Untuk menghapus entri ACL jaringan

Contoh ini menghapus aturan ingress nomor 100 dari ACL jaringan yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

**Perintah:**

```
aws ec2 delete-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-
number 100
```

- Untuk detail API, lihat [DeleteNetworkAcEntry](#) di Referensi AWS CLI Perintah.

**delete-network-acl**

Contoh kode berikut menunjukkan cara menggunakan `delete-network-acl`.

**AWS CLI**

Untuk menghapus jaringan ACL

Contoh ini menghapus ACL jaringan yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

**Perintah:**

```
aws ec2 delete-network-acl --network-acl-id acl-5fb85d36
```

- Untuk detail API, lihat [DeleteNetworkAcId](#) di Referensi AWS CLI Perintah.



## delete-network-insights-access-scope-analysis

Contoh kode berikut menunjukkan cara menggunakan `delete-network-insights-access-scope-analysis`.

### AWS CLI

Untuk menghapus analisis Network Access Scope

`delete-network-insights-access-scope-analysis` Contoh berikut menghapus analisis Network Access Scope yang ditentukan.

```
aws ec2 delete-network-insights-access-scope-analysis \
  --network-insights-access-scope-analysis-id nisa-01234567891abcdef
```

Output:

```
{
  "NetworkInsightsAccessScopeAnalysisId": "nisa-01234567891abcdef"
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [DeleteNetworkInsightsAccessScopeAnalysis](#) di Referensi AWS CLI Perintah.

## delete-network-insights-access-scope

Contoh kode berikut menunjukkan cara menggunakan `delete-network-insights-access-scope`.

### AWS CLI

Untuk menghapus Cakupan Akses Jaringan

`delete-network-insights-access-scope` Contoh berikut menghapus Network Access Scope yang ditentukan.

```
aws ec2 delete-network-insights-access-scope \
  --network-insights-access-scope-id nis-123456789abc01234
```

Output:

```
{
  "NetworkInsightsAccessScopeId": "nis-123456789abc01234"
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [DeleteNetworkInsightsAccessScoped](#) di Referensi AWS CLI Perintah.

## **delete-network-insights-analysis**

Contoh kode berikut menunjukkan cara menggunakan `delete-network-insights-analysis`.

AWS CLI

Untuk menghapus analisis jalur

`delete-network-insights-analysis` Contoh berikut menghapus analisis yang ditentukan.

```
aws ec2 delete-network-insights-analysis \
  --network-insights-analysis-id nia-02207aa13eb480c7a
```

Output:

```
{
  "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a"
}
```

Untuk informasi selengkapnya, lihat [Memulai menggunakan AWS CLI](#) di Panduan Reachability Analyzer.

- Untuk detail API, lihat [DeleteNetworkInsightsAnalysis](#) di Referensi AWS CLI Perintah.

## **delete-network-insights-path**

Contoh kode berikut menunjukkan cara menggunakan `delete-network-insights-path`.

AWS CLI

Untuk menghapus jalur

`delete-network-insights-path` Contoh berikut menghapus jalur yang ditentukan. Sebelum Anda dapat menghapus jalur, Anda harus menghapus semua analisisnya menggunakan `delete-network-insights-analysis` perintah.

```
aws ec2 delete-network-insights-path \  
  --network-insights-path-id nip-0b26f224f1d131fa8
```

Output:

```
{  
  "NetworkInsightsPathId": "nip-0b26f224f1d131fa8"  
}
```

Untuk informasi selengkapnya, lihat [Memulai menggunakan AWS CLI](#) di Panduan Reachability Analyzer.

- Untuk detail API, lihat [DeleteNetworkInsightsPath](#) di Referensi AWS CLI Perintah.

## **delete-network-interface-permission**

Contoh kode berikut menunjukkan cara menggunakan `delete-network-interface-permission`.

AWS CLI

Untuk menghapus izin antarmuka jaringan

Contoh ini menghapus izin antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 delete-network-interface-permission --network-interface-permission-id eni-  
perm-06fd19020ede149ea
```

Output:

```
{  
  "Return": true  
}
```

- Untuk detail API, lihat [DeleteNetworkInterfacePermission](#) di Referensi AWS CLI Perintah.

## **delete-network-interface**

Contoh kode berikut menunjukkan cara menggunakan `delete-network-interface`.

### AWS CLI

Untuk menghapus antarmuka jaringan

Contoh ini menghapus antarmuka jaringan yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-network-interface --network-interface-id eni-e5aa89a3
```

- Untuk detail API, lihat [DeleteNetworkInterface](#) di Referensi AWS CLI Perintah.

## **delete-placement-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-placement-group`.

### AWS CLI

Untuk menghapus grup penempatan

Perintah contoh ini menghapus grup penempatan yang ditentukan.

Perintah:

```
aws ec2 delete-placement-group --group-name my-cluster
```

- Untuk detail API, lihat [DeletePlacementGroup](#) di Referensi AWS CLI Perintah.

## **delete-queued-reserved-instances**

Contoh kode berikut menunjukkan cara menggunakan `delete-queued-reserved-instances`.

### AWS CLI

Untuk menghapus pembelian antrian

`delete-queued-reserved-instances` Contoh berikut menghapus Instans Cadangan yang ditentukan, yang antri untuk pembelian.

```
aws ec2 delete-queued-reserved-instances \
  --reserved-instances-ids af9f760e-6f91-4559-85f7-4980eexample
```

Output:

```
{
  "SuccessfulQueuedPurchaseDeletions": [
    {
      "ReservedInstancesId": "af9f760e-6f91-4559-85f7-4980eexample"
    }
  ],
  "FailedQueuedPurchaseDeletions": []
}
```

- Untuk detail API, lihat [DeleteQueuedReservedInstances](#) di Referensi AWS CLI Perintah.

## `delete-route-table`

Contoh kode berikut menunjukkan cara menggunakan `delete-route-table`.

### AWS CLI

Untuk menghapus tabel rute

Contoh ini menghapus tabel rute yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-route-table --route-table-id rtb-22574640
```

- Untuk detail API, lihat [DeleteRouteTable](#) di Referensi AWS CLI Perintah.

## `delete-route`

Contoh kode berikut menunjukkan cara menggunakan `delete-route`.

## AWS CLI

Untuk menghapus rute

Contoh ini menghapus rute yang ditentukan dari tabel rute yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-route --route-table-id rtb-22574640 --destination-cidr-block
0.0.0.0/0
```

- Untuk detail API, lihat [DeleteRoute](#) di Referensi AWS CLI Perintah.

## delete-security-group

Contoh kode berikut menunjukkan cara menggunakan `delete-security-group`.

### AWS CLI

[EC2-Classic] Untuk menghapus grup keamanan

Contoh ini menghapus grup keamanan bernama `MySecurityGroup`. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-security-group --group-name MySecurityGroup
```

[EC2-VPC] Untuk menghapus grup keamanan

Contoh ini menghapus grup keamanan dengan ID `sg-903004f8`. Perhatikan bahwa Anda tidak dapat mereferensikan grup keamanan untuk EC2-VPC berdasarkan nama. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-security-group --group-id sg-903004f8
```

Untuk informasi selengkapnya, lihat [Menggunakan Grup Keamanan](#) di Panduan Pengguna Antarmuka Baris Perintah AWS .

- Untuk detail API, lihat [DeleteSecurityGroup](#) di Referensi AWS CLI Perintah.

## **delete-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `delete-snapshot`.

### AWS CLI

Untuk menghapus snapshot

Perintah contoh ini menghapus snapshot dengan ID snapshot `snap-1234567890abcdef0`. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-snapshot --snapshot-id snap-1234567890abcdef0
```

- Untuk detail API, lihat [DeleteSnapshot](#) di Referensi AWS CLI Perintah.

## **delete-spot-datafeed-subscription**

Contoh kode berikut menunjukkan cara menggunakan `delete-spot-datafeed-subscription`.

### AWS CLI

Untuk membatalkan langganan umpan data Instans Spot

Perintah contoh ini menghapus langganan umpan data Spot untuk akun tersebut. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-spot-datafeed-subscription
```

- Untuk detail API, lihat [DeleteSpotDatafeedSubscription](#) di Referensi AWS CLI Perintah.

## **delete-subnet-cidr-reservation**

Contoh kode berikut menunjukkan cara menggunakan `delete-subnet-cidr-reservation`.

## AWS CLI

Untuk menghapus reservasi CIDR subnet

`delete-subnet-cidr-reservation` Contoh berikut menghapus reservasi CIDR subnet yang ditentukan.

```
aws ec2 delete-subnet-cidr-reservation \
  --subnet-cidr-reservation-id scr-044f977c4eEXAMPLE
```

Output:

```
{
  "DeletedSubnetCidrReservation": {
    "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
    "SubnetId": "subnet-03c51e2e6cEXAMPLE",
    "Cidr": "10.1.0.16/28",
    "ReservationType": "prefix",
    "OwnerId": "123456789012"
  }
}
```

Untuk informasi selengkapnya, lihat [Reservasi CIDR Subnet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DeleteSubnetCidrReservation](#) di Referensi AWS CLI Perintah.

## `delete-subnet`

Contoh kode berikut menunjukkan cara menggunakan `delete-subnet`.

### AWS CLI

Untuk menghapus subnet

Contoh ini menghapus subnet yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-subnet --subnet-id subnet-9d4a7b6c
```

- Untuk detail API, lihat [DeleteSubnet](#) di Referensi AWS CLI Perintah.



## delete-tags

Contoh kode berikut menunjukkan cara menggunakan `delete-tags`.

### AWS CLI

Contoh 1: Untuk menghapus tag dari sumber daya

`delete-tags` Contoh berikut menghapus tag `Stack=Test` dari gambar yang ditentukan. Saat Anda menentukan nilai dan nama kunci, tag akan dihapus hanya jika nilai tag cocok dengan nilai yang ditentukan.

```
aws ec2 delete-tags \  
  --resources ami-1234567890abcdef0 \  
  --tags Key=Stack,Value=Test
```

Ini opsional untuk menentukan nilai untuk tag. `delete-tags` Contoh berikut menghapus tag dengan nama kunci `purpose` dari contoh yang ditentukan, terlepas dari nilai tag untuk tag.

```
aws ec2 delete-tags \  
  --resources i-1234567890abcdef0 \  
  --tags Key=purpose
```

Jika Anda menentukan string kosong sebagai nilai tag, tag akan dihapus hanya jika nilai tag adalah string kosong. `delete-tags` Contoh berikut menentukan string kosong sebagai nilai tag untuk tag untuk menghapus.

```
aws ec2 delete-tags \  
  --resources i-1234567890abcdef0 \  
  --tags Key=Name,Value=
```

Contoh 2: Untuk menghapus tag dari beberapa sumber daya

`delete-tags` Contoh berikut menghapus tag `purpose=test` dari kedua instance dan AMI. Seperti yang ditunjukkan pada contoh sebelumnya, Anda dapat menghilangkan nilai tag dari perintah.

```
aws ec2 delete-tags \  
  --resources i-1234567890abcdef0 ami-1234567890abcdef0 \  
  --tags Key=purpose,Value=test
```

```
--tags Key=Purpose
```

- Untuk detail API, lihat [DeleteTags](#) di Referensi AWS CLI Perintah.

## **delete-traffic-mirror-filter-rule**

Contoh kode berikut menunjukkan cara menggunakan `delete-traffic-mirror-filter-rule`.

### AWS CLI

Untuk menghapus aturan filter cermin lalu lintas

`delete-traffic-mirror-filter-rule` Contoh berikut menghapus aturan filter cermin lalu lintas yang ditentukan.

```
aws ec2 delete-traffic-mirror-filter-rule \  
  --traffic-mirror-filter-rule-id tmfr-081f71283bEXAMPLE
```

Output:

```
{  
  "TrafficMirrorFilterRuleId": "tmfr-081f71283bEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Aturan Filter Cermin Lalu Lintas Anda](#) di Panduan Pencermian AWS Lalu Lintas.

- Untuk detail API, lihat [DeleteTrafficMirrorFilterRule](#) di Referensi AWS CLI Perintah.

## **delete-traffic-mirror-filter**

Contoh kode berikut menunjukkan cara menggunakan `delete-traffic-mirror-filter`.

### AWS CLI

Untuk menghapus filter cermin lalu lintas

`delete-traffic-mirror-filter` Contoh berikut menghapus filter cermin lalu lintas yang ditentukan.

```
aws ec2 delete-traffic-mirror-filter \  
  --filter-id
```

```
--traffic-mirror-filter-id tmf-0be0b25fcdEXAMPLE
```

Output:

```
{
  "TrafficMirrorFilterId": "tmf-0be0b25fcdEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menghapus Filter Cermin Lalu Lintas](#) di Panduan Pencerminan AWS Lalu Lintas.

- Untuk detail API, lihat [DeleteTrafficMirrorFilter](#) di Referensi AWS CLI Perintah.

## **delete-traffic-mirror-session**

Contoh kode berikut menunjukkan cara menggunakan `delete-traffic-mirror-session`.

AWS CLI

Untuk menghapus sesi cermin lalu lintas

`delete-traffic-mirror-session` Contoh berikut menghapus sesi cermin lalu lintas yang ditentukan.

```
aws ec2 delete-traffic-mirror-session \
  --traffic-mirror-session-id tms-0af3141ce5EXAMPLE
```

Output:

```
{
  "TrafficMirrorSessionId": "tms-0af3141ce5EXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menghapus Sesi Cermin Lalu Lintas](#) di Panduan Pencerminan AWS Lalu Lintas.

- Untuk detail API, lihat [DeleteTrafficMirrorSession](#) di Referensi AWS CLI Perintah.

## **delete-traffic-mirror-target**

Contoh kode berikut menunjukkan cara menggunakan `delete-traffic-mirror-target`.

## AWS CLI

Untuk menghapus target cermin lalu lintas

`delete-traffic-mirror-target` Contoh berikut menghapus target cermin lalu lintas yang ditentukan.

```
aws ec2 delete-traffic-mirror-target \  
  --traffic-mirror-target-id tmt-060f48ce9EXAMPLE
```

Output:

```
{  
  "TrafficMirrorTargetId": "tmt-060f48ce9EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Target Cermin Lalu Lintas](#) di Panduan Pencermian AWS Lalu Lintas.

- Untuk detail API, lihat [DeleteTrafficMirrorTarget](#) di Referensi AWS CLI Perintah.

## `delete-transit-gateway-connect-peer`

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-connect-peer`.

## AWS CLI

Untuk menghapus rekan Transit Gateway Connect

`delete-transit-gateway-connect-peer` Contoh berikut menghapus rekan Connect yang ditentukan.

```
aws ec2 delete-transit-gateway-connect-peer \  
  --transit-gateway-connect-peer-id tgw-connect-peer-0666adbac4EXAMPLE
```

Output:

```
{  
  "TransitGatewayConnectPeer": {
```

```

"TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
"TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
"State": "deleting",
"CreationTime": "2021-10-13T03:35:17.000Z",
"ConnectPeerConfiguration": {
  "TransitGatewayAddress": "10.0.0.234",
  "PeerAddress": "172.31.1.11",
  "InsideCidrBlocks": [
    "169.254.6.0/29"
  ],
  "Protocol": "gre",
  "BgpConfigurations": [
    {
      "TransitGatewayAsn": 64512,
      "PeerAsn": 64512,
      "TransitGatewayAddress": "169.254.6.2",
      "PeerAddress": "169.254.6.1",
      "BgpStatus": "down"
    },
    {
      "TransitGatewayAsn": 64512,
      "PeerAsn": 64512,
      "TransitGatewayAddress": "169.254.6.3",
      "PeerAddress": "169.254.6.1",
      "BgpStatus": "down"
    }
  ]
}
}
}
}
}

```

Untuk informasi selengkapnya, lihat [Lampiran Transit gateway Connect dan rekan Transit Gateway Connect](#) di Panduan Transit Gateway.

- Untuk detail API, lihat [DeleteTransitGatewayConnectPeer](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway-connect

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-connect`.

### AWS CLI

Untuk menghapus lampiran Connect gateway transit

`delete-transit-gateway-connect` Contoh berikut menghapus lampiran Connect yang ditentukan.

```
aws ec2 delete-transit-gateway-connect \  
  --transit-gateway-attachment-id tgw-attach-037012e5dcEXAMPLE
```

Output:

```
{  
  "TransitGatewayConnect": {  
    "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",  
    "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",  
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",  
    "State": "deleting",  
    "CreationTime": "2021-03-09T19:59:17+00:00",  
    "Options": {  
      "Protocol": "gre"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Lampiran Transit gateway Connect dan rekan Transit Gateway Connect](#) di Panduan Transit Gateway.

- Untuk detail API, lihat [DeleteTransitGatewayConnect](#) di Referensi AWS CLI Perintah.

## `delete-transit-gateway-multicast-domain`

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-multicast-domain`.

### AWS CLI

Untuk menghapus domain multicast gateway transit

`delete-transit-gateway-multicast-domain` Contoh berikut menghapus domain multicast yang ditentukan.

```
aws ec2 delete-transit-gateway-multicast-domain \  
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{
  "TransitGatewayMulticastDomain": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-02bb79002bEXAMPLE",
    "TransitGatewayId": "tgw-0d88d2d0d5EXAMPLE",
    "State": "deleting",
    "CreationTime": "2019-11-20T22:02:03.000Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Transit Gateways.

- Untuk detail API, lihat [Delete TransitGatewayMulticastDomain](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway-peering-attachment

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-peering-attachment`.

### AWS CLI

Untuk menghapus lampiran peering gateway transit

`delete-transit-gateway-peering-attachment` Contoh berikut menghapus lampiran peering gateway transit yang ditentukan.

```
aws ec2 delete-transit-gateway-peering-attachment \
  --transit-gateway-attachment-id tgw-attach-4455667788aabbccd
```

### Output:

```
{
  "TransitGatewayPeeringAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
    "RequesterTgwInfo": {
      "TransitGatewayId": "tgw-123abc05e04123abc",
      "OwnerId": "123456789012",
      "Region": "us-west-2"
    },
    "AcceptorTgwInfo": {
      "TransitGatewayId": "tgw-11223344aabbcc112",
      "OwnerId": "123456789012",
      "Region": "us-east-2"
    }
  }
}
```

```

    },
    "State": "deleting",
    "CreationTime": "2019-12-09T11:38:31.000Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Lampiran Pengintip Transit Gateway](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DeleteTransitGatewayPeeringAttachment](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway-policy-table

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-policy-table`.

### AWS CLI

Untuk menghapus tabel kebijakan gateway transit

`delete-transit-gateway-policy-table` Contoh berikut menghapus tabel kebijakan gateway transit yang ditentukan.

```

aws ec2 delete-transit-gateway-policy-table \
  --transit-gateway-policy-table-id tgw-ptb-0a16f134b78668a81

```

Output:

```

{
  "TransitGatewayPolicyTables": [
    {
      "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
      "TransitGatewayId": "tgw-067f8505c18f0bd6e",
      "State": "deleting",
      "CreationTime": "2023-11-28T16:36:43+00:00",
      "Tags": []
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Tabel kebijakan gateway transit](#) di Panduan Pengguna Gateway Transit.



- Untuk detail API, lihat [DeleteTransitGatewayPolicyTable](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway-prefix-list-reference

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-prefix-list-reference`.

### AWS CLI

Untuk menghapus referensi daftar awalan

`delete-transit-gateway-prefix-list-reference` Contoh berikut menghapus referensi daftar awalan yang ditentukan.

```
aws ec2 delete-transit-gateway-prefix-list-reference \
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
  --prefix-list-id pl-11111122222222333
```

Output:

```
{
  "TransitGatewayPrefixListReference": {
    "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
    "PrefixListId": "pl-11111122222222333",
    "PrefixListOwnerId": "123456789012",
    "State": "deleting",
    "Blackhole": false,
    "TransitGatewayAttachment": {
      "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
      "ResourceType": "vpc",
      "ResourceId": "vpc-112233445566aabbcc"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Referensi daftar awalan](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DeleteTransitGatewayPrefixListReference](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway-route-table

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-route-table`.

## AWS CLI

Untuk menghapus tabel rute gateway transit

`delete-transit-gateway-route-table` Contoh berikut menghapus tabel rute gateway transit yang ditentukan.

```
aws ec2 delete-transit-gateway-route-table \  
  --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE
```

Output:

```
{  
  "TransitGatewayRouteTable": {  
    "TransitGatewayRouteTableId": "tgw-rtb-0b6f6aaa01EXAMPLE",  
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",  
    "State": "deleting",  
    "DefaultAssociationRouteTable": false,  
    "DefaultPropagationRouteTable": false,  
    "CreationTime": "2019-07-17T20:27:26.000Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus tabel rute gateway transit](#) di Panduan Gateway Transit.

- Untuk detail API, lihat [DeleteTransitGatewayRouteTable](#) di Referensi AWS CLI Perintah.

## `delete-transit-gateway-route`

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-route`.

## AWS CLI

Untuk menghapus blok CIDR dari tabel rute

`delete-transit-gateway-route` Contoh berikut menghapus blok CIDR dari tabel rute gateway transit yang ditentukan.

```
aws ec2 delete-transit-gateway-route \  
  --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE \  
  --cidr-block-association-id tgw-rtb-0b6f6aaa01EXAMPLE
```

```
--destination-cidr-block 10.0.2.0/24
```

Output:

```
{
  "Route": {
    "DestinationCidrBlock": "10.0.2.0/24",
    "TransitGatewayAttachments": [
      {
        "ResourceId": "vpc-0065acced4EXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
        "ResourceType": "vpc"
      }
    ],
    "Type": "static",
    "State": "deleted"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus rute statis](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DeleteTransitGatewayRoute](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway-vpc-attachment

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway-vpc-attachment`.

### AWS CLI

Untuk menghapus lampiran VPC gateway transit

`delete-transit-gateway-vpc-attachment` Contoh berikut menghapus lampiran VPC yang ditentukan.

```
aws ec2 delete-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-0d2c54bdbEXAMPLE
```

Output:

```
{
  "TransitGatewayVpcAttachment": {
```

```

    "TransitGatewayAttachmentId": "tgw-attach-0d2c54bdb3EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "VpcId": "vpc-0065acced4f61c651",
    "VpcOwnerId": "111122223333",
    "State": "deleting",
    "CreationTime": "2019-07-17T16:04:27.000Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Menghapus lampiran VPC di Panduan](#) Gerbang Transit.

- Untuk detail API, lihat [DeleteTransitGatewayVpcAttachment](#) di Referensi AWS CLI Perintah.

## delete-transit-gateway

Contoh kode berikut menunjukkan cara menggunakan `delete-transit-gateway`.

### AWS CLI

Untuk menghapus gateway transit

`delete-transit-gateway` Contoh berikut menghapus gateway transit yang ditentukan.

```

aws ec2 delete-transit-gateway \
  --transit-gateway-id tgw-01f04542b2EXAMPLE

```

Output:

```

{
  "TransitGateway": {
    "TransitGatewayId": "tgw-01f04542b2EXAMPLE",
    "State": "deleting",
    "OwnerId": "123456789012",
    "Description": "Example Transit Gateway",
    "CreationTime": "2019-08-27T15:04:35.000Z",
    "Options": {
      "AmazonSideAsn": 64515,
      "AutoAcceptSharedAttachments": "disable",
      "DefaultRouteTableAssociation": "enable",
      "AssociationDefaultRouteTableId": "tgw-rtb-0ce7a6948fEXAMPLE",
      "DefaultRouteTablePropagation": "enable",
      "PropagationDefaultRouteTableId": "tgw-rtb-0ce7a6948fEXAMPLE",
      "VpnEcmpSupport": "enable",
    }
  }
}

```

```

        "DnsSupport": "enable"
    }
}
}

```

Untuk informasi selengkapnya, lihat [Menghapus gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [Delete Transit Gateway](#) di Referensi AWS CLI Perintah.

## delete-verified-access-endpoint

Contoh kode berikut menunjukkan cara menggunakan `delete-verified-access-endpoint`.

### AWS CLI

Untuk menghapus titik akhir Akses Terverifikasi

`delete-verified-access-endpoint` Contoh berikut menghapus titik akhir Akses Terverifikasi yang ditentukan.

```

aws ec2 delete-verified-access-endpoint \
  --verified-access-endpoint-id vae-066fac616d4d546f2

```

### Output:

```

{
  "VerifiedAccessEndpoint": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
    "ApplicationDomain": "example.com",
    "EndpointType": "network-interface",
    "AttachmentType": "vpc",
    "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
    "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
    "SecurityGroupIds": [
      "sg-004915970c4c8f13a"
    ],
    "NetworkInterfaceOptions": {

```

```

        "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
        "Protocol": "https",
        "Port": 443
    },
    "Status": {
        "Code": "deleting"
    },
    "Description": "Testing Verified Access",
    "CreationTime": "2023-08-25T20:54:43",
    "LastUpdatedTime": "2023-08-25T22:46:32"
}
}

```

Untuk informasi selengkapnya, lihat [Titik akhir Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DeleteVerifiedAccessEndpoint](#) di Referensi AWS CLI Perintah.

## delete-verified-access-group

Contoh kode berikut menunjukkan cara menggunakan `delete-verified-access-group`.

### AWS CLI

Untuk menghapus grup Akses Terverifikasi

`delete-verified-access-group` Contoh berikut menghapus grup Akses Terverifikasi yang ditentukan.

```
aws ec2 delete-verified-access-group \
  --verified-access-group-id vagr-0dbe967baf14b7235
```

Output:

```
{
  "VerifiedAccessGroup": {
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "Owner": "123456789012",
    "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
  }
}
```

```
    "CreationTime": "2023-08-25T19:55:19",
    "LastUpdatedTime": "2023-08-25T22:49:03",
    "DeletionTime": "2023-08-26T00:58:31"
  }
}
```

Untuk informasi selengkapnya, lihat [Grup Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DeleteVerifiedAccessGroup](#) di Referensi AWS CLI Perintah.

## **delete-verified-access-instance**

Contoh kode berikut menunjukkan cara menggunakan `delete-verified-access-instance`.

### AWS CLI

Untuk menghapus instans Akses Terverifikasi

`delete-verified-access-instance` Contoh berikut menghapus instance Akses Terverifikasi yang ditentukan.

```
aws ec2 delete-verified-access-instance \
  --verified-access-instance-id vai-0ce000c0b7643abea
```

Output:

```
{
  "VerifiedAccessInstance": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "VerifiedAccessTrustProviders": [],
    "CreationTime": "2023-08-25T18:27:56",
    "LastUpdatedTime": "2023-08-26T01:00:18"
  }
}
```

Untuk informasi selengkapnya, lihat [Instans Akses Terverifikasi](#) di Panduan Pengguna Akses AWS Terverifikasi.

- Untuk detail API, lihat [DeleteVerifiedAccessInstance](#) di Referensi AWS CLI Perintah.

## delete-verified-access-trust-provider

Contoh kode berikut menunjukkan cara menggunakan `delete-verified-access-trust-provider`.

### AWS CLI

Untuk menghapus penyedia kepercayaan Akses Terverifikasi

`delete-verified-access-trust-provider` Contoh berikut menghapus penyedia kepercayaan Akses Terverifikasi yang ditentukan.

```
aws ec2 delete-verified-access-trust-provider \
  --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

### Output:

```
{
  "VerifiedAccessTrustProvider": {
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
    "Description": "Testing Verified Access",
    "TrustProviderType": "user",
    "UserTrustProviderType": "iam-identity-center",
    "PolicyReferenceName": "idc",
    "CreationTime": "2023-08-25T18:40:36",
    "LastUpdatedTime": "2023-08-25T18:40:36"
  }
}
```

Untuk informasi selengkapnya, lihat [Penyedia kepercayaan untuk Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DeleteVerifiedAccessTrustProvider](#) di Referensi AWS CLI Perintah.

## delete-volume

Contoh kode berikut menunjukkan cara menggunakan `delete-volume`.

### AWS CLI

Untuk menghapus volume



Perintah contoh ini menghapus volume yang tersedia dengan ID volume dari.

vol-049df61146c4d7901 Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-volume --volume-id vol-049df61146c4d7901
```

- Untuk detail API, lihat [DeleteVolume](#) di Referensi AWS CLI Perintah.

## **delete-vpc-endpoint-connection-notifications**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpc-endpoint-connection-notifications`.

AWS CLI

Untuk menghapus pemberitahuan koneksi titik akhir

Contoh ini menghapus pemberitahuan koneksi titik akhir yang ditentukan.

Perintah:

```
aws ec2 delete-vpc-endpoint-connection-notifications --connection-notification-ids  
vpce-nfn-008776de7e03f5abc
```

Output:

```
{  
  "Unsuccessful": []  
}
```

- Untuk detail API, lihat [DeleteVpcEndpointConnectionNotifications](#) di Referensi AWS CLI Perintah.

## **delete-vpc-endpoint-service-configurations**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpc-endpoint-service-configurations`.

## AWS CLI

Untuk menghapus konfigurasi layanan endpoint

Contoh ini menghapus konfigurasi layanan endpoint yang ditentukan.

Perintah:

```
aws ec2 delete-vpc-endpoint-service-configurations --service-ids vpce-  
svc-03d5ebb7d9579a2b3
```

Output:

```
{  
  "Unsuccessful": []  
}
```

- Untuk detail API, lihat [DeleteVpcEndpointServiceConfigurations](#) di Referensi AWS CLI Perintah.

## delete-vpc-endpoints

Contoh kode berikut menunjukkan cara menggunakan `delete-vpc-endpoints`.

### AWS CLI

Untuk menghapus titik akhir

Contoh ini menghapus titik akhir `vpce-aa22bb33` dan `vpce-1a2b3c4d`. Jika perintah sebagian berhasil atau tidak berhasil, daftar item yang gagal dikembalikan. Jika perintah berhasil, daftar yang dikembalikan kosong.

Perintah:

```
aws ec2 delete-vpc-endpoints --vpc-endpoint-ids vpce-aa22bb33 vpce-1a2b3c4d
```

Output:

```
{  
  "Unsuccessful": []  
}
```

```
}
```

- Untuk detail API, lihat [DeleteVpcEndpoints](#) di Referensi AWS CLI Perintah.

## **delete-vpc-peering-connection**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpc-peering-connection`.

### AWS CLI

Untuk menghapus koneksi peering VPC

Contoh ini menghapus koneksi peering VPC yang ditentukan.

Perintah:

```
aws ec2 delete-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [DeleteVpcPeeringConnection](#) di Referensi AWS CLI Perintah.

## **delete-vpc**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpc`.

### AWS CLI

Untuk menghapus VPC

Contoh ini menghapus VPC yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-vpc --vpc-id vpc-a01106c2
```

- Untuk detail API, lihat [DeleteVpc](#) di Referensi AWS CLI Perintah.

## **delete-vpn-connection-route**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpn-connection-route`.

### AWS CLI

Untuk menghapus rute statis dari koneksi VPN

Contoh ini menghapus rute statis yang ditentukan dari koneksi VPN yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-vpn-connection-route --vpn-connection-id vpn-40f41529 --destination-cidr-block 11.12.0.0/16
```

- Untuk detail API, lihat [DeleteVpnConnectionRoute](#) di Referensi AWS CLI Perintah.

## **delete-vpn-connection**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpn-connection`.

### AWS CLI

Untuk menghapus koneksi VPN

Contoh ini menghapus koneksi VPN yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-vpn-connection --vpn-connection-id vpn-40f41529
```

- Untuk detail API, lihat [DeleteVpnConnection](#) di Referensi AWS CLI Perintah.

## **delete-vpn-gateway**

Contoh kode berikut menunjukkan cara menggunakan `delete-vpn-gateway`.

## AWS CLI

Untuk menghapus gateway pribadi virtual

Contoh ini menghapus gateway pribadi virtual yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 delete-vpn-gateway --vpn-gateway-id vgw-9a4cacf3
```

- Untuk detail API, lihat [DeleteVpnGateway](#) di Referensi AWS CLI Perintah.

## deprovision-byoip-cidr

Contoh kode berikut menunjukkan cara menggunakan `deprovision-byoip-cidr`.

## AWS CLI

Untuk menghapus rentang alamat IP dari penggunaan

Contoh berikut menghapus rentang alamat yang ditentukan dari penggunaan dengan AWS.

```
aws ec2 deprovision-byoip-cidr \  
  --cidr 203.0.113.25/24
```

Output:

```
{  
  "ByoipCidr": {  
    "Cidr": "203.0.113.25/24",  
    "State": "pending-deprovision"  
  }  
}
```

- Untuk detail API, lihat [DeprovisionByoipCidr](#) di Referensi AWS CLI Perintah.

## deprovision-ipam-pool-cidr

Contoh kode berikut menunjukkan cara menggunakan `deprovision-ipam-pool-cidr`.

## AWS CLI

Untuk menghentikan penyediaan CIDR kolam IPAM

`deprovision-ipam-pool-cidr` Contoh berikut membatalkan ketentuan CIDR yang disediakan ke kolam IPAM.

(Linux):

```
aws ec2 deprovision-ipam-pool-cidr \  
  --ipam-pool-id ipam-pool-02ec043a19bbe5d08 \  
  --cidr 11.0.0.0/16
```

(Windows):

```
aws ec2 deprovision-ipam-pool-cidr ^  
  --ipam-pool-id ipam-pool-02ec043a19bbe5d08 ^  
  --cidr 11.0.0.0/16
```

Output:

```
{  
  "IpamPoolCidr": {  
    "Cidr": "11.0.0.0/16",  
    "State": "pending-deprovision"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Deprovision pool CIDR di Panduan Pengguna Amazon VPC IPAM](#).

- Untuk detail API, lihat [DeprovisionIpamPoolCidr](#) di Referensi AWS CLI Perintah.

## **deregister-image**

Contoh kode berikut menunjukkan cara menggunakan `deregister-image`.

## AWS CLI

Untuk membatalkan pendaftaran AMI

Contoh ini membatalkan pendaftaran AMI yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 deregister-image --image-id ami-4fa54026
```

- Untuk detail API, lihat [DeregisterImage](#) di Referensi AWS CLI Perintah.

## deregister-instance-event-notification-attributes

Contoh kode berikut menunjukkan cara menggunakan `deregister-instance-event-notification-attributes`.

AWS CLI

Contoh 1: Untuk menghapus semua tag dari pemberitahuan acara

`deregister-instance-event-notification-attributes` Contoh berikut menghapus `IncludeAllTagsOfInstance=true`, yang memiliki efek pengaturan `IncludeAllTagsOfInstance` ke `false`.

```
aws ec2 deregister-instance-event-notification-attributes \  
  --instance-tag-attribute IncludeAllTagsOfInstance=true
```

Output:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [],  
    "IncludeAllTagsOfInstance": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Acara terjadwal untuk instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 2: Untuk menghapus tag tertentu dari pemberitahuan acara

`deregister-instance-event-notification-attributes` Contoh berikut menghapus tag yang ditentukan dari tag yang disertakan dalam pemberitahuan acara. Untuk menjelaskan

tag yang tersisa yang disertakan dalam pemberitahuan acara, gunakan `describe-instance-event-notification-attributes`.

```
aws ec2 deregister-instance-event-notification-attributes \  
  --instance-tag-attribute InstanceTagKeys="tag-key2"
```

Output:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [  
      "tag-key2"  
    ],  
    "IncludeAllTagsOfInstance": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Acara terjadwal untuk instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DeregisterInstanceEventNotificationAttributes](#) di Referensi AWS CLI Perintah.

## deregister-transit-gateway-multicast-group-members

Contoh kode berikut menunjukkan cara menggunakan `deregister-transit-gateway-multicast-group-members`.

AWS CLI

Untuk membatalkan pendaftaran anggota grup dari grup multicast

Contoh ini membatalkan pendaftaran anggota grup antarmuka jaringan yang ditentukan dari grup multicast gateway transit.

```
aws ec2 deregister-transit-gateway-multicast-group-members \  
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE \  
  --group-ip-address 224.0.1.0 \  
  --network-interface-ids eni-0e246d3269EXAMPLE
```

Output:



```
{
  "DeregisteredMulticastGroupMembers": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef7EXAMPLE",
    "RegisteredNetworkInterfaceIds": [
      "eni-0e246d3269EXAMPLE"
    ],
    "GroupIpAddress": "224.0.1.0"
  }
}
```

Untuk informasi selengkapnya, lihat [membatalkan pendaftaran Anggota dari Grup Multicast](#) di Panduan Pengguna Transit Gateways.AWS

- Untuk detail API, lihat [DeregisterTransitGatewayMulticastGroupMembers](#) di Referensi AWS CLI Perintah.

## deregister-transit-gateway-multicast-group-source

Contoh kode berikut menunjukkan cara menggunakan `deregister-transit-gateway-multicast-group-source`.

### AWS CLI

Untuk membatalkan pendaftaran sumber dari grup multicast gateway transit

Contoh ini membatalkan pendaftaran sumber grup antarmuka jaringan yang ditentukan dari grup multicast.

```
aws ec2 register-transit-gateway-multicast-group-sources \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-07f290fc3c090cbae
```

Output:

```
{
  "DeregisteredMulticastGroupSources": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "DeregisteredNetworkInterfaceIds": [
      "eni-07f290fc3c090cbae"
    ],
  },
}
```

```
    "GroupIpAddress": "224.0.1.0"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Deregister Sources from a Multicast Group](#) di Panduan Pengguna Transit Gateways.AWS

- Untuk detail API, lihat [DeregisterTransitGatewayMulticastGroupSource](#) di Referensi AWS CLI Perintah.

## describe-account-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-account-attributes`.

### AWS CLI

Untuk mendeskripsikan semua atribut untuk AWS akun Anda

Contoh ini menjelaskan atribut untuk AWS akun Anda.

Perintah:

```
aws ec2 describe-account-attributes
```

Output:

```
{  
  "AccountAttributes": [  
    {  
      "AttributeName": "vpc-max-security-groups-per-interface",  
      "AttributeValues": [  
        {  
          "AttributeValue": "5"  
        }  
      ]  
    },  
    {  
      "AttributeName": "max-instances",  
      "AttributeValues": [  
        {  
          "AttributeValue": "20"  
        }  
      ]  
    }  
  ]  
}
```

```
    },
    {
      "AttributeName": "supported-platforms",
      "AttributeValues": [
        {
          "AttributeValue": "EC2"
        },
        {
          "AttributeValue": "VPC"
        }
      ]
    },
    {
      "AttributeName": "default-vpc",
      "AttributeValues": [
        {
          "AttributeValue": "none"
        }
      ]
    },
    {
      "AttributeName": "max-elastic-ips",
      "AttributeValues": [
        {
          "AttributeValue": "5"
        }
      ]
    },
    {
      "AttributeName": "vpc-max-elastic-ips",
      "AttributeValues": [
        {
          "AttributeValue": "5"
        }
      ]
    }
  ]
}
```

Untuk mendeskripsikan satu atribut untuk AWS akun Anda

Contoh ini menjelaskan supported-platforms atribut untuk AWS akun Anda.

Perintah:

```
aws ec2 describe-account-attributes --attribute-names supported-platforms
```

Output:

```
{
  "AccountAttributes": [
    {
      "AttributeName": "supported-platforms",
      "AttributeValues": [
        {
          "AttributeValue": "EC2"
        },
        {
          "AttributeValue": "VPC"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeAccountAttributes](#) di Referensi AWS CLI Perintah.

## describe-address-transfers

Contoh kode berikut menunjukkan cara menggunakan `describe-address-transfers`.

AWS CLI

Untuk menggambarkan transfer alamat IP Elastis

`describe-address-transfers` Contoh berikut menjelaskan transfer alamat IP Elastis untuk alamat IP Elastis yang ditentukan.

```
aws ec2 describe-address-transfers \
  --allocation-ids eipalloc-09ad461b0d03f6aaf
```

Output:

```
{
  "AddressTransfers": [
```

```
{
  "PublicIp": "100.21.184.216",
  "AllocationId": "eipalloc-09ad461b0d03f6aaf",
  "TransferAccountId": "123456789012",
  "TransferOfferExpirationTimestamp": "2023-02-22T22:51:01.000Z",
  "AddressTransferStatus": "pending"
}
]
```

Untuk informasi selengkapnya, lihat [Mentransfer alamat IP Elastis](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeAddressTransfers](#) di Referensi AWS CLI Perintah.

## describe-addresses-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-addresses-attribute`.

### AWS CLI

Untuk melihat atribut nama domain yang terkait dengan alamat IP elastis

`describe-addresses-attribute` Contoh berikut mengembalikan atribut nama domain yang terkait dengan alamat IP elastis.

Linux:

```
aws ec2 describe-addresses-attribute \
  --allocation-ids eipalloc-abcdef01234567890 \
  --attribute domain-name
```

Windows:

```
aws ec2 describe-addresses-attribute ^
  --allocation-ids eipalloc-abcdef01234567890 ^
  --attribute domain-name
```

Output:

```
{
```

```
"Addresses": [  
  {  
    "PublicIp": "192.0.2.0",  
    "AllocationId": "eipalloc-abcdef01234567890",  
    "PtrRecord": "example.com."  
  }  
]
```

Untuk melihat atribut alamat IP elastis, Anda harus terlebih dahulu mengaitkan nama domain dengan alamat IP elastis. Untuk informasi selengkapnya, lihat [Menggunakan DNS terbalik untuk aplikasi email](#) di Panduan Pengguna Amazon EC2 [modify-address-attribute](#) atau di Referensi Perintah AWS CLI.

- Untuk detail API, lihat [DescribeAddressesAttribute](#) di Referensi AWS CLI Perintah.

## describe-addresses

Contoh kode berikut menunjukkan cara menggunakan `describe-addresses`.

### AWS CLI

Contoh 1: Untuk mengambil detail tentang semua alamat IP Elastis Anda

Contoh `describe addresses` berikut menampilkan detail tentang alamat IP Elastis Anda.

```
aws ec2 describe-addresses
```

Output:

```
{  
  "Addresses": [  
    {  
      "InstanceId": "i-1234567890abcdef0",  
      "PublicIp": "198.51.100.0",  
      "PublicIpv4Pool": "amazon",  
      "Domain": "standard"  
    },  
    {  
      "Domain": "vpc",  
      "PublicIpv4Pool": "amazon",  
      "InstanceId": "i-1234567890abcdef0",
```

```
    "NetworkInterfaceId": "eni-12345678",
    "AssociationId": "eipassoc-12345678",
    "NetworkInterfaceOwnerId": "123456789012",
    "PublicIp": "203.0.113.0",
    "AllocationId": "eipalloc-12345678",
    "PrivateIpAddress": "10.0.1.241"
  }
]
```

Contoh 2: Guna mengambil detail alamat IP Elastis Anda untuk EC2-VPC

Contoh `describe-addresses` berikut menampilkan detail tentang alamat IP Elastis Anda untuk digunakan dengan instans di VPC.

```
aws ec2 describe-addresses \
  --filters "Name=domain,Values=vpc"
```

Output:

```
{
  "Addresses": [
    {
      "Domain": "vpc",
      "PublicIpv4Pool": "amazon",
      "InstanceId": "i-1234567890abcdef0",
      "NetworkInterfaceId": "eni-12345678",
      "AssociationId": "eipassoc-12345678",
      "NetworkInterfaceOwnerId": "123456789012",
      "PublicIp": "203.0.113.0",
      "AllocationId": "eipalloc-12345678",
      "PrivateIpAddress": "10.0.1.241"
    }
  ]
}
```

Contoh 3: Untuk mengambil detail tentang alamat IP Elastis yang ditentukan oleh ID alokasi

Contoh `describe-addresses` berikut menampilkan detail tentang alamat IP Elastis dengan ID alokasi tertentu, yang dikaitkan dengan instans di EC2-VPC.

```
aws ec2 describe-addresses \
```

```
--allocation-ids eipalloc-282d9641
```

Output:

```
{
  "Addresses": [
    {
      "Domain": "vpc",
      "PublicIpv4Pool": "amazon",
      "InstanceId": "i-1234567890abcdef0",
      "NetworkInterfaceId": "eni-1a2b3c4d",
      "AssociationId": "eipassoc-123abc12",
      "NetworkInterfaceOwnerId": "1234567891012",
      "PublicIp": "203.0.113.25",
      "AllocationId": "eipalloc-282d9641",
      "PrivateIpAddress": "10.251.50.12"
    }
  ]
}
```

Contoh 4: Untuk mengambil detail tentang alamat IP Elastis yang ditentukan oleh alamat IP privat VPC-nya

Contoh `describe-addresses` berikut menampilkan detail tentang alamat IP Elastis yang terkait dengan alamat IP privat tertentu di EC2-VPC.

```
aws ec2 describe-addresses \
  --filters "Name=private-ip-address,Values=10.251.50.12"
```

Contoh 5: Untuk mengambil detail tentang alamat IP Elastis di EC2-Classic

Contoh `describe-addresses` berikut menampilkan detail tentang alamat IP Elastis Anda untuk digunakan di EC2-Classic.

```
aws ec2 describe-addresses \
  --filters "Name=domain,Values=standard"
```

Output:

```
{
  "Addresses": [
```



```

    {
      "InstanceId": "i-1234567890abcdef0",
      "PublicIp": "203.0.110.25",
      "PublicIpv4Pool": "amazon",
      "Domain": "standard"
    }
  ]
}

```

Contoh 6: Untuk mengambil detail tentang alamat IP Elastis yang ditentukan oleh alamat IP publiknya

Contoh `describe-addresses` berikut menampilkan detail tentang alamat IP Elastis dengan nilai `203.0.110.25`, yang dikaitkan dengan instans di EC2-Classic.

```

aws ec2 describe-addresses \
  --public-ips 203.0.110.25

```

Output:

```

{
  "Addresses": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "PublicIp": "203.0.110.25",
      "PublicIpv4Pool": "amazon",
      "Domain": "standard"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeAddresses](#) di Referensi AWS CLI Perintah.

## **describe-aggregate-id-format**

Contoh kode berikut menunjukkan cara menggunakan `describe-aggregate-id-format`.

### AWS CLI

Untuk menjelaskan setelan format ID yang lebih panjang untuk semua jenis sumber daya di Wilayah

`describe-aggregate-id-format` Contoh berikut menjelaskan status format ID panjang keseluruhan untuk Wilayah saat ini. `Deadline` Nilai menunjukkan bahwa tenggat waktu untuk sumber daya ini untuk secara permanen beralih dari format ID pendek ke format ID panjang kedaluwarsa. `UseLongIdsAggregated` Nilai menunjukkan bahwa semua pengguna IAM dan peran IAM dikonfigurasi untuk menggunakan format ID panjang untuk semua jenis sumber daya.

```
aws ec2 describe-aggregate-id-format
```

Output:

```
{
  "UseLongIdsAggregated": true,
  "Statuses": [
    {
      "Deadline": "2018-08-13T02:00:00.000Z",
      "Resource": "network-interface-attachment",
      "UseLongIds": true
    },
    {
      "Deadline": "2016-12-13T02:00:00.000Z",
      "Resource": "instance",
      "UseLongIds": true
    },
    {
      "Deadline": "2018-08-13T02:00:00.000Z",
      "Resource": "elastic-ip-association",
      "UseLongIds": true
    },
    ...
  ]
}
```

- Untuk detail API, lihat [DescribeAggregateIdFormat](#) di Referensi AWS CLI Perintah.

## `describe-availability-zones`

Contoh kode berikut menunjukkan cara menggunakan `describe-availability-zones`.

### AWS CLI

Untuk menjelaskan Zona Ketersediaan Anda

Contoh `describe-availability-zones` berikut menampilkan detail untuk Zona Ketersediaan yang tersedia untuk Anda. Responsnya mencakup Zona Ketersediaan hanya untuk Wilayah saat ini. Dalam contoh ini, respons menggunakan default profil Wilayah `us-west-2` (Oregon).

```
aws ec2 describe-availability-zones
```

Output:

```
{
  "AvailabilityZones": [
    {
      "State": "available",
      "OptInStatus": "opt-in-not-required",
      "Messages": [],
      "RegionName": "us-west-2",
      "ZoneName": "us-west-2a",
      "ZoneId": "usw2-az1",
      "GroupName": "us-west-2",
      "NetworkBorderGroup": "us-west-2"
    },
    {
      "State": "available",
      "OptInStatus": "opt-in-not-required",
      "Messages": [],
      "RegionName": "us-west-2",
      "ZoneName": "us-west-2b",
      "ZoneId": "usw2-az2",
      "GroupName": "us-west-2",
      "NetworkBorderGroup": "us-west-2"
    },
    {
      "State": "available",
      "OptInStatus": "opt-in-not-required",
      "Messages": [],
      "RegionName": "us-west-2",
      "ZoneName": "us-west-2c",
      "ZoneId": "usw2-az3",
      "GroupName": "us-west-2",
      "NetworkBorderGroup": "us-west-2"
    },
    {
      "State": "available",
      "OptInStatus": "opt-in-not-required",
```

```

    "Messages": [],
    "RegionName": "us-west-2",
    "ZoneName": "us-west-2d",
    "ZoneId": "usw2-az4",
    "GroupName": "us-west-2",
    "NetworkBorderGroup": "us-west-2"
  },
  {
    "State": "available",
    "OptInStatus": "opted-in",
    "Messages": [],
    "RegionName": "us-west-2",
    "ZoneName": "us-west-2-lax-1a",
    "ZoneId": "usw2-lax1-az1",
    "GroupName": "us-west-2-lax-1",
    "NetworkBorderGroup": "us-west-2-lax-1"
  }
]
}

```

- Untuk detail API, lihat [DescribeAvailabilityZones](#) di Referensi AWS CLI Perintah.

## describe-aws-network-performance-metric-subscription

Contoh kode berikut menunjukkan cara menggunakan `describe-aws-network-performance-metric-subscription`.

### AWS CLI

Untuk mendeskripsikan langganan metrik Anda

`describe-aws-network-performance-metric-subscriptions` Contoh berikut menjelaskan langganan metrik Anda.

```
aws ec2 describe-aws-network-performance-metric-subscriptions
```

Output:

```
{
  "Subscriptions": [
    {
```

```
        "Source": "us-east-1",
        "Destination": "eu-west-1",
        "Metric": "aggregate-latency",
        "Statistic": "p50",
        "Period": "five-minutes"
    }
]
```

Untuk informasi selengkapnya, lihat [Mengelola langganan](#) di Panduan Pengguna Kinerja Infrastruktur.

- Untuk detail API, lihat [DescribeAwsNetworkPerformanceMetricSubscription](#) di Referensi AWS CLI Perintah.

## **describe-aws-network-performance-metric-subscriptions**

Contoh kode berikut menunjukkan cara menggunakan `describe-aws-network-performance-metric-subscriptions`.

### AWS CLI

Untuk mendeskripsikan langganan metrik Anda

`describe-aws-network-performance-metric-subscriptions` Contoh berikut menjelaskan langganan metrik Anda.

```
aws ec2 describe-aws-network-performance-metric-subscriptions
```

Output:

```
{
  "Subscriptions": [
    {
      "Source": "us-east-1",
      "Destination": "eu-west-1",
      "Metric": "aggregate-latency",
      "Statistic": "p50",
      "Period": "five-minutes"
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengelola langganan](#) di Panduan Pengguna Kinerja Infrastruktur.

- Untuk detail API, lihat [DescribeAwsNetworkPerformanceMetricSubscriptions](#) di Referensi AWS CLI Perintah.

## describe-bundle-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-bundle-tasks`.

### AWS CLI

Untuk menggambarkan tugas bundel Anda

Contoh ini menjelaskan semua tugas bundel Anda.

Perintah:

```
aws ec2 describe-bundle-tasks
```

Output:

```
{
  "BundleTasks": [
    {
      "UpdateTime": "2015-09-15T13:26:54.000Z",
      "InstanceId": "i-1234567890abcdef0",
      "Storage": {
        "S3": {
          "Prefix": "winami",
          "Bucket": "bundletasks"
        }
      },
      "State": "bundling",
      "StartTime": "2015-09-15T13:24:35.000Z",
      "Progress": "3%",
      "BundleId": "bun-2a4e041c"
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [DescribeBundleTasks](#) di Referensi AWS CLI Perintah.

## describe-byoip-cidrs

Contoh kode berikut menunjukkan cara menggunakan `describe-byoip-cidrs`.

### AWS CLI

Untuk mendeskripsikan rentang alamat yang Anda berikan

`describe-byoip-cidrs` Contoh berikut menampilkan detail tentang rentang alamat IPv4 publik yang Anda sediakan untuk digunakan oleh. AWS

```
aws ec2 describe-byoip-cidrs
```

Output:

```
{
  "ByoipCidrs": [
    {
      "Cidr": "203.0.113.25/24",
      "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
      "State": "provisioned"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeByoipCidrs](#) di Referensi AWS CLI Perintah.

## describe-capacity-reservation-fleets

Contoh kode berikut menunjukkan cara menggunakan `describe-capacity-reservation-fleets`.

### AWS CLI

Untuk melihat Armada Reservasi Kapasitas

`describe-capacity-reservation-fleets` Contoh berikut mencantumkan informasi konfigurasi dan kapasitas untuk Armada Reservasi Kapasitas yang ditentukan. Ini juga mencantumkan rincian tentang Reservasi Kapasitas individu yang ada di dalam Armada. :

```
aws ec2 describe-capacity-reservation-fleets \  
  --capacity-reservation-fleet-ids crf-abcdef01234567890
```

Output:

```
{  
  "CapacityReservationFleets": [  
    {  
      "Status": "active",  
      "EndDate": "2022-12-31T23:59:59.000Z",  
      "InstanceMatchCriteria": "open",  
      "Tags": [],  
      "CapacityReservationFleetId": "crf-abcdef01234567890",  
      "Tenancy": "default",  
      "InstanceTypeSpecifications": [  
        {  
          "CapacityReservationId": "cr-1234567890abcdef0",  
          "AvailabilityZone": "us-east-1a",  
          "FulfilledCapacity": 5.0,  
          "Weight": 1.0,  
          "CreateDate": "2022-07-02T08:34:33.398Z",  
          "InstancePlatform": "Linux/UNIX",  
          "TotalInstanceCount": 5,  
          "Priority": 1,  
          "EbsOptimized": true,  
          "InstanceType": "m5.xlarge"  
        }  
      ],  
      "TotalTargetCapacity": 5,  
      "TotalFulfilledCapacity": 5.0,  
      "CreateTime": "2022-07-02T08:34:33.397Z",  
      "AllocationStrategy": "prioritized"  
    }  
  ]  
}
```

Untuk informasi selengkapnya tentang Armada Reservasi [Kapasitas](#), lihat [Armada Reservasi Kapasitas](#) di Panduan Pengguna Amazon EC2.



- Untuk detail API, lihat [DescribeCapacityReservations](#) di Referensi AWS CLI Perintah.

## describe-capacity-reservations

Contoh kode berikut menunjukkan cara menggunakan `describe-capacity-reservations`.

### AWS CLI

Contoh 1: Untuk menggambarkan satu atau lebih reservasi kapasitas Anda

`describe-capacity-reservations` Contoh berikut menampilkan detail tentang semua reservasi kapasitas Anda di AWS Wilayah saat ini.

```
aws ec2 describe-capacity-reservations
```

Output:

```
{
  "CapacityReservations": [
    {
      "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
      "EndDateType": "unlimited",
      "AvailabilityZone": "eu-west-1a",
      "InstanceMatchCriteria": "open",
      "Tags": [],
      "EphemeralStorage": false,
      "CreateDate": "2019-08-16T09:03:18.000Z",
      "AvailableInstanceCount": 1,
      "InstancePlatform": "Linux/UNIX",
      "TotalInstanceCount": 1,
      "State": "active",
      "Tenancy": "default",
      "EbsOptimized": true,
      "InstanceType": "a1.medium"
    },
    {
      "CapacityReservationId": "cr-abcdEXAMPLE9876ef ",
      "EndDateType": "unlimited",
      "AvailabilityZone": "eu-west-1a",
      "InstanceMatchCriteria": "open",
      "Tags": [],
      "EphemeralStorage": false,
```

```

    "CreateDate": "2019-08-07T11:34:19.000Z",
    "AvailableInstanceCount": 3,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 3,
    "State": "cancelled",
    "Tenancy": "default",
    "EbsOptimized": true,
    "InstanceType": "m5.large"
  }
]
}

```

Contoh 2: Untuk menggambarkan satu atau lebih reservasi kapasitas Anda

`describe-capacity-reservations` Contoh berikut menampilkan rincian tentang reservasi kapasitas yang ditentukan.

```

aws ec2 describe-capacity-reservations \
  --capacity-reservation-ids cr-1234abcd56EXAMPLE

```

Output:

```

{
  "CapacityReservations": [
    {
      "CapacityReservationId": "cr-1234abcd56EXAMPLE",
      "EndDateType": "unlimited",
      "AvailabilityZone": "eu-west-1a",
      "InstanceMatchCriteria": "open",
      "Tags": [],
      "EphemeralStorage": false,
      "CreateDate": "2019-08-16T09:03:18.000Z",
      "AvailableInstanceCount": 1,
      "InstancePlatform": "Linux/UNIX",
      "TotalInstanceCount": 1,
      "State": "active",
      "Tenancy": "default",
      "EbsOptimized": true,
      "InstanceType": "a1.medium"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat Reservasi Kapasitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeCapacityReservations](#) di Referensi AWS CLI Perintah.

## describe-carrier-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-carrier-gateways`.

### AWS CLI

Untuk menggambarkan semua gateway operator

`describe-carrier-gateways` Contoh berikut mencantumkan semua gateway operator Anda.

```
aws ec2 describe-carrier-gateways
```

Output:

```
{
  "CarrierGateways": [
    {
      "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
      "VpcId": "vpc-0c529aEXAMPLE",
      "State": "available",
      "OwnerId": "123456789012",
      "Tags": [
        {
          "Key": "example",
          "Value": "tag"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Gateway operator < [https://docs.aws.amazon.com/vpc/latest/userguide/Carrier\\_Gateway.html](https://docs.aws.amazon.com/vpc/latest/userguide/Carrier_Gateway.html) > di Panduan Pengguna Amazon Virtual Private Cloud.

- Untuk detail API, lihat [DescribeCarrierGateways](#) di Referensi AWS CLI Perintah.

## describe-classic-link-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-classic-link-instances`.

### AWS CLI

Untuk mendeskripsikan instance EC2-Classic yang ditautkan

Contoh ini mencantumkan semua instans EC2-Classic yang ditautkan.

Perintah:

```
aws ec2 describe-classic-link-instances
```

Output:

```
{
  "Instances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "VpcId": "vpc-88888888",
      "Groups": [
        {
          "GroupId": "sg-11122233"
        }
      ],
      "Tags": [
        {
          "Value": "ClassicInstance",
          "Key": "Name"
        }
      ]
    },
    {
      "InstanceId": "i-0598c7d356eba48d7",
      "VpcId": "vpc-12312312",
      "Groups": [
        {
          "GroupId": "sg-aabbccdd"
        }
      ],
      "Tags": [
        {
          "Value": "ClassicInstance2",
```

```

    "Key": "Name"
  }
]
}

```

Contoh ini mencantumkan semua instans EC2-Classic yang ditautkan, dan memfilter respons agar hanya menyertakan instance yang ditautkan ke VPC vpc-88888888.

Perintah:

```
aws ec2 describe-classic-link-instances --filter "Name=vpc-id,Values=vpc-88888888"
```

Output:

```

{
  "Instances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "VpcId": "vpc-88888888",
      "Groups": [
        {
          "GroupId": "sg-11122233"
        }
      ],
      "Tags": [
        {
          "Value": "ClassicInstance",
          "Key": "Name"
        }
      ]
    }
  ]
}

```

- Untuk detail API, lihat [DescribeClassicLinkInstances](#) di Referensi AWS CLI Perintah.

## describe-client-vpn-authorization-rules

Contoh kode berikut menunjukkan cara menggunakan `describe-client-vpn-authorization-rules`.

## AWS CLI

Untuk menjelaskan aturan otorisasi untuk titik akhir Client VPN

`describe-client-vpn-authorization-rules` Contoh berikut menampilkan detail tentang aturan otorisasi untuk titik akhir Client VPN yang ditentukan.

```
aws ec2 describe-client-vpn-authorization-rules \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
  "AuthorizationRules": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "GroupId": "",
      "AccessAll": true,
      "DestinationCidr": "0.0.0.0/0",
      "Status": {
        "Code": "active"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Aturan Otorisasi](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DescribeClientVpnAuthorizationRules](#) di Referensi AWS CLI Perintah.

## `describe-client-vpn-connections`

Contoh kode berikut menunjukkan cara menggunakan `describe-client-vpn-connections`.

### AWS CLI

Untuk mendeskripsikan koneksi ke titik akhir Client VPN

`describe-client-vpn-connections` Contoh berikut menampilkan detail tentang koneksi klien ke titik akhir Client VPN yang ditentukan.

```
aws ec2 describe-client-vpn-connections \
```

```
--client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

**Output:**

```
{
  "Connections": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "Timestamp": "2019-08-12 07:58:34",
      "ConnectionId": "cvpn-connection-0e03eb24267165acd",
      "ConnectionEstablishedTime": "2019-08-12 07:57:14",
      "IngressBytes": "32302",
      "EgressBytes": "5696",
      "IngressPackets": "332",
      "EgressPackets": "67",
      "ClientIp": "172.31.0.225",
      "CommonName": "client1.domain.tld",
      "Status": {
        "Code": "terminated"
      },
      "ConnectionEndTime": "2019-08-12 07:58:34"
    },
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "Timestamp": "2019-08-12 08:02:54",
      "ConnectionId": "cvpn-connection-00668867a40f18253",
      "ConnectionEstablishedTime": "2019-08-12 08:02:53",
      "IngressBytes": "2951",
      "EgressBytes": "2611",
      "IngressPackets": "9",
      "EgressPackets": "6",
      "ClientIp": "172.31.0.226",
      "CommonName": "client1.domain.tld",
      "Status": {
        "Code": "active"
      },
      "ConnectionEndTime": "-"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Koneksi Klien](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DescribeClientVpnConnections](#) di Referensi AWS CLI Perintah.

## describe-client-vpn-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-client-vpn-endpoints`.

### AWS CLI

Untuk mendeskripsikan titik akhir Client VPN Anda

`describe-client-vpn-endpoints` Contoh berikut menampilkan detail tentang semua titik akhir Client VPN Anda.

```
aws ec2 describe-client-vpn-endpoints
```

Output:

```
{
  "ClientVpnEndpoints": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "Description": "Endpoint for Admin access",
      "Status": {
        "Code": "available"
      },
      "CreationTime": "2020-11-13T11:37:27",
      "DnsName": "*.cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-
south-1.amazonaws.com",
      "ClientCidrBlock": "172.31.0.0/16",
      "DnsServers": [
        "8.8.8.8"
      ],
      "SplitTunnel": false,
      "VpnProtocol": "openvpn",
      "TransportProtocol": "udp",
      "VpnPort": 443,
      "ServerCertificateArn": "arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "AuthenticationOptions": [
        {
          "Type": "certificate-authentication",
          "MutualAuthentication": {
            "ClientRootCertificateChain": "arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
          }
        }
      ]
    }
  ]
}
```



```

        }
    ],
    "ConnectionLogOptions": {
        "Enabled": true,
        "CloudwatchLogGroup": "Client-vpn-connection-logs",
        "CloudwatchLogStream": "cvpn-endpoint-123456789123abcde-ap-
south-1-2020/11/13-FCD8HEMvaCcw"
    },
    "Tags": [
        {
            "Key": "Name",
            "Value": "Client VPN"
        }
    ],
    "SecurityGroupIds": [
        "sg-aabbcc11223344567"
    ],
    "VpcId": "vpc-a87f92c1",
    "SelfServicePortalUrl": "https://self-service.clientvpn.amazonaws.com/
endpoints/cvpn-endpoint-123456789123abcde",
    "ClientConnectOptions": {
        "Enabled": false
    }
}
]
}

```

Untuk informasi selengkapnya, lihat [Titik Akhir Client VPN](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DescribeClientVpnEndpoints](#) di Referensi AWS CLI Perintah.

## describe-client-vpn-routes

Contoh kode berikut menunjukkan cara menggunakan `describe-client-vpn-routes`.

### AWS CLI

Untuk mendeskripsikan rute untuk titik akhir Client VPN

`describe-client-vpn-routes` Contoh berikut menampilkan detail tentang rute untuk titik akhir Client VPN yang ditentukan.

```
aws ec2 describe-client-vpn-routes \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

### Output:

```
{
  "Routes": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "DestinationCidr": "10.0.0.0/16",
      "TargetSubnet": "subnet-0123456789abcabca",
      "Type": "Nat",
      "Origin": "associate",
      "Status": {
        "Code": "active"
      },
      "Description": "Default Route"
    },
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "DestinationCidr": "0.0.0.0/0",
      "TargetSubnet": "subnet-0123456789abcabca",
      "Type": "Nat",
      "Origin": "add-route",
      "Status": {
        "Code": "active"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Route](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DescribeClientVpnRoutes](#) di Referensi AWS CLI Perintah.

## **describe-client-vpn-target-networks**

Contoh kode berikut menunjukkan cara menggunakan `describe-client-vpn-target-networks`.

## AWS CLI

Untuk mendeskripsikan jaringan target untuk titik akhir Client VPN

`describe-client-vpn-target-networks` Contoh berikut menampilkan detail tentang jaringan target untuk titik akhir Client VPN yang ditentukan.

```
aws ec2 describe-client-vpn-target-networks \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{  
  "ClientVpnTargetNetworks": [  
    {  
      "AssociationId": "cvpn-assoc-012e837060753dc3d",  
      "VpcId": "vpc-11111222222333333",  
      "TargetNetworkId": "subnet-0123456789abcabca",  
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",  
      "Status": {  
        "Code": "associating"  
      },  
      "SecurityGroups": [  
        "sg-012345678910abcab"  
      ]  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Jaringan Target](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DescribeClientVpnTargetNetworks](#) di Referensi AWS CLI Perintah.

## **describe-coip-pools**

Contoh kode berikut menunjukkan cara menggunakan `describe-coip-pools`.

## AWS CLI

Untuk menggambarkan kumpulan alamat IP milik pelanggan

`describe-coip-pools` Contoh berikut menjelaskan kumpulan alamat IP milik pelanggan di akun Anda AWS .

```
aws ec2 describe-coip-pools
```

Output:

```
{
  "CoipPools": [
    {
      "PoolId": "ipv4pool-coip-123a45678bEXAMPLE",
      "PoolCidrs": [
        "0.0.0.0/0"
      ],
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
      "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-coip-123a45678bEXAMPLE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Alamat IP milik pelanggan](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [DescribeCoipPools](#) di Referensi AWS CLI Perintah.

## describe-conversion-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-conversion-tasks`.

AWS CLI

Untuk melihat status tugas konversi

Contoh ini mengembalikan status tugas konversi dengan ID `import-i-ffvko9js`.

Perintah:

```
aws ec2 describe-conversion-tasks --conversion-task-ids import-i-ffvko9js
```

Output:

```
{
  "ConversionTasks": [
```

```

    {
      "ConversionTaskId": "import-i-ffvko9js",
      "ImportInstance": {
        "InstanceId": "i-1234567890abcdef0",
        "Volumes": [
          {
            "Volume": {
              "Id": "vol-049df61146c4d7901",
              "Size": 16
            },
            "Status": "completed",
            "Image": {
              "Size": 1300687360,
              "ImportManifestUrl": "https://s3.amazonaws.com/myimportbucket/411443cd-d620-4f1c-9d66-13144EXAMPLE/RHEL5.vmdkmanifest.xml?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Expires=140EXAMPLE&Signature=XYNhznHNgCqsjDxL9wRL%2FJvEXAMPLE",
              "Format": "VMDK"
            },
            "BytesConverted": 1300682960,
            "AvailabilityZone": "us-east-1d"
          }
        ]
      },
      "ExpirationTime": "2014-05-14T22:06:23Z",
      "State": "completed"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeConversionTasks](#) di Referensi AWS CLI Perintah.

## describe-customer-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-customer-gateways`.

### AWS CLI

Untuk mendeskripsikan gateway pelanggan Anda

Contoh ini menjelaskan gateway pelanggan Anda.

Perintah:

```
aws ec2 describe-customer-gateways
```

Output:

```
{
  "CustomerGateways": [
    {
      "CustomerGatewayId": "cgw-b4dc3961",
      "IpAddress": "203.0.113.12",
      "State": "available",
      "Type": "ipsec.1",
      "BgpAsn": "65000"
    },
    {
      "CustomerGatewayId": "cgw-0e11f167",
      "IpAddress": "12.1.2.3",
      "State": "available",
      "Type": "ipsec.1",
      "BgpAsn": "65534"
    }
  ]
}
```

Untuk menggambarkan gateway pelanggan tertentu

Contoh ini menjelaskan gateway pelanggan yang ditentukan.

Perintah:

```
aws ec2 describe-customer-gateways --customer-gateway-ids cgw-0e11f167
```

Output:

```
{
  "CustomerGateways": [
    {
      "CustomerGatewayId": "cgw-0e11f167",
      "IpAddress": "12.1.2.3",
      "State": "available",
      "Type": "ipsec.1",
      "BgpAsn": "65534"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [DescribeCustomerGateways](#) di Referensi AWS CLI Perintah.

## describe-dhcp-options

Contoh kode berikut menunjukkan cara menggunakan `describe-dhcp-options`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan opsi DHCP Anda

`describe-dhcp-options` Contoh berikut mengambil rincian tentang opsi DHCP Anda.

```
aws ec2 describe-dhcp-options
```

Output:

```
{
  "DhcpOptions": [
    {
      "DhcpConfigurations": [
        {
          "Key": "domain-name",
          "Values": [
            {
              "Value": "us-east-2.compute.internal"
            }
          ]
        },
        {
          "Key": "domain-name-servers",
          "Values": [
            {
              "Value": "AmazonProvidedDNS"
            }
          ]
        }
      ],
      "DhcpOptionsId": "dopt-19edf471",
      "OwnerId": "111122223333"
    },
  ],
}
```

```

    {
      "DhcpConfigurations": [
        {
          "Key": "domain-name",
          "Values": [
            {
              "Value": "us-east-2.compute.internal"
            }
          ]
        },
        {
          "Key": "domain-name-servers",
          "Values": [
            {
              "Value": "AmazonProvidedDNS"
            }
          ]
        }
      ],
      "DhcpOptionsId": "dopt-fEXAMPLE",
      "OwnerId": "111122223333"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Set Opsi DHCP](#) di Panduan AWS Pengguna VPC.

Contoh 2: Untuk menjelaskan opsi DHCP Anda dan memfilter output

`describe-dhcp-options` Contoh berikut menjelaskan opsi DHCP Anda dan menggunakan filter untuk mengembalikan hanya opsi DHCP yang dimiliki `example.com` untuk server nama domain. Contoh menggunakan `--query` parameter untuk menampilkan hanya informasi konfigurasi dan ID dalam output.

```

aws ec2 describe-dhcp-options \
  --filters Name=key,Values=domain-name-servers Name=value,Values=example.com \
  --query "DhcpOptions[*].[DhcpConfigurations,DhcpOptionsId]"

```

Output:

```
[
```



```
[
  [
    {
      "Key": "domain-name",
      "Values": [
        {
          "Value": "example.com"
        }
      ]
    },
    {
      "Key": "domain-name-servers",
      "Values": [
        {
          "Value": "172.16.16.16"
        }
      ]
    }
  ],
  "dopt-001122334455667ab"
]
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Set Opsi DHCP](#) di Panduan AWS Pengguna VPC.

- Untuk detail API, lihat [DescribeDhcpOptions](#) di Referensi AWS CLI Perintah.

## **describe-egress-only-internet-gateways**

Contoh kode berikut menunjukkan cara menggunakan `describe-egress-only-internet-gateways`.

### AWS CLI

Untuk menggambarkan gateway Internet khusus egres Anda

Contoh ini menjelaskan gateway Internet khusus egres Anda.

Perintah:

```
aws ec2 describe-egress-only-internet-gateways
```

**Output:**

```
{
  "EgressOnlyInternetGateways": [
    {
      "EgressOnlyInternetGatewayId": "eigw-015e0e244e24dfe8a",
      "Attachments": [
        {
          "State": "attached",
          "VpcId": "vpc-0c62a468"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeEgressOnlyInternetGateways](#) di Referensi AWS CLI Perintah.

**describe-elastic-gpus**

Contoh kode berikut menunjukkan cara menggunakan `describe-elastic-gpus`.

**AWS CLI**

Untuk menggambarkan GPU Elastis

Perintah:

```
aws ec2 describe-elastic-gpus --elastic-gpu-ids
egpu-12345678901234567890abcdefghijkl
```

- Untuk detail API, lihat [DescribeElasticGpus](#) di Referensi AWS CLI Perintah.

**describe-export-image-tasks**

Contoh kode berikut menunjukkan cara menggunakan `describe-export-image-tasks`.

**AWS CLI**

Untuk memantau tugas ekspor gambar

`describe-export-image-tasks` Contoh berikut memeriksa status tugas gambar ekspor yang ditentukan. File gambar yang dihasilkan di Amazon S3 adalah `my-export-bucket/exports/export-ami-1234567890abcdef0.vmdk`

```
aws ec2 describe-export-image-tasks \
  --export-image-task-ids export-ami-1234567890abcdef0
```

Output untuk tugas ekspor gambar yang sedang berlangsung.

```
{
  "ExportImageTasks": [
    {
      "ExportImageTaskId": "export-ami-1234567890abcdef0"
      "Progress": "21",
      "S3ExportLocation": {
        "S3Bucket": "my-export-bucket",
        "S3Prefix": "exports/"
      },
      "Status": "active",
      "StatusMessage": "updating"
    }
  ]
}
```

Output untuk tugas ekspor gambar yang selesai.

```
{
  "ExportImageTasks": [
    {
      "ExportImageTaskId": "export-ami-1234567890abcdef0"
      "S3ExportLocation": {
        "S3Bucket": "my-export-bucket",
        "S3Prefix": "exports/"
      },
      "Status": "completed"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengekspor VM dari AMI di Panduan Pengguna Impor/ Ekspor VM](#).

- Untuk detail API, lihat [DescribeExportImageTasks](#) di Referensi AWS CLI Perintah.

## describe-export-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-export-tasks`.

### AWS CLI

Untuk membuat daftar detail tentang tugas ekspor instance

Contoh ini menjelaskan tugas ekspor dengan ID `export-i-fh8sjjsq`.

Perintah:

```
aws ec2 describe-export-tasks --export-task-ids export-i-fh8sjjsq
```

Output:

```
{
  "ExportTasks": [
    {
      "State": "active",
      "InstanceExportDetails": {
        "InstanceId": "i-1234567890abcdef0",
        "TargetEnvironment": "vmware"
      },
      "ExportToS3Task": {
        "S3Bucket": "myexportbucket",
        "S3Key": "RHEL5export-i-fh8sjjsq.ova",
        "DiskImageFormat": "vmdk",
        "ContainerFormat": "ova"
      },
      "Description": "RHEL5 instance",
      "ExportTaskId": "export-i-fh8sjjsq"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeExportTasks](#) di Referensi AWS CLI Perintah.

## describe-fast-launch-images

Contoh kode berikut menunjukkan cara menggunakan `describe-fast-launch-images`.

### AWS CLI

Untuk menjelaskan detail untuk AMI Windows yang dikonfigurasi untuk peluncuran lebih cepat `describe-fast-launch-images` Contoh berikut menjelaskan detail untuk setiap AMI di akun Anda yang dikonfigurasi untuk peluncuran lebih cepat, termasuk jenis sumber daya, konfigurasi snapshot, detail template peluncuran, jumlah maksimum peluncuran paralel, ID pemilik AMI, status konfigurasi peluncuran cepat, alasan status diubah, dan waktu terjadinya perubahan status.

```
aws ec2 describe-fast-launch-images
```

### Output:

```
{
  "FastLaunchImages": [
    {
      "ImageId": "ami-01234567890abcdef",
      "ResourceType": "snapshot",
      "SnapshotConfiguration": {},
      "LaunchTemplate": {
        "LaunchTemplateId": "lt-01234567890abcdef",
        "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
        "Version": "1"
      },
      "MaxParallelLaunches": 6,
      "OwnerId": "0123456789123",
      "State": "enabled",
      "StateTransitionReason": "Client.UserInitiated",
      "StateTransitionTime": "2022-01-27T22:20:06.552000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya tentang mengonfigurasi AMI Windows untuk peluncuran yang lebih cepat, lihat [Mengonfigurasi AMI Anda untuk peluncuran lebih cepat](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeFastLaunchImages](#) di Referensi AWS CLI Perintah.

## describe-fast-snapshot-restores

Contoh kode berikut menunjukkan cara menggunakan `describe-fast-snapshot-restores`.

### AWS CLI

Untuk menggambarkan pemulihan snapshot cepat

`describe-fast-snapshot-restores` Contoh berikut menampilkan detail untuk semua pemulihan snapshot cepat dengan status `disabled`

```
aws ec2 describe-fast-snapshot-restores \
  --filters Name=state,Values=disabled
```

Output:

```
{
  "FastSnapshotRestores": [
    {
      "SnapshotId": "snap-1234567890abcdef0",
      "AvailabilityZone": "us-west-2c",
      "State": "disabled",
      "StateTransitionReason": "Client.UserInitiated - Lifecycle state
transition",
      "OwnerId": "123456789012",
      "EnablingTime": "2020-01-25T23:57:49.596Z",
      "OptimizingTime": "2020-01-25T23:58:25.573Z",
      "EnabledTime": "2020-01-25T23:59:29.852Z",
      "DisablingTime": "2020-01-26T00:40:56.069Z",
      "DisabledTime": "2020-01-26T00:41:27.390Z"
    }
  ]
}
```

`describe-fast-snapshot-restores` Contoh berikut menjelaskan semua pemulihan snapshot cepat.

```
aws ec2 describe-fast-snapshot-restores
```

- Untuk detail API, lihat [DescribeFastSnapshotRestores](#) di Referensi AWS CLI Perintah.

## describe-fleet-history

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-history`.

### AWS CLI

Untuk menggambarkan sejarah Armada EC2

`describe-fleet-history` Contoh berikut mengembalikan riwayat untuk Armada EC2 yang ditentukan mulai pada waktu yang ditentukan. Outputnya adalah untuk Armada EC2 dengan dua instans yang berjalan.

```
aws ec2 describe-fleet-history \
  --fleet-id fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
  --start-time 2020-09-01T00:00:00Z
```

Output:

```
{
  "HistoryRecords": [
    {
      "EventInformation": {
        "EventSubType": "submitted"
      },
      "EventType": "fleetRequestChange",
      "Timestamp": "2020-09-01T18:26:05.000Z"
    },
    {
      "EventInformation": {
        "EventSubType": "active"
      },
      "EventType": "fleetRequestChange",
      "Timestamp": "2020-09-01T18:26:15.000Z"
    },
    {
      "EventInformation": {
        "EventDescription": "t2.small, ami-07c8bc5c1ce9598c3, ...",
        "EventSubType": "progress"
      },
      "EventType": "fleetRequestChange",
      "Timestamp": "2020-09-01T18:26:17.000Z"
    },
    {
```

```

    "EventInformation": {
      "EventDescription": "{\"instanceType\": \"t2.small\", ...}\",
      "EventSubType": "launched",
      "InstanceId": "i-083a1c446e66085d2"
    },
    "EventType": "instanceChange",
    "Timestamp": "2020-09-01T18:26:17.000Z"
  },
  {
    "EventInformation": {
      "EventDescription": "{\"instanceType\": \"t2.small\", ...}\",
      "EventSubType": "launched",
      "InstanceId": "i-090db02406cc3c2d6"
    },
    "EventType": "instanceChange",
    "Timestamp": "2020-09-01T18:26:17.000Z"
  }
],
"LastEvaluatedTime": "2020-09-01T19:10:19.000Z",
"FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
"StartTime": "2020-08-31T23:53:20.000Z"
}

```

Untuk informasi selengkapnya, lihat [Mengelola Armada EC2](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeFleetHistory](#) di Referensi AWS CLI Perintah.

## describe-fleet-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-instances`.

### AWS CLI

Untuk menjelaskan instans yang sedang berjalan untuk Armada EC2

`describe-fleet-instances` Contoh berikut menjelaskan instance yang sedang berjalan untuk Armada EC2 yang ditentukan.

```
aws ec2 describe-fleet-instances \
  --fleet-id 12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE
```

Output:



```
{
  "ActiveInstances": [
    {
      "InstanceId": "i-090db02406cc3c2d6",
      "InstanceType": "t2.small",
      "SpotInstanceRequestId": "sir-a43gtpfk",
      "InstanceHealth": "healthy"
    },
    {
      "InstanceId": "i-083a1c446e66085d2",
      "InstanceType": "t2.small",
      "SpotInstanceRequestId": "sir-iwcit2nj",
      "InstanceHealth": "healthy"
    }
  ],
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Mengelola Armada EC2](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeFleetInstances](#) di Referensi AWS CLI Perintah.

## describe-fleets

Contoh kode berikut menunjukkan cara menggunakan `describe-fleets`.

### AWS CLI

Untuk menggambarkan Armada EC2

`describe-fleets` Contoh berikut menjelaskan Armada EC2 yang ditentukan.

```
aws ec2 describe-fleets \
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE
```

### Output:

```
{
  "Fleets": [
    {
      "ActivityStatus": "pending_fulfillment",
```

```

"CreateTime": "2020-09-01T18:26:05.000Z",
"FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
"FleetState": "active",
"ExcessCapacityTerminationPolicy": "termination",
"FulfilledCapacity": 0.0,
"FulfilledOnDemandCapacity": 0.0,
"LaunchTemplateConfigs": [
  {
    "LaunchTemplateSpecification": {
      "LaunchTemplateId": "lt-0e632f2855a979cd5",
      "Version": "1"
    }
  }
],
"TargetCapacitySpecification": {
  "TotalTargetCapacity": 2,
  "OnDemandTargetCapacity": 0,
  "SpotTargetCapacity": 2,
  "DefaultTargetCapacityType": "spot"
},
"TerminateInstancesWithExpiration": false,
"Type": "maintain",
"ReplaceUnhealthyInstances": false,
"SpotOptions": {
  "AllocationStrategy": "lowestPrice",
  "InstanceInterruptionBehavior": "terminate",
  "InstancePoolsToUseCount": 1
},
"OnDemandOptions": {
  "AllocationStrategy": "lowestPrice"
}
}
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola Armada EC2](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeFleets](#) di Referensi AWS CLI Perintah.

## describe-flow-logs

Contoh kode berikut menunjukkan cara menggunakan `describe-flow-logs`.

## AWS CLI

Contoh 1: Untuk mendeskripsikan semua flow log

`describe-flow-logs` Contoh berikut menampilkan detail untuk semua log aliran Anda.

```
aws ec2 describe-flow-logs
```

Output:

```
{
  "FlowLogs": [
    {
      "CreationTime": "2018-02-21T13:22:12.644Z",
      "DeliverLogsPermissionArn": "arn:aws:iam::123456789012:role/flow-logs-
role",
      "DeliverLogsStatus": "SUCCESS",
      "FlowLogId": "fl-aabbccdd112233445",
      "MaxAggregationInterval": 600,
      "FlowLogStatus": "ACTIVE",
      "LogGroupName": "FlowLogGroup",
      "ResourceId": "subnet-12345678901234567",
      "TrafficType": "ALL",
      "LogDestinationType": "cloud-watch-logs",
      "LogFormat": "${version} ${account-id} ${interface-id} ${srcaddr}
${dstaddr} ${srcport} ${dstport} ${protocol} ${packets} ${bytes} ${start} ${end}
${action} ${log-status}"
    },
    {
      "CreationTime": "2020-02-04T15:22:29.986Z",
      "DeliverLogsStatus": "SUCCESS",
      "FlowLogId": "fl-01234567890123456",
      "MaxAggregationInterval": 60,
      "FlowLogStatus": "ACTIVE",
      "ResourceId": "vpc-00112233445566778",
      "TrafficType": "ACCEPT",
      "LogDestinationType": "s3",
      "LogDestination": "arn:aws:s3::my-flow-log-bucket/custom",
      "LogFormat": "${version} ${vpc-id} ${subnet-id} ${instance-id}
${interface-id} ${account-id} ${type} ${srcaddr} ${dstaddr} ${srcport} ${dstport}
${pkt-srcaddr} ${pkt-dstaddr} ${protocol} ${bytes} ${packets} ${start} ${end}
${action} ${tcp-flags} ${log-status}"
    }
  ]
}
```

```
]
}
```

Contoh 2: Untuk mendeskripsikan subset dari flow log

`describe-flow-logs` Contoh berikut menggunakan filter untuk menampilkan detail hanya untuk log aliran yang ada di grup log yang ditentukan di Amazon CloudWatch Logs.

```
aws ec2 describe-flow-logs \
  --filter "Name=log-group-name,Values=MyFlowLogs"
```

- Untuk detail API, lihat [DescribeFlowLogs](#) di Referensi AWS CLI Perintah.

## **describe-fpga-image-attribute**

Contoh kode berikut menunjukkan cara menggunakan `describe-fpga-image-attribute`.

AWS CLI

Untuk menggambarkan atribut gambar Amazon FPGA

Contoh ini menjelaskan izin pemuatan untuk AFI yang ditentukan.

Perintah:

```
aws ec2 describe-fpga-image-attribute --fpga-image-id afi-0d123e123bfc85abc --
attribute loadPermission
```

Output:

```
{
  "FpgaImageAttribute": {
    "FpgaImageId": "afi-0d123e123bfc85abc",
    "LoadPermissions": [
      {
        "UserId": "123456789012"
      }
    ]
  }
}
```

- Untuk detail API, lihat [DescribeFpgaImageAttribute](#) di Referensi AWS CLI Perintah.

## describe-fpga-images

Contoh kode berikut menunjukkan cara menggunakan `describe-fpga-images`.

### AWS CLI

Untuk menggambarkan gambar Amazon FPGA

Contoh ini menjelaskan AFI yang dimiliki oleh akun123456789012.

Perintah:

```
aws ec2 describe-fpga-images --filters Name=owner-id,Values=123456789012
```

Output:

```
{
  "FpgaImages": [
    {
      "UpdateTime": "2017-12-22T12:09:14.000Z",
      "Name": "my-afi",
      "PciId": {
        "SubsystemVendorId": "0xfedd",
        "VendorId": "0x1d0f",
        "DeviceId": "0xf000",
        "SubsystemId": "0x1d51"
      },
      "FpgaImageGlobalId": "agfi-123cb27b5e84a0abc",
      "Public": false,
      "State": {
        "Code": "available"
      },
      "ShellVersion": "0x071417d3",
      "OwnerId": "123456789012",
      "FpgaImageId": "afi-0d123e123bfc85abc",
      "CreateTime": "2017-12-22T11:43:33.000Z",
      "Description": "my-afi"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeFpgaImages](#) di Referensi AWS CLI Perintah.

## describe-host-reservation-offerings

Contoh kode berikut menunjukkan cara menggunakan `describe-host-reservation-offerings`.

### AWS CLI

Untuk menjelaskan penawaran Reservasi Tuan Rumah Khusus

Contoh ini menjelaskan Reservasi Tuan Rumah Khusus untuk keluarga instans M4 yang tersedia untuk dibeli.

Perintah:

```
aws ec2 describe-host-reservation-offerings --filter Name=instance-family,Values=m4
```

Output:

```
{
  "OfferingSet": [
    {
      "HourlyPrice": "1.499",
      "OfferingId": "hro-03f707bf363b6b324",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "UpfrontPrice": "0.000",
      "Duration": 31536000
    },
    {
      "HourlyPrice": "1.045",
      "OfferingId": "hro-0ef9181cabdef7a02",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "UpfrontPrice": "0.000",
      "Duration": 94608000
    },
    {
      "HourlyPrice": "0.714",
      "OfferingId": "hro-04567a15500b92a51",
      "InstanceFamily": "m4",
      "PaymentOption": "PartialUpfront",
      "UpfrontPrice": "6254.000",
      "Duration": 31536000
    }
  ]
}
```

```
    },
    {
      "HourlyPrice": "0.484",
      "OfferingId": "hro-0d5d7a9d23ed7fbfe",
      "InstanceFamily": "m4",
      "PaymentOption": "PartialUpfront",
      "UpfrontPrice": "12720.000",
      "Duration": 94608000
    },
    {
      "HourlyPrice": "0.000",
      "OfferingId": "hro-05da4108ca998c2e5",
      "InstanceFamily": "m4",
      "PaymentOption": "AllUpfront",
      "UpfrontPrice": "23913.000",
      "Duration": 94608000
    },
    {
      "HourlyPrice": "0.000",
      "OfferingId": "hro-0a9f9be3b95a3dc8f",
      "InstanceFamily": "m4",
      "PaymentOption": "AllUpfront",
      "UpfrontPrice": "12257.000",
      "Duration": 31536000
    }
  ]
}
```

- Untuk detail API, lihat [DescribeHostReservationOfferings](#) di Referensi AWS CLI Perintah.

## describe-host-reservations

Contoh kode berikut menunjukkan cara menggunakan `describe-host-reservations`.

### AWS CLI

Untuk menjelaskan Reservasi Tuan Rumah Khusus di akun Anda

Contoh ini menjelaskan Reservasi Tuan Rumah Khusus di akun Anda.

Perintah:

```
aws ec2 describe-host-reservations
```

**Output:**

```
{
  "HostReservationSet": [
    {
      "Count": 1,
      "End": "2019-01-10T12:14:09Z",
      "HourlyPrice": "1.499",
      "InstanceFamily": "m4",
      "OfferingId": "hro-03f707bf363b6b324",
      "PaymentOption": "NoUpfront",
      "State": "active",
      "HostIdSet": [
        "h-013abcd2a00cbd123"
      ],
      "Start": "2018-01-10T12:14:09Z",
      "HostReservationId": "hr-0d418a3a4ffc669ae",
      "UpfrontPrice": "0.000",
      "Duration": 31536000
    }
  ]
}
```

- Untuk detail API, lihat [DescribeHostReservations](#) di Referensi AWS CLI Perintah.

**describe-hosts**

Contoh kode berikut menunjukkan cara menggunakan `describe-hosts`.

**AWS CLI**

Untuk melihat detail tentang Host Khusus

`describe-hosts` Contoh berikut menampilkan detail untuk Host available Khusus di AWS akun Anda.

```
aws ec2 describe-hosts --filter "Name=state,Values=available"
```

**Output:**

```
{
  "Hosts": [
```



```
{
  "HostId": "h-07879acf49EXAMPLE",
  "Tags": [
    {
      "Value": "production",
      "Key": "purpose"
    }
  ],
  "HostProperties": {
    "Cores": 48,
    "TotalVCpus": 96,
    "InstanceType": "m5.large",
    "Sockets": 2
  },
  "Instances": [],
  "State": "available",
  "AvailabilityZone": "eu-west-1a",
  "AvailableCapacity": {
    "AvailableInstanceCapacity": [
      {
        "AvailableCapacity": 48,
        "InstanceType": "m5.large",
        "TotalCapacity": 48
      }
    ],
    "AvailableVCpus": 96
  },
  "HostRecovery": "on",
  "AllocationTime": "2019-08-19T08:57:44.000Z",
  "AutoPlacement": "off"
}
]
```

Untuk informasi selengkapnya, lihat [Melihat Host Khusus](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeHosts](#) di Referensi AWS CLI Perintah.

## **describe-iam-instance-profile-associations**

Contoh kode berikut menunjukkan cara menggunakan `describe-iam-instance-profile-associations`.

## AWS CLI

Untuk menjelaskan kaitan profil instans IAM

Contoh ini menjelaskan semua kaitan profil instans IAM Anda.

Perintah:

```
aws ec2 describe-iam-instance-profile-associations
```

Output:

```
{
  "IamInstanceProfileAssociations": [
    {
      "InstanceId": "i-09eb09efa73ec1dee",
      "State": "associated",
      "AssociationId": "iip-assoc-0db249b1f25fa24b8",
      "IamInstanceProfile": {
        "Id": "AIPAJVQN4F5WVLGCJDRGM",
        "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
      }
    },
    {
      "InstanceId": "i-0402909a2f4dff14",
      "State": "associating",
      "AssociationId": "iip-assoc-0d1ec06278d29f44a",
      "IamInstanceProfile": {
        "Id": "AGJAJVQN4F5WVLGCJABCM",
        "Arn": "arn:aws:iam::123456789012:instance-profile/user1-role"
      }
    }
  ]
}
```

- Untuk detail API, lihat [DescribeIamInstanceProfileAssociations](#) di Referensi AWS CLI Perintah.

### describe-id-format

Contoh kode berikut menunjukkan cara menggunakan `describe-id-format`.

## AWS CLI

Contoh 1: Untuk mendeskripsikan format ID sumber daya

`describe-id-format` Contoh berikut menjelaskan format ID untuk grup keamanan.

```
aws ec2 describe-id-format \  
  --resource security-group
```

Dalam contoh output berikut, `Deadline` nilai menunjukkan bahwa batas waktu untuk jenis sumber daya ini untuk secara permanen beralih dari format ID pendek ke format ID panjang berakhir pada 00:00 UTC pada 15 Agustus 2018.

```
{  
  "Statuses": [  
    {  
      "Deadline": "2018-08-15T00:00:00.000Z",  
      "Resource": "security-group",  
      "UseLongIds": true  
    }  
  ]  
}
```

Contoh 2: Untuk mendeskripsikan format ID untuk semua sumber daya

`describe-id-format` Contoh berikut menjelaskan format ID untuk semua jenis sumber daya. Semua jenis sumber daya yang mendukung format ID pendek dialihkan untuk menggunakan format ID panjang.

```
aws ec2 describe-id-format
```

- Untuk detail API, lihat [DescribeIdFormat](#) di Referensi AWS CLI Perintah.

## **describe-identity-id-format**

Contoh kode berikut menunjukkan cara menggunakan `describe-identity-id-format`.

## AWS CLI

Untuk mendeskripsikan format ID untuk peran IAM

`describe-identity-id-format` Contoh berikut menjelaskan format ID yang diterima oleh instance yang dibuat oleh peran IAM `EC2Role` di akun Anda AWS .

```
aws ec2 describe-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:role/my-iam-role \  
  --resource instance
```

Output berikut menunjukkan bahwa instance yang dibuat oleh peran ini menerima ID dalam format ID panjang.

```
{  
  "Statuses": [  
    {  
      "Deadline": "2016-12-15T00:00:00Z",  
      "Resource": "instance",  
      "UseLongIds": true  
    }  
  ]  
}
```

Untuk mendeskripsikan format ID untuk pengguna IAM

`describe-identity-id-format` Contoh berikut menjelaskan format ID yang diterima oleh snapshot yang dibuat oleh pengguna IAM `AdminUser` di akun Anda AWS .

```
aws ec2 describe-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:user/AdminUser \  
  --resource snapshot
```

Output menunjukkan bahwa snapshot yang dibuat oleh pengguna ini menerima ID dalam format ID panjang.

```
{  
  "Statuses": [  
    {  
      "Deadline": "2016-12-15T00:00:00Z",  
      "Resource": "snapshot",  
      "UseLongIds": true  
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeIdentityIdFormat](#) di Referensi AWS CLI Perintah.

## describe-image-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-image-attribute`.

### AWS CLI

Untuk menjelaskan izin peluncuran untuk AMI

Contoh ini menjelaskan izin peluncuran untuk AMI yang ditentukan.

Perintah:

```
aws ec2 describe-image-attribute --image-id ami-5731123e --attribute
  launchPermission
```

Output:

```
{
  "LaunchPermissions": [
    {
      "UserId": "123456789012"
    }
  ],
  "ImageId": "ami-5731123e",
}
```

Untuk mendeskripsikan kode produk untuk AMI

Contoh ini menjelaskan kode produk untuk AMI yang ditentukan. Perhatikan bahwa AMI ini tidak memiliki kode produk.

Perintah:

```
aws ec2 describe-image-attribute --image-id ami-5731123e --attribute productCodes
```

Output:

```
{
  "ProductCodes": [],
}
```

```
"ImageId": "ami-5731123e",
}
```

- Untuk detail API, lihat [DescribeImageAttribute](#) di Referensi AWS CLI Perintah.

## describe-images

Contoh kode berikut menunjukkan cara menggunakan `describe-images`.

### AWS CLI

Contoh 1: Untuk menjelaskan AMI

Contoh `describe-images` berikut menjelaskan AMI tertentu di Wilayah tertentu.

```
aws ec2 describe-images \
  --region us-east-1 \
  --image-ids ami-1234567890EXAMPLE
```

Output:

```
{
  "Images": [
    {
      "VirtualizationType": "hvm",
      "Description": "Provided by Red Hat, Inc.",
      "PlatformDetails": "Red Hat Enterprise Linux",
      "EnaSupport": true,
      "Hypervisor": "xen",
      "State": "available",
      "SriovNetSupport": "simple",
      "ImageId": "ami-1234567890EXAMPLE",
      "UsageOperation": "RunInstances:0010",
      "BlockDeviceMappings": [
        {
          "DeviceName": "/dev/sda1",
          "Ebs": {
            "SnapshotId": "snap-111222333444aaabb",
            "DeleteOnTermination": true,
            "VolumeType": "gp2",
            "VolumeSize": 10,
            "Encrypted": false
          }
        }
      ]
    }
  ]
}
```

```

        }
      }
    ],
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": true,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Amazon Machine Image \(AMI\)](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menjelaskan AMI berdasarkan filter

Contoh `describe-images` berikut menjelaskan AMI Windows yang disediakan oleh Amazon yang didukung oleh Amazon EBS.

```

aws ec2 describe-images \
  --owners amazon \
  --filters "Name=platform,Values=windows" "Name=root-device-type,Values=ebs"

```

Untuk contoh output `describe-images`, lihat Contoh 1.

Untuk contoh tambahan menggunakan filter, lihat [Mencantumkan dan memfilter sumber daya Anda](#) di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk menjelaskan AMI berdasarkan tanda

Contoh `describe-images` berikut menjelaskan semua AMI yang memiliki tanda `Type=Custom`. Contoh tersebut menggunakan parameter `--query` untuk hanya menampilkan ID AMI.

```

aws ec2 describe-images \
  --filters "Name=tag:Type,Values=Custom" \
  --query 'Images[*].[ImageId]' \

```

```
--output text
```

### Output:

```
ami-1234567890EXAMPLE  
ami-0abcdef1234567890
```

Untuk contoh tambahan menggunakan filter tanda, lihat [Bekerja dengan tanda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeImages](#) di Referensi AWS CLI Perintah.

## describe-import-image-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-import-image-tasks`.

### AWS CLI

Untuk memantau tugas impor gambar

`describe-import-image-tasks` Contoh berikut memeriksa status tugas gambar impor yang ditentukan.

```
aws ec2 describe-import-image-tasks \  
  --import-task-ids import-ami-1234567890abcdef0
```

Output untuk tugas impor gambar yang sedang berlangsung.

```
{  
  "ImportImageTasks": [  
    {  
      "ImportTaskId": "import-ami-1234567890abcdef0",  
      "Progress": "28",  
      "SnapshotDetails": [  
        {  
          "DiskImageSize": 705638400.0,  
          "Format": "ova",  
          "Status": "completed",  
          "UserBucket": {  
            "S3Bucket": "my-import-bucket",  
            "S3Key": "vms/my-server-vm.ova"  
          }  
        }  
      ]  
    }  
  ]  
}
```



```

    }
  },
  "Status": "active",
  "StatusMessage": "converting"
}
]
}

```

Output untuk tugas gambar impor yang selesai. ID AMI yang dihasilkan disediakan oleh `ImageId`.

```

{
  "ImportImageTasks": [
    {
      "ImportTaskId": "import-ami-1234567890abcdef0",
      "ImageId": "ami-1234567890abcdef0",
      "SnapshotDetails": [
        {
          "DiskImageSize": 705638400.0,
          "Format": "ova",
          "SnapshotId": "snap-1234567890abcdef0"
          "Status": "completed",
          "UserBucket": {
            "S3Bucket": "my-import-bucket",
            "S3Key": "vms/my-server-vm.ova"
          }
        }
      ],
      "Status": "completed"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeImportImageTasks](#) di Referensi AWS CLI Perintah.

## describe-import-snapshot-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-import-snapshot-tasks`.

### AWS CLI

Untuk memantau tugas snapshot impor

`describe-import-snapshot-tasks` Contoh berikut memeriksa status tugas snapshot impor yang ditentukan.

```
aws ec2 describe-import-snapshot-tasks \
  --import-task-ids import-snap-1234567890abcdef0
```

Output untuk tugas snapshot impor yang sedang berlangsung:

```
{
  "ImportSnapshotTasks": [
    {
      "Description": "My server VMDK",
      "ImportTaskId": "import-snap-1234567890abcdef0",
      "SnapshotTaskDetail": {
        "Description": "My server VMDK",
        "DiskImageSize": "705638400.0",
        "Format": "VMDK",
        "Progress": "42",
        "Status": "active",
        "StatusMessage": "downloading/convertng",
        "UserBucket": {
          "S3Bucket": "my-import-bucket",
          "S3Key": "vms/my-server-vm.vmdk"
        }
      }
    }
  ]
}
```

Output untuk tugas snapshot impor yang selesai. ID snapshot yang dihasilkan disediakan oleh `SnapshotId`.

```
{
  "ImportSnapshotTasks": [
    {
      "Description": "My server VMDK",
      "ImportTaskId": "import-snap-1234567890abcdef0",
      "SnapshotTaskDetail": {
        "Description": "My server VMDK",
        "DiskImageSize": "705638400.0",
        "Format": "VMDK",
        "SnapshotId": "snap-1234567890abcdef0"
      }
    }
  ]
}
```

```

        "Status": "completed",
        "UserBucket": {
            "S3Bucket": "my-import-bucket",
            "S3Key": "vms/my-server-vm.vmdk"
        }
    }
}
]
}

```

- Untuk detail API, lihat [DescribeImportSnapshotTasks](#) di Referensi AWS CLI Perintah.

## describe-instance-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-attribute`.

### AWS CLI

Untuk menggambarkan jenis instance

Contoh ini menjelaskan jenis instance dari instance yang ditentukan.

Perintah:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
instanceType
```

Output:

```
{
  "InstanceId": "i-1234567890abcdef0"
  "InstanceType": {
    "Value": "t1.micro"
  }
}
```

Untuk menggambarkan `disableApiTermination` atribut

Contoh ini menjelaskan `disableApiTermination` atribut dari contoh yang ditentukan.

Perintah:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
disableApiTermination
```

Output:

```
{
  "InstanceId": "i-1234567890abcdef0"
  "DisableApiTermination": {
    "Value": "false"
  }
}
```

Untuk menggambarkan pemetaan perangkat blok untuk sebuah instance

Contoh ini menjelaskan `blockDeviceMapping` atribut dari contoh yang ditentukan.

Perintah:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
blockDeviceMapping
```

Output:

```
{
  "InstanceId": "i-1234567890abcdef0"
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/sda1",
      "Ebs": {
        "Status": "attached",
        "DeleteOnTermination": true,
        "VolumeId": "vol-049df61146c4d7901",
        "AttachTime": "2013-05-17T22:42:34.000Z"
      }
    },
    {
      "DeviceName": "/dev/sdf",
      "Ebs": {
        "Status": "attached",
        "DeleteOnTermination": false,
        "VolumeId": "vol-049df61146c4d7901",
```

```

    "AttachTime": "2013-09-10T23:07:00.000Z"
  }
}
],
}

```

- Untuk detail API, lihat [DescribeInstanceAttribute](#) di Referensi AWS CLI Perintah.

## describe-instance-connect-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-connect-endpoints`.

### AWS CLI

Untuk mendeskripsikan Instance Connect Endpoint EC2

`describe-instance-connect-endpoints` Contoh berikut menjelaskan EC2 Instance Connect Endpoint yang ditentukan.

```

aws ec2 describe-instance-connect-endpoints \
  --region us-east-1 \
  --instance-connect-endpoint-ids eice-0123456789example

```

### Output:

```

{
  "InstanceConnectEndpoints": [
    {
      "OwnerId": "111111111111",
      "InstanceConnectEndpointId": "eice-0123456789example",
      "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-connect-endpoint/eice-0123456789example",
      "State": "create-complete",
      "StateMessage": "",
      "DnsName": "eice-0123456789example.b67b86ba.ec2-instance-connect-endpoint.us-east-1.amazonaws.com",
      "NetworkInterfaceIds": [
        "eni-0123456789example"
      ],
      "VpcId": "vpc-0123abcd",
      "AvailabilityZone": "us-east-1d",
    }
  ]
}

```

```
        "CreatedAt": "2023-02-07T12:05:37+00:00",
        "SubnetId": "subnet-0123abcd",
        "Tags": []
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Membuat Titik Akhir Connect Instans EC2](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeInstanceConnectEndpoints](#) di Referensi AWS CLI Perintah.

## describe-instance-credit-specifications

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-credit-specifications`.

### AWS CLI

Untuk menjelaskan opsi kredit untuk penggunaan CPU dari satu atau lebih contoh

`describe-instance-credit-specifications` Contoh berikut menjelaskan opsi kredit CPU untuk contoh yang ditentukan.

```
aws ec2 describe-instance-credit-specifications \
  --instance-ids i-1234567890abcdef0
```

### Output:

```
{
  "InstanceCreditSpecifications": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "CpuCredits": "unlimited"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan instans performa burstable](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeInstanceCreditSpecifications](#) di Referensi AWS CLI Perintah.

## describe-instance-event-notification-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-event-notification-attributes`.

### AWS CLI

Untuk menjelaskan tag untuk pemberitahuan acara terjadwal

`describe-instance-event-notification-attributes` Contoh berikut menjelaskan tag yang akan muncul dalam pemberitahuan acara terjadwal.

```
aws ec2 describe-instance-event-notification-attributes
```

Output:

```
{
  "InstanceTagAttribute": {
    "InstanceTagKeys": [],
    "IncludeAllTagsOfInstance": true
  }
}
```

Untuk informasi selengkapnya, lihat [Acara terjadwal untuk instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeInstanceEventNotificationAttributes](#) di Referensi AWS CLI Perintah.

## describe-instance-event-windows

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-event-windows`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan semua jendela acara

`describe-instance-event-windows` Contoh berikut menjelaskan semua jendela acara di Wilayah tertentu.

```
aws ec2 describe-instance-event-windows \
```

```
--region us-east-1
```

Output:

```
{
  "InstanceEventWindows": [
    {
      "InstanceEventWindowId": "iew-0abcdef1234567890",
      "Name": "myEventWindowName",
      "CronExpression": "* 21-23 * * 2,3",
      "AssociationTarget": {
        "InstanceIds": [
          "i-1234567890abcdef0",
          "i-0598c7d356eba48d7"
        ],
        "Tags": [],
        "DedicatedHostIds": []
      },
      "State": "active",
      "Tags": []
    }
    ...
  ],
  "NextToken": "9d624e0c-388b-4862-a31e-a85c64fc1d4a"
}
```

Contoh 2: Untuk mendeskripsikan jendela acara tertentu

`describe-instance-event-windows` Contoh berikut menjelaskan peristiwa tertentu dengan menggunakan `instance-event-window` parameter untuk menggambarkan jendela peristiwa tertentu.

```
aws ec2 describe-instance-event-windows \
  --region us-east-1 \
  --instance-event-window-ids iew-0abcdef1234567890
```

Output:

```
{
  "InstanceEventWindows": [
```



```

    {
      "InstanceEventWindowId": "iew-0abcdef1234567890",
      "Name": "myEventWindowName",
      "CronExpression": "* 21-23 * * 2,3",
      "AssociationTarget": {
        "InstanceIds": [
          "i-1234567890abcdef0",
          "i-0598c7d356eba48d7"
        ],
        "Tags": [],
        "DedicatedHostIds": []
      },
      "State": "active",
      "Tags": []
    }
  }
}

```

Contoh 3: Untuk mendeskripsikan jendela acara yang cocok dengan satu atau beberapa filter `describe-instance-event-windows` Contoh berikut menjelaskan jendela acara yang cocok dengan satu atau lebih filter menggunakan filter parameter. `instance-idFilter` digunakan untuk menggambarkan semua jendela acara yang terkait dengan instance yang ditentukan. Saat digunakan, filter melakukan pencocokan langsung. Namun, filter `instance-id` berbeda. Jika tidak ada kecocokan langsung dengan ID instance, maka itu akan kembali ke asosiasi tidak langsung dengan jendela acara, seperti tag instance atau ID Host Khusus (jika instance adalah Host Khusus).

```

aws ec2 describe-instance-event-windows \
  --region us-east-1 \
  --filters Name=instance-id,Values=i-1234567890abcdef0 \
  --max-results 100 \
  --next-token <next-token-value>

```

Output:

```

{
  "InstanceEventWindows": [
    {
      "InstanceEventWindowId": "iew-0dbc0adb66f235982",
      "TimeRanges": [
        {
          "StartWeekDay": "sunday",

```

```

        "StartHour": 2,
        "EndWeekDay": "sunday",
        "EndHour": 8
    }
],
"Name": "myEventWindowName",
"AssociationTarget": {
    "InstanceIds": [],
    "Tags": [],
    "DedicatedHostIds": [
        "h-0140d9a7ecbd102dd"
    ]
},
"State": "active",
"Tags": []
}
]
}

```

Dalam contoh output, instance berada di Dedicated Host, yang dikaitkan dengan jendela acara.

Untuk batasan jendela acara, lihat [Pertimbangan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeInstanceEventWindows](#) di Referensi AWS CLI Perintah.

## describe-instance-status

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-status`.

### AWS CLI

Untuk menjelaskan status instans

Contoh `describe-instance-status` berikut menjelaskan status saat ini dari instans tertentu.

```
aws ec2 describe-instance-status \
  --instance-ids i-1234567890abcdef0
```

Output:

```
{
  "InstanceStatuses": [
    {
```

```

    "InstanceId": "i-1234567890abcdef0",
    "InstanceState": {
      "Code": 16,
      "Name": "running"
    },
    "AvailabilityZone": "us-east-1d",
    "SystemStatus": {
      "Status": "ok",
      "Details": [
        {
          "Status": "passed",
          "Name": "reachability"
        }
      ]
    },
    "InstanceState": {
      "Status": "ok",
      "Details": [
        {
          "Status": "passed",
          "Name": "reachability"
        }
      ]
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Pantau status instans Anda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeInstanceStatus](#) di Referensi AWS CLI Perintah.

## describe-instance-topology

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-topology`.

### AWS CLI

Untuk menggambarkan topologi instance dari semua instance Anda

`describe-instance-topology` Contoh berikut menjelaskan topologi semua instance Anda yang cocok dengan tipe instans yang didukung untuk perintah ini.

```
aws ec2 describe-instance-topology \  
  --region us-west-2
```

**Output:**

```
{  
  "Instances": [  
    {  
      "InstanceId": "i-1111111111example",  
      "InstanceType": "p4d.24xlarge",  
      "GroupName": "my-ml-cpg",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "usw2-az2",  
      "AvailabilityZone": "us-west-2a"  
    },  
    {  
      "InstanceId": "i-2222222222example",  
      "InstanceType": "p4d.24xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "usw2-az2",  
      "AvailabilityZone": "us-west-2a"  
    },  
    {  
      "InstanceId": "i-3333333333example",  
      "InstanceType": "trn1.32xlarge",  
      "NetworkNodes": [  
        "nn-1212121212example",  
        "nn-1211122211example",  
        "nn-1311133311example"  
      ],  
      "ZoneId": "usw2-az4",  
      "AvailabilityZone": "us-west-2d"  
    },  
    {  
      "InstanceId": "i-4444444444example",
```

```
    "InstanceType": "trn1.2xlarge",
    "NetworkNodes": [
        "nn-1111111111example",
        "nn-5434334334example",
        "nn-1235301234example"
    ],
    "ZoneId": "usw2-az2",
    "AvailabilityZone": "us-west-2a"
  }
],
"NextToken": "SomeEncryptedToken"
}
```

Untuk informasi selengkapnya, termasuk contoh lainnya, lihat [topologi instans Amazon EC2 di Panduan Pengguna Amazon EC2](#).

- Untuk detail API, lihat [DescribeInstanceTopology](#) di Referensi AWS CLI Perintah.

## describe-instance-type-offerings

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-type-offerings`.

### AWS CLI

Contoh 1: Untuk mencantumkan jenis instans yang ditawarkan di Wilayah

`describe-instance-type-offerings` Contoh berikut mencantumkan jenis instance yang ditawarkan di Wilayah yang dikonfigurasi sebagai Region default untuk AWS CLI.

```
aws ec2 describe-instance-type-offerings
```

Untuk mencantumkan jenis instance yang ditawarkan di Region yang berbeda, tentukan Region menggunakan `--region` parameter.

```
aws ec2 describe-instance-type-offerings \
  --region us-east-2
```

Output:

```
{
  "InstanceTypeOfferings": [
    {
```

```

        "InstanceType": "m5.2xlarge",
        "LocationType": "region",
        "Location": "us-east-2"
    },
    {
        "InstanceType": "t3.micro",
        "LocationType": "region",
        "Location": "us-east-2"
    },
    ...
]
}

```

Contoh 2: Untuk mencantumkan jenis instans yang ditawarkan di Availability Zone

`describe-instance-type-offerings` Contoh berikut mencantumkan jenis instance yang ditawarkan di Availability Zone yang ditentukan. Availability Zone harus berada di Region yang ditentukan.

```

aws ec2 describe-instance-type-offerings \
  --location-type availability-zone \
  --filters Name=location,Values=us-east-2a \
  --region us-east-2

```

Contoh 3: Untuk memeriksa apakah jenis instance didukung

`describe-instance-type-offerings` Perintah berikut menunjukkan apakah jenis `c5.xlarge` instance didukung di Wilayah tertentu.

```

aws ec2 describe-instance-type-offerings \
  --filters Name=instance-type,Values=c5.xlarge \
  --region us-east-2

```

`describe-instance-type-offerings` Contoh berikut mencantumkan semua jenis instance C5 yang didukung di Wilayah tertentu.

```

aws ec2 describe-instance-type-offerings \
  --filters Name=instance-type,Values=c5* \
  --query "InstanceTypeOfferings[].InstanceType" \
  --region us-east-2

```

Output:

```
[
  "c5d.12xlarge",
  "c5d.9xlarge",
  "c5n.xlarge",
  "c5.xlarge",
  "c5d.metal",
  "c5n.metal",
  "c5.large",
  "c5d.2xlarge",
  "c5n.4xlarge",
  "c5.2xlarge",
  "c5n.large",
  "c5n.9xlarge",
  "c5d.large",
  "c5.18xlarge",
  "c5d.18xlarge",
  "c5.12xlarge",
  "c5n.18xlarge",
  "c5.metal",
  "c5d.4xlarge",
  "c5.24xlarge",
  "c5d.xlarge",
  "c5n.2xlarge",
  "c5d.24xlarge",
  "c5.9xlarge",
  "c5.4xlarge"
]
```

- Untuk detail API, lihat [DescribeInstanceTypeOfferings](#) di Referensi AWS CLI Perintah.

## **describe-instance-types**

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-types`.

### AWS CLI

Contoh 1: Untuk menjelaskan tipe instans

Contoh `describe-instance-types` berikut menampilkan detail untuk tipe instans tertentu.

```
aws ec2 describe-instance-types \
  --instance-types t2.micro
```

## Output:

```
{
  "InstanceTypes": [
    {
      "InstanceType": "t2.micro",
      "CurrentGeneration": true,
      "FreeTierEligible": true,
      "SupportedUsageClasses": [
        "on-demand",
        "spot"
      ],
      "SupportedRootDeviceTypes": [
        "ebs"
      ],
      "BareMetal": false,
      "Hypervisor": "xen",
      "ProcessorInfo": {
        "SupportedArchitectures": [
          "i386",
          "x86_64"
        ],
        "SustainedClockSpeedInGhz": 2.5
      },
      "VCpuInfo": {
        "DefaultVCpus": 1,
        "DefaultCores": 1,
        "DefaultThreadsPerCore": 1,
        "ValidCores": [
          1
        ],
        "ValidThreadsPerCore": [
          1
        ]
      },
      "MemoryInfo": {
        "SizeInMiB": 1024
      },
      "InstanceStorageSupported": false,
      "EbsInfo": {
        "EbsOptimizedSupport": "unsupported",
        "EncryptionSupport": "supported"
      },
      "NetworkInfo": {
```



```
        "NetworkPerformance": "Low to Moderate",
        "MaximumNetworkInterfaces": 2,
        "Ipv4AddressesPerInterface": 2,
        "Ipv6AddressesPerInterface": 2,
        "Ipv6Supported": true,
        "EnaSupport": "unsupported"
    },
    "PlacementGroupInfo": {
        "SupportedStrategies": [
            "partition",
            "spread"
        ]
    },
    "HibernationSupported": false,
    "BurstablePerformanceSupported": true,
    "DedicatedHostsSupported": false,
    "AutoRecoverySupported": true
}
]
}
```

Untuk informasi selengkapnya, lihat [Jenis Instans](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 2: Untuk memfilter tipe instans yang tersedia

Anda dapat menentukan filter guna mencakup hasil untuk tipe instans yang memiliki karakteristik khusus. Contoh `describe-instance-types` berikut mencantumkan tipe instans yang mendukung hibernasi.

```
aws ec2 describe-instance-types \
  --filters Name=hibernation-supported,Values=true --query
  'InstanceTypes[*].InstanceType'
```

Output:

```
[
  "m5.8xlarge",
  "r3.large",
  "c3.8xlarge",
  "r5.large",
  "m4.4xlarge",
  "c4.large",
```

```
"m5.xlarge",  
"m4.xlarge",  
"c3.large",  
"c4.8xlarge",  
"c4.4xlarge",  
"c5.xlarge",  
"c5.12xlarge",  
"r5.4xlarge",  
"c5.4xlarge"  
]
```

Untuk informasi selengkapnya, lihat [Jenis Instans](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [DescribeInstanceTypes](#) di Referensi AWS CLI Perintah.

## describe-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-instances`.

### AWS CLI

Contoh 1: Untuk menjelaskan instans

Contoh `describe-instances` berikut menjelaskan instans tertentu.

```
aws ec2 describe-instances \  
--instance-ids i-1234567890abcdef0
```

Output:

```
{  
  "Reservations": [  
    {  
      "Groups": [],  
      "Instances": [  
        {  
          "AmiLaunchIndex": 0,  
          "ImageId": "ami-0abcdef1234567890",  
          "InstanceId": "i-1234567890abcdef0",  
          "InstanceType": "t3.nano",  
          "KeyName": "my-key-pair",  
          "LaunchTime": "2022-11-15T10:48:59+00:00",
```

```
    "Monitoring": {
      "State": "disabled"
    },
    "Placement": {
      "AvailabilityZone": "us-east-2a",
      "GroupName": "",
      "Tenancy": "default"
    },
    "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",
    "PrivateIpAddress": "10-0-0-157",
    "ProductCodes": [],
    "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
    "PublicIpAddress": "34.253.223.13",
    "State": {
      "Code": 16,
      "Name": "running"
    },
    "StateTransitionReason": "",
    "SubnetId": "subnet-04a636d18e83cfacb",
    "VpcId": "vpc-1234567890abcdef0",
    "Architecture": "x86_64",
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/xvda",
        "Ebs": {
          "AttachTime": "2022-11-15T10:49:00+00:00",
          "DeleteOnTermination": true,
          "Status": "attached",
          "VolumeId": "vol-02e6ccdca7de29cf2"
        }
      }
    ],
    "ClientToken": "1234abcd-1234-abcd-1234-d46a8903e9bc",
    "EbsOptimized": true,
    "EnaSupport": true,
    "Hypervisor": "xen",
    "IamInstanceProfile": {
      "Arn": "arn:aws:iam::111111111111:instance-profile/
AmazonSSMRoleForInstancesQuickSetup",
      "Id": "11111111111111111111111111111111"
    },
    "NetworkInterfaces": [
      {
```

```
    "Association": {
      "IpOwnerId": "amazon",
      "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
      "PublicIp": "34.253.223.13"
    },
    "Attachment": {
      "AttachTime": "2022-11-15T10:48:59+00:00",
      "AttachmentId": "eni-attach-1234567890abcdefg",
      "DeleteOnTermination": true,
      "DeviceIndex": 0,
      "Status": "attached",
      "NetworkCardIndex": 0
    },
    "Description": "",
    "Groups": [
      {
        "GroupName": "launch-wizard-146",
        "GroupId": "sg-1234567890abcdefg"
      }
    ],
    "Ipv6Addresses": [],
    "MacAddress": "00:11:22:33:44:55",
    "NetworkInterfaceId": "eni-1234567890abcdefg",
    "OwnerId": "104024344472",
    "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
    "PrivateIpAddress": "10-0-0-157",
    "PrivateIpAddresses": [
      {
        "Association": {
          "IpOwnerId": "amazon",
          "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
          "PublicIp": "34.253.223.13"
        },
        "Primary": true,
        "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
        "PrivateIpAddress": "10-0-0-157"
      }
    ],
    "SourceDestCheck": true,
    "Status": "in-use",
```

```
        "SubnetId": "subnet-1234567890abcdefg",
        "VpcId": "vpc-1234567890abcdefg",
        "InterfaceType": "interface"
    }
],
"RootDeviceName": "/dev/xvda",
"RootDeviceType": "ebs",
"SecurityGroups": [
    {
        "GroupName": "launch-wizard-146",
        "GroupId": "sg-1234567890abcdefg"
    }
],
"SourceDestCheck": true,
"Tags": [
    {
        "Key": "Name",
        "Value": "my-instance"
    }
],
"VirtualizationType": "hvm",
"CpuOptions": {
    "CoreCount": 1,
    "ThreadsPerCore": 2
},
"CapacityReservationSpecification": {
    "CapacityReservationPreference": "open"
},
"HibernationOptions": {
    "Configured": false
},
"MetadataOptions": {
    "State": "applied",
    "HttpTokens": "optional",
    "HttpPutResponseHopLimit": 1,
    "HttpEndpoint": "enabled",
    "HttpProtocolIpv6": "disabled",
    "InstanceMetadataTags": "enabled"
},
"EnclaveOptions": {
    "Enabled": false
},
"PlatformDetails": "Linux/UNIX",
"UsageOperation": "RunInstances",
```

```

        "UsageOperationUpdateTime": "2022-11-15T10:48:59+00:00",
        "PrivateDnsNameOptions": {
            "HostnameType": "ip-name",
            "EnableResourceNameDnsARecord": true,
            "EnableResourceNameDnsAAAARecord": false
        },
        "MaintenanceOptions": {
            "AutoRecovery": "default"
        }
    },
    "OwnerId": "111111111111",
    "ReservationId": "r-1234567890abcdefg"
}
]
}

```

Contoh 2: Untuk memfilter instans dengan tipe tertentu

Contoh `describe-instances` berikut menggunakan filter guna mencakup hasil untuk instans dari tipe tertentu.

```
aws ec2 describe-instances \
  --filters Name=instance-type,Values=m5.large
```

Untuk contoh output, lihat Contoh 1.

Untuk informasi selengkapnya, lihat [Cantumkan dan filter menggunakan CLI](#) di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk memfilter instans dengan tipe dan Zona Ketersediaan tertentu

Contoh `describe-instances` berikut menggunakan banyak filter guna mencakup hasil untuk instans dengan tipe tertentu yang juga ada di Zona Ketersediaan tertentu.

```
aws ec2 describe-instances \
  --filters Name=instance-type,Values=t2.micro,t3.micro Name=availability-
  zone,Values=us-east-2c
```

Untuk contoh output, lihat Contoh 1.

Contoh 4: Untuk memfilter instans dengan tipe dan Zona Ketersediaan tertentu menggunakan file JSON

Contoh `describe-instances` berikut menggunakan file input JSON untuk melakukan pemfilteran yang sama seperti contoh sebelumnya. Ketika filter menjadi lebih rumit, filter tersebut dapat lebih mudah ditentukan dalam file JSON.

```
aws ec2 describe-instances \  
  --filters file://filters.json
```

Isi dari `filters.json`:

```
[  
  {  
    "Name": "instance-type",  
    "Values": ["t2.micro", "t3.micro"]  
  },  
  {  
    "Name": "availability-zone",  
    "Values": ["us-east-2c"]  
  }  
]
```

Untuk contoh output, lihat Contoh 1.

Contoh 5: Untuk memfilter instans dengan tanda Pemilik tertentu

Contoh `describe-instances` berikut menggunakan filter tanda guna mencakup hasil untuk instans yang memiliki tanda dengan kunci tanda tertentu (Pemilik), terlepas dari nilai tandanya.

```
aws ec2 describe-instances \  
  --filters "Name=tag-key,Values=Owner"
```

Untuk contoh output, lihat Contoh 1.

Contoh 6: Untuk memfilter instans dengan nilai tanda my-team tertentu

Contoh `describe-instances` berikut menggunakan filter tanda guna mencakup hasil untuk instans yang memiliki tanda dengan kunci tanda tertentu (`my-team`), terlepas dari nilai tandanya.

```
aws ec2 describe-instances \  
  --filters "Name=tag-value,Values=my-team"
```

Untuk contoh output, lihat Contoh 1.

Contoh 7: Untuk memfilter instans dengan tanda Pemilik dan nilai my-team tertentu

Contoh `describe-instances` berikut menggunakan filter tanda guna mencakup hasil untuk instans yang memiliki tanda tertentu (Pemilik=my-team).

```
aws ec2 describe-instances \  
  --filters "Name=tag:Owner,Values=my-team"
```

Untuk contoh output, lihat Contoh 1.

Contoh 8: Guna hanya menampilkan ID instans dan subnet untuk semua instans

Contoh `describe-instances` berikut menggunakan parameter `--query` untuk hanya menampilkan ID instans dan subnet untuk semua instans, dalam format JSON.

Linux dan macOS:

```
aws ec2 describe-instances \  
  --query 'Reservations[*].Instances[*].{Instance:InstanceId,Subnet:SubnetId}' \  
  --output json
```

Windows:

```
aws ec2 describe-instances ^  
  --query "Reservations[*].Instances[*].{Instance:InstanceId,Subnet:SubnetId}" ^  
  --output json
```

Output:

```
[  
  {  
    "Instance": "i-057750d42936e468a",  
    "Subnet": "subnet-069beee9b12030077"  
  },  
  {  
    "Instance": "i-001efd250faaa6ffa",  
    "Subnet": "subnet-0b715c6b7db68927a"  
  },  
  {  
    "Instance": "i-027552a73f021f3bd",  
    "Subnet": "subnet-0250c25a1f4e15235"  
  }  
]
```



```

    }
    ...
]

```

Contoh 9: Untuk memfilter instans dari tipe tertentu dan hanya menampilkan ID instansnya

Contoh `describe-instances` berikut menggunakan filter guna mencakup hasil untuk instans dari tipe tertentu dan parameter `--query` agar hanya menampilkan ID instans.

```

aws ec2 describe-instances \
  --filters "Name=instance-type,Values=t2.micro" \
  --query "Reservations[*].Instances[*].[InstanceId]" \
  --output text

```

Output:

```

i-031c0dc19de2fb70c
i-00d8bfff789a736b75
i-0b715c6b7db68927a
i-0626d4edd54f1286d
i-00b8ae04f9f99908e
i-0fc71c25d2374130c

```

Contoh 10: Untuk memfilter instans dari tipe tertentu dan hanya menampilkan ID instans, Zona Ketersediaan, dan nilai tanda tertentu

Contoh `describe-instances` berikut menampilkan ID instans, Zona Ketersediaan, dan nilai dari tanda Name untuk instans yang memiliki tanda dengan nama tag-key, dalam format tabel.

Linux dan macOS:

```

aws ec2 describe-instances \
  --filters Name=tag-key,Values=Name \
  --query 'Reservations[*].Instances[*].
{Instance:InstanceId,AZ:Placement.AvailabilityZone,Name:Tags[?Key==`Name`]|
[0].Value}' \
  --output table

```

Windows:

```

aws ec2 describe-instances ^

```

```

--filters Name=tag-key,Values=Name ^
--query "Reservations[*].Instances[*].
{Instance:InstanceId,AZ:Placement.AvailabilityZone,Name:Tags[?Key=='Name']|
[0].Value}" ^
--output table

```

### Output:

```

-----
|                               DescribeInstances                               |
+-----+-----+-----+
|      AZ      | Instance      |      Name      |
+-----+-----+-----+
| us-east-2b  | i-057750d42936e468a | my-prod-server |
| us-east-2a  | i-001efd250faaa6ffa | test-server-1   |
| us-east-2a  | i-027552a73f021f3bd | test-server-2   |
+-----+-----+-----+

```

Contoh 11: Untuk menjelaskan instans dalam grup penempatan partisi

Contoh `describe-instances` berikut menjelaskan instans tertentu. Output-nya mencakup informasi penempatan untuk instans, yang berisi nama grup penempatan dan nomor partisi untuk instans tersebut.

```

aws ec2 describe-instances \
  --instance-ids i-0123a456700123456 \
  --query "Reservations[*].Instances[*].Placement"

```

### Output:

```

[
  [
    {
      "AvailabilityZone": "us-east-1c",
      "GroupName": "HDFS-Group-A",
      "PartitionNumber": 3,
      "Tenancy": "default"
    }
  ]
]

```

Untuk informasi selengkapnya, lihat [Menjelaskan instans dalam grup penempatan](#) di Panduan Pengguna Amazon EC2.

Contoh 12: Untuk memfilter ke instans dengan grup penempatan dan nomor partisi tertentu

Contoh `describe-instances` berikut memfilter hasil menjadi hanya instans dengan grup penempatan dan nomor partisi tertentu.

```
aws ec2 describe-instances \  
  --filters "Name=placement-group-name,Values=HDFS-Group-A" "Name=placement-  
partition-number,Values=7"
```

Berikut ini hanya menunjukkan informasi yang relevan dari output.

```
"Instances": [  
  {  
    "InstanceId": "i-0123a456700123456",  
    "InstanceType": "r4.large",  
    "Placement": {  
      "AvailabilityZone": "us-east-1c",  
      "GroupName": "HDFS-Group-A",  
      "PartitionNumber": 7,  
      "Tenancy": "default"  
    }  
  },  
  {  
    "InstanceId": "i-9876a543210987654",  
    "InstanceType": "r4.large",  
    "Placement": {  
      "AvailabilityZone": "us-east-1c",  
      "GroupName": "HDFS-Group-A",  
      "PartitionNumber": 7,  
      "Tenancy": "default"  
    }  
  },  
],
```

Untuk informasi selengkapnya, lihat [Menjelaskan instans dalam grup penempatan](#) di Panduan Pengguna Amazon EC2.

Contoh 13: Untuk memfilter ke instans yang dikonfigurasi guna memungkinkan akses ke tanda dari metadata instans

Contoh `describe-instances` berikut memfilter hasil menjadi hanya instans yang dikonfigurasi guna memungkinkan akses ke tanda instans dari metadata instans.

```
aws ec2 describe-instances \  
  --filters "Name=metadata-options.instance-metadata-tags,Values=enabled" \  
  --query "Reservations[*].Instances[*].InstanceId" \  
  --output text
```

Berikut menunjukkan output yang diharapkan.

```
i-1234567890abcdefg  
i-abcdefg1234567890  
i-111111111aaaaaaaa  
i-aaaaaaaa111111111
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tanda instans dalam metadata instans](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeInstances](#) di Referensi AWS CLI Perintah.

## **describe-internet-gateways**

Contoh kode berikut menunjukkan cara menggunakan `describe-internet-gateways`.

### AWS CLI

Untuk menggambarkan gateway internet

`describe-internet-gateways` Contoh berikut menjelaskan gateway internet yang ditentukan.

```
aws ec2 describe-internet-gateways \  
  --internet-gateway-ids igw-0d0fb496b3EXAMPLE
```

Output:

```
{  
  "InternetGateways": [  
    {  
      "Attachments": [  
        {
```

```

        "State": "available",
        "VpcId": "vpc-0a60eb65b4EXAMPLE"
    }
],
"InternetGatewayId": "igw-0d0fb496b3EXAMPLE",
"OwnerId": "123456789012",
"Tags": [
    {
        "Key": "Name",
        "Value": "my-igw"
    }
]
}
]
}

```

Untuk informasi lebih lanjut, lihat [Gateway internet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeInternetGateways](#) di Referensi AWS CLI Perintah.

## describe-ipam-pools

Contoh kode berikut menunjukkan cara menggunakan `describe-ipam-pools`.

### AWS CLI

Untuk melihat detail untuk kolam IPAM

`describe-ipam-pools` Contoh berikut menunjukkan rincian untuk pool.

(Linux):

```
aws ec2 describe-ipam-pools \
  --filters Name=owner-id,Values=123456789012 Name=ipam-scope-id,Values=ipam-
  scope-02fc38cd4c48e7d38
```

(Windows):

```
aws ec2 describe-ipam-pools ^
  --filters Name=owner-id,Values=123456789012 Name=ipam-scope-id,Values=ipam-
  scope-02fc38cd4c48e7d38
```

**Output:**

```
{
  "IpamPools": [
    {
      "OwnerId": "123456789012",
      "IpamPoolId": "ipam-pool-02ec043a19bbe5d08",
      "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-02ec043a19bbe5d08",
      "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
      "IpamScopeType": "private",
      "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
      "IpamRegion": "us-east-1",
      "Locale": "None",
      "PoolDepth": 1,
      "State": "create-complete",
      "AutoImport": true,
      "AddressFamily": "ipv4",
      "AllocationMinNetmaskLength": 16,
      "AllocationMaxNetmaskLength": 26,
      "AllocationDefaultNetmaskLength": 24,
      "AllocationResourceTags": [
        {
          "Key": "Environment",
          "Value": "Preprod"
        }
      ],
      "Tags": [
        {
          "Key": "Name",
          "Value": "Preprod pool"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeIpamPools](#) di Referensi AWS CLI Perintah.

**describe-ipam-resource-discoveries**

Contoh kode berikut menunjukkan cara menggunakan `describe-ipam-resource-discoveries`.

## AWS CLI

### Contoh 1: Lihat detail lengkap penemuan sumber daya

Dalam contoh ini, Anda adalah admin IPAM yang didelegasikan yang ingin membuat dan berbagi penemuan sumber daya dengan admin IPAM di AWS Organisasi lain sehingga admin dapat mengelola dan memantau alamat IP sumber daya di organisasi Anda.

Contoh ini mungkin berguna jika:

Anda mencoba membuat penemuan sumber daya, tetapi Anda mendapat kesalahan bahwa Anda telah mencapai batas 1. Anda menyadari bahwa Anda mungkin telah membuat penemuan sumber daya dan Anda ingin melihatnya di akun Anda. Anda memiliki sumber daya di Wilayah yang tidak ditemukan oleh IPAM. Anda ingin melihat yang `--operating-regions` ditentukan untuk sumber daya dan memastikan bahwa Anda telah menambahkan Wilayah yang tepat sebagai Wilayah operasi sehingga sumber daya di sana dapat ditemukan.

`describe-ipam-resource-discoveries` Contoh berikut mencantumkan rincian penemuan sumber daya di AWS akun Anda. Anda dapat memiliki satu penemuan sumber daya per AWS Wilayah.

```
aws ec2 describe-ipam-resource-discoveries \
  --region us-east-1
```

### Output:

```
{
  "IpamResourceDiscoveries": [
    {
      "OwnerId": "149977607591",
      "IpamResourceDiscoveryId": "ipam-res-disco-0f8bdee9067137c0d",
      "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-
discovery/ipam-res-disco-0f8bdee9067137c0d",
      "IpamResourceDiscoveryRegion": "us-east-1",
      "OperatingRegions": [
        {
          "RegionName": "us-east-1"
        }
      ],
      "IsDefault": false,
      "State": "create-complete",
      "Tags": []
    }
  ]
}
```

```
}  
]  
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

Contoh 2: Lihat hanya ID penemuan sumber daya

`describe-ipam-resource-discoveries` Contoh berikut mencantumkan ID penemuan sumber daya di AWS akun Anda. Anda dapat memiliki satu penemuan sumber daya per AWS Wilayah.

```
aws ec2 describe-ipam-resource-discoveries \  
  --query "IpamResourceDiscoveries[*].IpamResourceDiscoveryId" \  
  --output text
```

Output:

```
ipam-res-disco-0481e39b242860333
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DescribeIpamResourceDiscoveries](#) di Referensi AWS CLI Perintah.

## **describe-ipam-resource-discovery-associations**

Contoh kode berikut menunjukkan cara menggunakan `describe-ipam-resource-discovery-associations`.

### AWS CLI

Untuk melihat semua asosiasi penemuan sumber daya dengan IPAM Anda

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM yang telah mengaitkan penemuan sumber daya dengan IPAM Anda untuk mengintegrasikan akun lain dengan IPAM Anda. Anda telah memperhatikan bahwa IPAM Anda tidak menemukan sumber daya di Wilayah operasi penemuan sumber daya seperti yang diharapkan. Anda ingin memeriksa status dan status penemuan sumber daya untuk memastikan bahwa akun yang membuatnya masih aktif dan penemuan sumber daya masih dibagikan.



--region Harus menjadi wilayah asal IPAM Anda.

describe-ipam-resource-discovery-associations Contoh berikut mencantumkan asosiasi penemuan sumber daya di AWS akun Anda.

```
aws ec2 describe-ipam-resource-discovery-associations \  
  --region us-east-1
```

Output:

```
{  
  "IpamResourceDiscoveryAssociations": [  
    {  
      "OwnerId": "320805250157",  
      "IpamResourceDiscoveryAssociationId": "ipam-res-disco-  
assoc-05e6b45eca5bf5cf7",  
      "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::320805250157:ipam-  
resource-discovery-association/ipam-res-disco-  
assoc-05e6b45eca5bf5cf7",  
      "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",  
      "IpamId": "ipam-005f921c17ebd5107",  
      "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",  
      "IpamRegion": "us-east-1",  
      "IsDefault": true,  
      "ResourceDiscoveryStatus": "active",  
      "State": "associate-complete",  
      "Tags": []  
    },  
    {  
      "OwnerId": "149977607591",  
      "IpamResourceDiscoveryAssociationId": "ipam-res-disco-  
assoc-0dfd21ae189ab5f62",  
      "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::149977607591:ipam-  
resource-discovery-association/ipam-res-disco-  
assoc-0dfd21ae189ab5f62",  
      "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",  
      "IpamId": "ipam-005f921c17ebd5107",  
      "IpamArn": "arn:aws:ec2::149977607591:ipam/ipam-005f921c17ebd5107",  
      "IpamRegion": "us-east-1",  
      "IsDefault": false,  
      "ResourceDiscoveryStatus": "active",  
      "State": "create-complete",  
      "Tags": []  
    }  
  ]  
}
```

```
}

```

Dalam contoh ini, setelah menjalankan perintah ini, Anda melihat bahwa Anda memiliki satu penemuan sumber daya non-default ("IsDefault": false ``) that is ``"ResourceDiscoveryStatus": "not-found" dan "State": "create-complete". Akun pemilik penemuan sumber daya telah ditutup. Jika, dalam kasus lain, Anda perhatikan itu "ResourceDiscoveryStatus": "not-found" dan "State": "associate-complete", ini menunjukkan bahwa salah satu dari hal berikut telah terjadi:

Penemuan sumber daya telah dihapus oleh pemilik penemuan sumber daya. Pemilik penemuan sumber daya tidak membagikan penemuan sumber daya.

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DescribeIpamResourceDiscoveryAssociations](#) di Referensi AWS CLI Perintah.

## describe-ipam-scopes

Contoh kode berikut menunjukkan cara menggunakan `describe-ipam-scopes`.

### AWS CLI

Untuk melihat detail untuk cakupan IPAM

`describe-ipam-scopes` Contoh berikut menunjukkan rincian untuk cakupan.

```
aws ec2 describe-ipam-scopes \
  --filters Name=owner-id,Values=123456789012 Name=ipam-
  id,Values=ipam-08440e7a3acde3908
```

Output:

```
{
  "IpamScopes": [
    {
      "OwnerId": "123456789012",
      "IpamScopeId": "ipam-scope-02fc38cd4c48e7d38",
      "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
      scope-02fc38cd4c48e7d38",
```

```

    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "IpamScopeType": "private",
    "IsDefault": true,
    "PoolCount": 2,
    "State": "create-complete",
    "Tags": []
  },
  {
    "OwnerId": "123456789012",
    "IpamScopeId": "ipam-scope-0b9eed026396dbc16",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0b9eed026396dbc16",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "IpamScopeType": "public",
    "IsDefault": true,
    "PoolCount": 0,
    "State": "create-complete",
    "Tags": []
  },
  {
    "OwnerId": "123456789012",
    "IpamScopeId": "ipam-scope-0f1aff29486355c22",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0f1aff29486355c22",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "IpamScopeType": "private",
    "IsDefault": false,
    "Description": "Example description",
    "PoolCount": 0,
    "State": "create-complete",
    "Tags": [
      {
        "Key": "Name",
        "Value": "Example name value"
      }
    ]
  }
]
}

```

- Untuk detail API, lihat [DescribepamScopes](#) di Referensi AWS CLI Perintah.

## describe-ipams

Contoh kode berikut menunjukkan cara menggunakan `describe-ipams`.

### AWS CLI

Untuk melihat detail untuk IPAM

`describe-ipams` Contoh berikut menunjukkan rincian IPAM.

```
aws ec2 describe-ipams \
  --filters Name=owner-id,Values=123456789012
```

Output:

```
{
  "Ipams": [
    {
      "OwnerId": "123456789012",
      "IpamId": "ipam-08440e7a3acde3908",
      "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
      "IpamRegion": "us-east-1",
      "PublicDefaultScopeId": "ipam-scope-0b9eed026396dbc16",
      "PrivateDefaultScopeId": "ipam-scope-02fc38cd4c48e7d38",
      "ScopeCount": 3,
      "OperatingRegions": [
        {
          "RegionName": "us-east-1"
        },
        {
          "RegionName": "us-east-2"
        },
        {
          "RegionName": "us-west-1"
        }
      ],
      "State": "create-complete",
      "Tags": [
        {
          "Key": "Name",
          "Value": "ExampleIPAM"
        }
      ]
    }
  ]
}
```

```
    }
  ]
}
```

- Untuk detail API, lihat [DescribePAMSDI](#) di Referensi AWS CLI Perintah.

## describe-ipv6-pools

Contoh kode berikut menunjukkan cara menggunakan `describe-ipv6-pools`.

### AWS CLI

Untuk mendeskripsikan kumpulan alamat IPv6 Anda

`describe-ipv6-pools` Contoh berikut menampilkan detail untuk semua kumpulan alamat IPv6 Anda.

```
aws ec2 describe-ipv6-pools
```

Output:

```
{
  "Ipv6Pools": [
    {
      "PoolId": "ipv6pool-ec2-012345abc12345abc",
      "PoolCidrBlocks": [
        {
          "Cidr": "2001:db8:123::/48"
        }
      ],
      "Tags": [
        {
          "Key": "pool-1",
          "Value": "public"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeIPv6Pools](#) di AWS CLI Command Reference.

## describe-key-pairs

Contoh kode berikut menunjukkan cara menggunakan `describe-key-pairs`.

### AWS CLI

Untuk menampilkan pasangan kunci

Contoh `describe-key-pairs` berikut menampilkan informasi tentang pasangan kunci tertentu.

```
aws ec2 describe-key-pairs \
  --key-names my-key-pair
```

Output:

```
{
  "KeyPairs": [
    {
      "KeyPairId": "key-0b94643da6EXAMPLE",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyName": "my-key-pair",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2022-05-27T21:51:16.000Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Jelaskan kunci publik](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeKeyPairs](#) di Referensi AWS CLI Perintah.

## describe-launch-template-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-launch-template-versions`.

### AWS CLI

Untuk menggambarkan versi template peluncuran

Contoh ini menjelaskan versi template peluncuran yang ditentukan.

Perintah:

```
aws ec2 describe-launch-template-versions --launch-template-id lt-068f72b72934aff71
```

**Output:**

```
{
  "LaunchTemplateVersions": [
    {
      "LaunchTemplateId": "lt-068f72b72934aff71",
      "LaunchTemplateName": "Webservers",
      "VersionNumber": 3,
      "CreatedBy": "arn:aws:iam::123456789102:root",
      "LaunchTemplateData": {
        "KeyName": "kp-us-east",
        "ImageId": "ami-6057e21a",
        "InstanceType": "t2.small",
        "NetworkInterfaces": [
          {
            "SubnetId": "subnet-7b16de0c",
            "DeviceIndex": 0,
            "Groups": [
              "sg-7c227019"
            ]
          }
        ]
      },
      "DefaultVersion": false,
      "CreateTime": "2017-11-20T13:19:54.000Z"
    },
    {
      "LaunchTemplateId": "lt-068f72b72934aff71",
      "LaunchTemplateName": "Webservers",
      "VersionNumber": 2,
      "CreatedBy": "arn:aws:iam::123456789102:root",
      "LaunchTemplateData": {
        "KeyName": "kp-us-east",
        "ImageId": "ami-6057e21a",
        "InstanceType": "t2.medium",
        "NetworkInterfaces": [
          {
            "SubnetId": "subnet-1a2b3c4d",
            "DeviceIndex": 0,
            "Groups": [
              "sg-7c227019"
            ]
          }
        ]
      }
    }
  ]
}
```

```

    ]
  }
]
},
"DefaultVersion": false,
"CreateTime": "2017-11-20T13:12:32.000Z"
},
{
  "LaunchTemplateId": "lt-068f72b72934aff71",
  "LaunchTemplateName": "Webservers",
  "VersionNumber": 1,
  "CreatedBy": "arn:aws:iam::123456789102:root",
  "LaunchTemplateData": {
    "UserData": "",
    "KeyName": "kp-us-east",
    "ImageId": "ami-aabbcc11",
    "InstanceType": "t2.medium",
    "NetworkInterfaces": [
      {
        "SubnetId": "subnet-7b16de0c",
        "DeviceIndex": 0,
        "DeleteOnTermination": false,
        "Groups": [
          "sg-7c227019"
        ],
        "AssociatePublicIpAddress": true
      }
    ]
  },
  "DefaultVersion": true,
  "CreateTime": "2017-11-20T12:52:33.000Z"
}
]
}

```

- Untuk detail API, lihat [DescribeLaunchTemplateVersions](#) di Referensi AWS CLI Perintah.

## describe-launch-templates

Contoh kode berikut menunjukkan cara menggunakan `describe-launch-templates`.



## AWS CLI

Untuk menggambarkan template peluncuran

Contoh ini menjelaskan template peluncuran Anda.

Perintah:

```
aws ec2 describe-launch-templates
```

Output:

```
{
  "LaunchTemplates": [
    {
      "LatestVersionNumber": 2,
      "LaunchTemplateId": "lt-0e06d290751193123",
      "LaunchTemplateName": "TemplateForWebServer",
      "DefaultVersionNumber": 2,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-11-27T09:30:23.000Z"
    },
    {
      "LatestVersionNumber": 6,
      "LaunchTemplateId": "lt-0c45b5e061ec98456",
      "LaunchTemplateName": "DBServersTemplate",
      "DefaultVersionNumber": 1,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-11-20T09:25:22.000Z"
    },
    {
      "LatestVersionNumber": 1,
      "LaunchTemplateId": "lt-0d47d774e8e52dabc",
      "LaunchTemplateName": "MyLaunchTemplate2",
      "DefaultVersionNumber": 1,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-11-02T12:06:21.000Z"
    },
    {
      "LatestVersionNumber": 3,
      "LaunchTemplateId": "lt-01e5f948eb4f589d6",
      "LaunchTemplateName": "testingtemplate2",
      "DefaultVersionNumber": 1,

```

```

      "CreatedBy": "arn:aws:sts::123456789012:assumed-role/AdminRole/i-03ee35176e2e5aabc",
      "CreateTime": "2017-12-01T08:19:48.000Z"
    },
  ]
}

```

- Untuk detail API, lihat [DescribeLaunchTemplates](#) di Referensi AWS CLI Perintah.

## describe-local-gateway-route-table-virtual-interface-group-associations

Contoh kode berikut menunjukkan cara menggunakan `describe-local-gateway-route-table-virtual-interface-group-associations`.

### AWS CLI

Untuk menggambarkan asosiasi antara grup antarmuka virtual dan tabel rute gateway lokal

`describe-local-gateway-route-table-virtual-interface-group-associations` Contoh berikut menjelaskan hubungan antara grup antarmuka virtual dan tabel rute gateway lokal di AWS akun Anda.

```
aws ec2 describe-local-gateway-route-table-virtual-interface-group-associations
```

### Output:

```

{
  "LocalGatewayRouteTableVirtualInterfaceGroupAssociations": [
    {
      "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-assoc-07145b276bEXAMPLE",
      "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
      "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
      "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:123456789012:local-gateway-route-table/lgw-rtb-059615ef7dEXAMPLE",
      "OwnerId": "123456789012",
      "State": "associated",
      "Tags": []
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan gateway lokal di Panduan Pengguna AWS Outposts](#).

- Untuk detail API, lihat [DescribeLocalGatewayRouteTableVirtualInterfaceGroupAssociations](#) di Referensi AWS CLI Perintah.

## **describe-local-gateway-route-table-vpc-associations**

Contoh kode berikut menunjukkan cara menggunakan `describe-local-gateway-route-table-vpc-associations`.

### AWS CLI

Untuk menggambarkan hubungan antara VPC dan tabel rute gateway lokal

`describe-local-gateway-route-table-vpc-associations` Contoh berikut menampilkan informasi tentang hubungan yang ditentukan antara VPC dan tabel rute gateway lokal.

```
aws ec2 describe-local-gateway-route-table-vpc-associations \
  --local-gateway-route-table-vpc-association-ids lgw-vpc-assoc-0e0f27af15EXAMPLE
```

### Output:

```
{
  "LocalGatewayRouteTableVpcAssociation": {
    "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-0e0f27af15EXAMPLE",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
    "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
    "VpcId": "vpc-0efe9bde08EXAMPLE",
    "State": "associated"
  }
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway lokal](#) di Panduan Pengguna Outposts.

- Untuk detail API, lihat [DescribeLocalGatewayRouteTableVpcAssociations](#) di Referensi AWS CLI Perintah.

## **describe-local-gateway-route-tables**

Contoh kode berikut menunjukkan cara menggunakan `describe-local-gateway-route-tables`.

## AWS CLI

Untuk menggambarkan Tabel Rute Gateway Lokal Anda

`describe-local-gateway-route-tables` Contoh berikut menampilkan rincian tentang tabel rute gateway lokal.

```
aws ec2 describe-local-gateway-route-tables
```

Output:

```
{
  "LocalGatewayRouteTables": [
    {
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7deEXAMPLE",
      "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
      "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-0dc11b66edEXAMPLE",
      "State": "available"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeLocalGatewayRouteTables](#) di Referensi AWS CLI Perintah.

## `describe-local-gateway-virtual-interface-groups`

Contoh kode berikut menunjukkan cara menggunakan `describe-local-gateway-virtual-interface-groups`.

## AWS CLI

Untuk menggambarkan grup antarmuka virtual gateway lokal

`describe-local-gateway-virtual-interface-groups` Contoh berikut menjelaskan grup antarmuka virtual gateway lokal di AWS akun Anda.

```
aws ec2 describe-local-gateway-virtual-interface-groups
```

Output:

```
{
  "LocalGatewayVirtualInterfaceGroups": [
    {
      "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
      "LocalGatewayVirtualInterfaceIds": [
        "lgw-vif-01a23bc4d5EXAMPLE",
        "lgw-vif-543ab21012EXAMPLE"
      ],
      "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
      "OwnerId": "123456789012",
      "Tags": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan gateway lokal di Panduan Pengguna AWS Outposts](#).

- Untuk detail API, lihat [DescribeLocalGatewayVirtualInterfaceGroups](#) di Referensi AWS CLI Perintah.

## describe-local-gateway-virtual-interfaces

Contoh kode berikut menunjukkan cara menggunakan `describe-local-gateway-virtual-interfaces`.

### AWS CLI

Untuk menggambarkan antarmuka virtual gateway lokal

`describe-local-gateway-virtual-interfaces` Contoh berikut menjelaskan antarmuka virtual gateway lokal di AWS akun Anda.

```
aws ec2 describe-local-gateway-virtual-interfaces
```

Output:

```
{
  "LocalGatewayVirtualInterfaces": [
    {
      "LocalGatewayVirtualInterfaceId": "lgw-vif-01a23bc4d5EXAMPLE",
```

```

    "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
    "Vlan": 2410,
    "LocalAddress": "0.0.0.0/0",
    "PeerAddress": "0.0.0.0/0",
    "LocalBgpAsn": 65010,
    "PeerBgpAsn": 65000,
    "OwnerId": "123456789012",
    "Tags": []
  },
  {
    "LocalGatewayVirtualInterfaceId": "lgw-vif-543ab21012EXAMPLE",
    "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
    "Vlan": 2410,
    "LocalAddress": "0.0.0.0/0",
    "PeerAddress": "0.0.0.0/0",
    "LocalBgpAsn": 65010,
    "PeerBgpAsn": 65000,
    "OwnerId": "123456789012",
    "Tags": []
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan gateway lokal di Panduan Pengguna AWS Outposts](#).

- Untuk detail API, lihat [DescribeLocalGatewayVirtualInterfaces](#) di Referensi AWS CLI Perintah.

## describe-local-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-local-gateways`.

### AWS CLI

Untuk menggambarkan Gateway Lokal Anda

`describe-local-gateways` Contoh berikut menampilkan detail untuk gateway lokal yang tersedia untuk Anda.

```
aws ec2 describe-local-gateways
```

Output:

```
{
  "LocalGateways": [
    {
      "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
      "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0dc11b66ed59f995a",
      "OwnerId": "123456789012",
      "State": "available"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeLocalGateways](#) di Referensi AWS CLI Perintah.

## describe-managed-prefix-lists

Contoh kode berikut menunjukkan cara menggunakan `describe-managed-prefix-lists`.

### AWS CLI

Untuk mendeskripsikan daftar awalan terkelola

`describe-managed-prefix-lists` Contoh berikut menjelaskan daftar awalan yang dimiliki oleh AWS akun `123456789012`.

```
aws ec2 describe-managed-prefix-lists \
  --filters Name=owner-id,Values=123456789012
```

### Output:

```
{
  "PrefixLists": [
    {
      "PrefixListId": "pl-11223344556677aab",
      "AddressFamily": "IPv6",
      "State": "create-complete",
      "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-11223344556677aab",
      "PrefixListName": "vpc-ipv6-cidrs",
      "MaxEntries": 25,
      "Version": 1,
      "Tags": [],
    }
  ]
}
```

```
    "OwnerId": "123456789012"
  },
  {
    "PrefixListId": "pl-0123456abcabc1",
    "AddressFamily": "IPv4",
    "State": "active",
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabc1",
    "PrefixListName": "vpc-cidrs",
    "MaxEntries": 10,
    "Version": 1,
    "Tags": [],
    "OwnerId": "123456789012"
  }
]
```

Untuk informasi selengkapnya, lihat [Daftar awalan terkelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeManagedPrefixLists](#) di Referensi AWS CLI Perintah.

## describe-moving-addresses

Contoh kode berikut menunjukkan cara menggunakan `describe-moving-addresses`.

### AWS CLI

Untuk mendeskripsikan alamat bergerak Anda

Contoh ini menjelaskan semua alamat IP Elastis Anda yang bergerak.

Perintah:

```
aws ec2 describe-moving-addresses
```

Output:

```
{
  "MovingAddressStatuses": [
    {
      "PublicIp": "198.51.100.0",
      "MoveStatus": "MovingToVpc"
    }
  ]
}
```



```
]
}
```

Contoh ini menjelaskan semua alamat yang pindah ke platform EC2-VPC.

Perintah:

```
aws ec2 describe-moving-addresses --filters Name=moving-status,Values=MovingToVpc
```

- Untuk detail API, lihat [DescribeMovingAddresses](#) di Referensi AWS CLI Perintah.

## describe-nat-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-nat-gateways`.

AWS CLI

Contoh 1: Untuk menggambarkan gateway NAT publik

`describe-nat-gateways` Contoh berikut menjelaskan gateway NAT publik yang ditentukan.

```
aws ec2 describe-nat-gateways \
  --nat-gateway-id nat-01234567890abcdef
```

Output:

```
{
  "NatGateways": [
    {
      "CreateTime": "2023-08-25T01:56:51.000Z",
      "NatGatewayAddresses": [
        {
          "AllocationId": "eipalloc-0790180cd2EXAMPLE",
          "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
          "PrivateIp": "10.0.0.211",
          "PublicIp": "54.85.121.213",
          "AssociationId": "eipassoc-04d295cc9b8815b24",
          "IsPrimary": true,
          "Status": "succeeded"
        },
        {
          "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
```

```

        "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
        "PrivateIp": "10.0.0.74",
        "PublicIp": "3.211.231.218",
        "AssociationId": "eipassoc-0f96bdca17EXAMPLE",
        "IsPrimary": false,
        "Status": "succeeded"
    }
],
"NatGatewayId": "nat-01234567890abcdef",
"State": "available",
"SubnetId": "subnet-655eab5f08EXAMPLE",
"VpcId": "vpc-098eb5ef58EXAMPLE",
"Tags": [
    {
        "Key": "Name",
        "Value": "public-nat"
    }
],
"ConnectivityType": "public"
}
]
}

```

Contoh 2: Untuk menggambarkan gateway NAT pribadi

`describe-nat-gateways` Contoh berikut menjelaskan gateway NAT pribadi yang ditentukan.

```

aws ec2 describe-nat-gateways \
  --nat-gateway-id nat-1234567890abcdef0

```

Output:

```

{
  "NatGateways": [
    {
      "CreateTime": "2023-08-25T00:50:05.000Z",
      "NatGatewayAddresses": [
        {
          "NetworkInterfaceId": "eni-0065a61b324d1897a",
          "PrivateIp": "10.0.20.240",
          "IsPrimary": true,
          "Status": "succeeded"
        }
      ],
    }
  ],
}

```

```
    {
      "NetworkInterfaceId": "eni-0065a61b324d1897a",
      "PrivateIp": "10.0.20.33",
      "IsPrimary": false,
      "Status": "succeeded"
    },
    {
      "NetworkInterfaceId": "eni-0065a61b324d1897a",
      "PrivateIp": "10.0.20.197",
      "IsPrimary": false,
      "Status": "succeeded"
    }
  ],
  "NatGatewayId": "nat-1234567890abcdef0",
  "State": "available",
  "SubnetId": "subnet-08fc749671EXAMPLE",
  "VpcId": "vpc-098eb5ef58EXAMPLE",
  "Tags": [
    {
      "Key": "Name",
      "Value": "private-nat"
    }
  ],
  "ConnectivityType": "private"
}
]
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeNatGateways](#) di Referensi AWS CLI Perintah.

## describe-network-acls

Contoh kode berikut menunjukkan cara menggunakan `describe-network-acls`.

### AWS CLI

Untuk menggambarkan ACL jaringan Anda

`describe-network-acls` Contoh berikut mengambil rincian tentang ACL jaringan Anda.

```
aws ec2 describe-network-acls
```

## Output:

```
{
  "NetworkAcls": [
    {
      "Associations": [
        {
          "NetworkAclAssociationId": "aclassoc-0c1679dc41EXAMPLE",
          "NetworkAclId": "acl-0ea1f54ca7EXAMPLE",
          "SubnetId": "subnet-0931fc2fa5EXAMPLE"
        }
      ],
      "Entries": [
        {
          "CidrBlock": "0.0.0.0/0",
          "Egress": true,
          "Protocol": "-1",
          "RuleAction": "allow",
          "RuleNumber": 100
        },
        {
          "CidrBlock": "0.0.0.0/0",
          "Egress": true,
          "Protocol": "-1",
          "RuleAction": "deny",
          "RuleNumber": 32767
        },
        {
          "CidrBlock": "0.0.0.0/0",
          "Egress": false,
          "Protocol": "-1",
          "RuleAction": "allow",
          "RuleNumber": 100
        },
        {
          "CidrBlock": "0.0.0.0/0",
          "Egress": false,
          "Protocol": "-1",
          "RuleAction": "deny",
          "RuleNumber": 32767
        }
      ],
      "IsDefault": true,
      "NetworkAclId": "acl-0ea1f54ca7EXAMPLE",
    }
  ]
}
```

```
    "Tags": [],
    "VpcId": "vpc-06e4ab6c6cEXAMPLE",
    "OwnerId": "111122223333"
  },
  {
    "Associations": [],
    "Entries": [
      {
        "CidrBlock": "0.0.0.0/0",
        "Egress": true,
        "Protocol": "-1",
        "RuleAction": "allow",
        "RuleNumber": 100
      },
      {
        "Egress": true,
        "Ipv6CidrBlock": ":::/0",
        "Protocol": "-1",
        "RuleAction": "allow",
        "RuleNumber": 101
      },
      {
        "CidrBlock": "0.0.0.0/0",
        "Egress": true,
        "Protocol": "-1",
        "RuleAction": "deny",
        "RuleNumber": 32767
      },
      {
        "Egress": true,
        "Ipv6CidrBlock": ":::/0",
        "Protocol": "-1",
        "RuleAction": "deny",
        "RuleNumber": 32768
      },
      {
        "CidrBlock": "0.0.0.0/0",
        "Egress": false,
        "Protocol": "-1",
        "RuleAction": "allow",
        "RuleNumber": 100
      },
      {
        "Egress": false,
```

```

        "Ipv6CidrBlock": "::/0",
        "Protocol": "-1",
        "RuleAction": "allow",
        "RuleNumber": 101
    },
    {
        "CidrBlock": "0.0.0.0/0",
        "Egress": false,
        "Protocol": "-1",
        "RuleAction": "deny",
        "RuleNumber": 32767
    },
    {
        "Egress": false,
        "Ipv6CidrBlock": "::/0",
        "Protocol": "-1",
        "RuleAction": "deny",
        "RuleNumber": 32768
    }
],
    "IsDefault": true,
    "NetworkAclId": "acl-0e2a78e4e2EXAMPLE",
    "Tags": [],
    "VpcId": "vpc-03914afb3eEXAMPLE",
    "OwnerId": "111122223333"
}
]
}

```

Untuk informasi selengkapnya, lihat [ACL Jaringan](#) di Panduan AWS Pengguna VPC.

- Untuk detail API, lihat [DescribeNetworkAcls](#) di Referensi AWS CLI Perintah.

## **describe-network-insights-access-scope-analyses**

Contoh kode berikut menunjukkan cara menggunakan `describe-network-insights-access-scope-analyses`.

### AWS CLI

Untuk menggambarkan analisis cakupan akses Network Insights

`describe-network-insights-access-scope-analyses` Contoh berikut menjelaskan analisis cakupan akses di AWS akun Anda.

```
aws ec2 describe-network-insights-access-scope-analyses \
  --region us-east-1
```

Output:

```
{
  "NetworkInsightsAccessScopeAnalyses": [
    {
      "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789111",
      "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope-analysis/nisa-123456789111",
      "NetworkInsightsAccessScopeId": "nis-123456789222",
      "Status": "succeeded",
      "StartDate": "2022-01-25T19:45:36.842000+00:00",
      "FindingsFound": "true",
      "Tags": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [DescribeNetworkInsightsAccessScopeAnalyses](#) di Referensi AWS CLI Perintah.

## `describe-network-insights-access-scopes`

Contoh kode berikut menunjukkan cara menggunakan `describe-network-insights-access-scopes`.

AWS CLI

Untuk menggambarkan cakupan akses Network Insights

`describe-network-insights-access-scopes` Contoh berikut menjelaskan analisis cakupan akses di akun Anda. AWS

```
aws ec2 describe-network-insights-access-scopes \
```

```
--region us-east-1
```

Output:

```
{
  "NetworkInsightsAccessScopes": [
    {
      "NetworkInsightsAccessScopeId": "nis-123456789111",
      "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope/nis-123456789111",
      "CreateDate": "2021-11-29T21:12:41.416000+00:00",
      "UpdatedDate": "2021-11-29T21:12:41.416000+00:00",
      "Tags": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [DescribeNetworkInsightsAccessScopes](#) di Referensi AWS CLI Perintah.

## describe-network-insights-analyses

Contoh kode berikut menunjukkan cara menggunakan `describe-network-insights-analyses`.

### AWS CLI

Untuk melihat hasil analisis jalur

`describe-network-insights-analyses` Contoh berikut menjelaskan analisis yang ditentukan. Dalam contoh ini, sumbernya adalah gateway internet, tujuannya adalah instance EC2, dan protokolnya adalah TCP. Analisis berhasil (`Status` `succeeded`) dan jalurnya tidak dapat dijangkau (`NetworkPathFound` `is` `false`). Kode penjelasan `ENI_SG_RULES_MISMATCH` menunjukkan bahwa grup keamanan untuk instance tidak berisi aturan yang memungkinkan lalu lintas di port tujuan.

```
aws ec2 describe-network-insights-analyses \
  --network-insights-analysis-ids nia-02207aa13eb480c7a
```

Output:



```
{
  "NetworkInsightsAnalyses": [
    {
      "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a",
      "NetworkInsightsAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-analysis/nia-02207aa13eb480c7a",
      "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
      "StartDate": "2021-01-20T22:58:37.495Z",
      "Status": "succeeded",
      "NetworkPathFound": false,
      "Explanations": [
        {
          "Direction": "ingress",
          "ExplanationCode": "ENI_SG_RULES_MISMATCH",
          "NetworkInterface": {
            "Id": "eni-0a25edef15a6cc08c",
            "Arn": "arn:aws:ec2:us-east-1:123456789012:network-
interface/eni-0a25edef15a6cc08c"
          },
          "SecurityGroups": [
            {
              "Id": "sg-02f0d35a850ba727f",
              "Arn": "arn:aws:ec2:us-east-1:123456789012:security-
group/sg-02f0d35a850ba727f"
            }
          ],
          "Subnet": {
            "Id": "subnet-004ff41eccb4d1194",
            "Arn": "arn:aws:ec2:us-east-1:123456789012:subnet/
subnet-004ff41eccb4d1194"
          },
          "Vpc": {
            "Id": "vpc-f1663d98ad28331c7",
            "Arn": "arn:aws:ec2:us-east-1:123456789012:vpc/vpc-
f1663d98ad28331c7"
          }
        }
      ],
      "Tags": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai menggunakan AWS CLI](#) di Panduan Reachability Analyzer.

- Untuk detail API, lihat [DescribeNetworkInsightsAnalyses](#) di Referensi AWS CLI Perintah.

## describe-network-insights-paths

Contoh kode berikut menunjukkan cara menggunakan `describe-network-insights-paths`.

### AWS CLI

Untuk menggambarkan jalan

`describe-network-insights-paths` Contoh berikut menjelaskan jalur yang ditentukan.

```
aws ec2 describe-network-insights-paths \
  --network-insights-path-ids nip-0b26f224f1d131fa8
```

Output:

```
{
  "NetworkInsightsPaths": [
    {
      "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
      "NetworkInsightsPathArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-path/nip-0b26f224f1d131fa8",
      "CreateDate": "2021-01-20T22:43:46.933Z",
      "Source": "igw-0797cccdc9d73b0e5",
      "Destination": "i-0495d385ad28331c7",
      "Protocol": "tcp"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai menggunakan AWS CLI](#) di Panduan Reachability Analyzer.

- Untuk detail API, lihat [DescribeNetworkInsightsPaths](#) di Referensi AWS CLI Perintah.

## describe-network-interface-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-network-interface-attribute`.

### AWS CLI

Untuk menggambarkan atribut lampiran dari antarmuka jaringan

Perintah contoh ini menjelaskan attachment atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --attribute attachment
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Attachment": {
    "Status": "attached",
    "DeviceIndex": 0,
    "AttachTime": "2015-05-21T20:02:20.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "DeleteOnTermination": true,
    "AttachmentId": "eni-attach-43348162",
    "InstanceOwnerId": "123456789012"
  }
}
```

Untuk menggambarkan atribut deskripsi antarmuka jaringan

Perintah contoh ini menjelaskan `description` atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --attribute description
```

Output:

```
{
```

```
"NetworkInterfaceId": "eni-686ea200",
  "Description": {
    "Value": "My description"
  }
}
```

Untuk menggambarkan atribut GroupSet dari antarmuka jaringan

Perintah contoh ini menjelaskan groupSet atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute groupSet
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Groups": [
    {
      "GroupName": "my-security-group",
      "GroupId": "sg-903004f8"
    }
  ]
}
```

Untuk menggambarkan sourceDestCheck atribut antarmuka jaringan

Perintah contoh ini menjelaskan sourceDestCheck atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute sourceDestCheck
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "SourceDestCheck": {
    "Value": true
  }
}
```

```
}  
}
```

- Untuk detail API, lihat [DescribeNetworkInterfaceAttributes](#) di Referensi AWS CLI Perintah.

## describe-network-interface-permissions

Contoh kode berikut menunjukkan cara menggunakan `describe-network-interface-permissions`.

### AWS CLI

Untuk mendeskripsikan izin antarmuka jaringan Anda

Contoh ini menjelaskan semua izin antarmuka jaringan Anda.

Perintah:

```
aws ec2 describe-network-interface-permissions
```

Output:

```
{  
  "NetworkInterfacePermissions": [  
    {  
      "PermissionState": {  
        "State": "GRANTED"  
      },  
      "NetworkInterfacePermissionId": "eni-perm-06fd19020ede149ea",  
      "NetworkInterfaceId": "eni-b909511a",  
      "Permission": "INSTANCE-ATTACH",  
      "AwsAccountId": "123456789012"  
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeNetworkInterfacePermissions](#) di Referensi AWS CLI Perintah.

## describe-network-interfaces

Contoh kode berikut menunjukkan cara menggunakan `describe-network-interfaces`.

## AWS CLI

Untuk mendeskripsikan antarmuka jaringan Anda

Contoh ini menjelaskan semua antarmuka jaringan Anda.

Perintah:

```
aws ec2 describe-network-interfaces
```

Output:

```
{
  "NetworkInterfaces": [
    {
      "Status": "in-use",
      "MacAddress": "02:2f:8f:b0:cf:75",
      "SourceDestCheck": true,
      "VpcId": "vpc-a01106c2",
      "Description": "my network interface",
      "Association": {
        "PublicIp": "203.0.113.12",
        "AssociationId": "eipassoc-0fbb766a",
        "PublicDnsName": "ec2-203-0-113-12.compute-1.amazonaws.com",
        "IpOwnerId": "123456789012"
      },
      "NetworkInterfaceId": "eni-e5aa89a3",
      "PrivateIpAddresses": [
        {
          "PrivateDnsName": "ip-10-0-1-17.ec2.internal",
          "Association": {
            "PublicIp": "203.0.113.12",
            "AssociationId": "eipassoc-0fbb766a",
            "PublicDnsName": "ec2-203-0-113-12.compute-1.amazonaws.com",
            "IpOwnerId": "123456789012"
          },
          "Primary": true,
          "PrivateIpAddress": "10.0.1.17"
        }
      ],
      "RequesterManaged": false,
      "Ipv6Addresses": [],
      "PrivateDnsName": "ip-10-0-1-17.ec2.internal",
      "AvailabilityZone": "us-east-1d",
```

```
"Attachment": {
  "Status": "attached",
  "DeviceIndex": 1,
  "AttachTime": "2013-11-30T23:36:42.000Z",
  "InstanceId": "i-1234567890abcdef0",
  "DeleteOnTermination": false,
  "AttachmentId": "eni-attach-66c4350a",
  "InstanceOwnerId": "123456789012"
},
"Groups": [
  {
    "GroupName": "default",
    "GroupId": "sg-8637d3e3"
  }
],
"SubnetId": "subnet-b61f49f0",
"OwnerId": "123456789012",
"TagSet": [],
"PrivateIpAddress": "10.0.1.17"
},
{
  "Status": "in-use",
  "MacAddress": "02:58:f5:ef:4b:06",
  "SourceDestCheck": true,
  "VpcId": "vpc-a01106c2",
  "Description": "Primary network interface",
  "Association": {
    "PublicIp": "198.51.100.0",
    "IpOwnerId": "amazon"
  },
  "NetworkInterfaceId": "eni-f9ba99bf",
  "PrivateIpAddresses": [
    {
      "Association": {
        "PublicIp": "198.51.100.0",
        "IpOwnerId": "amazon"
      },
      "Primary": true,
      "PrivateIpAddress": "10.0.1.149"
    }
  ],
  "RequesterManaged": false,
  "Ipv6Addresses": [],
  "AvailabilityZone": "us-east-1d",
```

```

    "Attachment": {
      "Status": "attached",
      "DeviceIndex": 0,
      "AttachTime": "2013-11-30T23:35:33.000Z",
      "InstanceId": "i-0598c7d356eba48d7",
      "DeleteOnTermination": true,
      "AttachmentId": "eni-attach-1b9db777",
      "InstanceOwnerId": "123456789012"
    },
    "Groups": [
      {
        "GroupName": "default",
        "GroupId": "sg-8637d3e3"
      }
    ],
    "SubnetId": "subnet-b61f49f0",
    "OwnerId": "123456789012",
    "TagSet": [],
    "PrivateIpAddress": "10.0.1.149"
  }
]
}

```

Contoh ini menjelaskan antarmuka jaringan yang memiliki tag dengan kunci Purpose dan nilainyaProd.

Perintah:

```
aws ec2 describe-network-interfaces --filters Name=tag:Purpose,Values=Prod
```

Output:

```

{
  "NetworkInterfaces": [
    {
      "Status": "available",
      "MacAddress": "12:2c:bd:f9:bf:17",
      "SourceDestCheck": true,
      "VpcId": "vpc-8941ebec",
      "Description": "ProdENI",
      "NetworkInterfaceId": "eni-b9a5ac93",
      "PrivateAddresses": [

```



```

        {
            "PrivateDnsName": "ip-10-0-1-55.ec2.internal",
            "Primary": true,
            "PrivateIpAddress": "10.0.1.55"
        },
        {
            "PrivateDnsName": "ip-10-0-1-117.ec2.internal",
            "Primary": false,
            "PrivateIpAddress": "10.0.1.117"
        }
    ],
    "RequesterManaged": false,
    "PrivateDnsName": "ip-10-0-1-55.ec2.internal",
    "AvailabilityZone": "us-east-1d",
    "Ipv6Addresses": [],
    "Groups": [
        {
            "GroupName": "MySG",
            "GroupId": "sg-905002f5"
        }
    ],
    "SubnetId": "subnet-31d6c219",
    "OwnerId": "123456789012",
    "TagSet": [
        {
            "Value": "Prod",
            "Key": "Purpose"
        }
    ],
    "PrivateIpAddress": "10.0.1.55"
}
]
}

```

- Untuk detail API, lihat [DescribeNetworkInterfaces](#) di Referensi AWS CLI Perintah.

## describe-placement-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-placement-groups`.

### AWS CLI

Untuk menggambarkan grup penempatan Anda

Perintah contoh ini menjelaskan semua grup penempatan Anda.

Perintah:

```
aws ec2 describe-placement-groups
```

Output:

```
{
  "PlacementGroups": [
    {
      "GroupName": "my-cluster",
      "State": "available",
      "Strategy": "cluster"
    },
    ...
  ]
}
```

- Untuk detail API, lihat [DescribePlacementGroups](#) di Referensi AWS CLI Perintah.

## **describe-prefix-lists**

Contoh kode berikut menunjukkan cara menggunakan `describe-prefix-lists`.

AWS CLI

Untuk menggambarkan daftar awalan

Contoh ini mencantumkan semua daftar awalan yang tersedia untuk wilayah tersebut.

Perintah:

```
aws ec2 describe-prefix-lists
```

Output:

```
{
  "PrefixLists": [
    {
      "PrefixListName": "com.amazonaws.us-east-1.s3",
      "Cidrs": [
```

```

    "54.231.0.0/17"
  ],
  "PrefixListId": "pl-63a5400a"
}
]
}

```

- Untuk detail API, lihat [DescribePrefixLists](#) di Referensi AWS CLI Perintah.

## describe-principal-id-format

Contoh kode berikut menunjukkan cara menggunakan `describe-principal-id-format`.

### AWS CLI

Untuk mendeskripsikan format ID untuk pengguna IAM dan peran dengan format ID panjang diaktifkan

`describe-principal-id-format` Contoh berikut menjelaskan format ID untuk pengguna root, semua peran IAM, dan semua pengguna IAM dengan format ID panjang diaktifkan.

```

aws ec2 describe-principal-id-format \
  --resource instance

```

### Output:

```

{
  "Principals": [
    {
      "Arn": "arn:aws:iam::123456789012:root",
      "Statuses": [
        {
          "Deadline": "2016-12-15T00:00:00.000Z",
          "Resource": "reservation",
          "UseLongIds": true
        },
        {
          "Deadline": "2016-12-15T00:00:00.000Z",
          "Resource": "instance",
          "UseLongIds": true
        }
      ]
    }
  ]
}

```

```
        "Deadline": "2016-12-15T00:00:00.000Z",
        "Resource": "volume",
        "UseLongIds": true
      },
    ]
  },
  ...
]
```

- Untuk detail API, lihat [DescribePrincipalIdFormat](#) di Referensi AWS CLI Perintah.

## describe-public-ipv4-pools

Contoh kode berikut menunjukkan cara menggunakan `describe-public-ipv4-pools`.

### AWS CLI

Untuk menggambarkan kumpulan alamat IPv4 publik Anda

`describe-public-ipv4-pools` Contoh berikut menampilkan detail tentang kumpulan alamat yang dibuat saat Anda menyediakan rentang alamat IPv4 publik menggunakan Bring Your Own IP Address (BYOIP).

```
aws ec2 describe-public-ipv4-pools
```

Output:

```
{
  "PublicIpv4Pools": [
    {
      "PoolId": "ipv4pool-ec2-1234567890abcdef0",
      "PoolAddressRanges": [
        {
          "FirstAddress": "203.0.113.0",
          "LastAddress": "203.0.113.255",
          "AddressCount": 256,
          "AvailableAddressCount": 256
        }
      ],
      "TotalAddressCount": 256,
      "TotalAvailableAddressCount": 256
    }
  ]
}
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribePublicIpv4Pools](#) di AWS CLI Command Reference.

## describe-regions

Contoh kode berikut menunjukkan cara menggunakan `describe-regions`.

### AWS CLI

Contoh 1: Untuk menjelaskan semua Wilayah yang diaktifkan

Contoh `describe-regions` berikut menjelaskan semua Wilayah yang diaktifkan untuk akun Anda.

```
aws ec2 describe-regions
```

Output:

```
{  
  "Regions": [  
    {  
      "Endpoint": "ec2.eu-north-1.amazonaws.com",  
      "RegionName": "eu-north-1",  
      "OptInStatus": "opt-in-not-required"  
    },  
    {  
      "Endpoint": "ec2.ap-south-1.amazonaws.com",  
      "RegionName": "ap-south-1",  
      "OptInStatus": "opt-in-not-required"  
    },  
    {  
      "Endpoint": "ec2.eu-west-3.amazonaws.com",  
      "RegionName": "eu-west-3",  
      "OptInStatus": "opt-in-not-required"  
    },  
    {  
      "Endpoint": "ec2.eu-west-2.amazonaws.com",  
      "RegionName": "eu-west-2",  
      "OptInStatus": "opt-in-not-required"  
    }  
  ]  
}
```

```
  },
  {
    "Endpoint": "ec2.eu-west-1.amazonaws.com",
    "RegionName": "eu-west-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-3.amazonaws.com",
    "RegionName": "ap-northeast-3",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-2.amazonaws.com",
    "RegionName": "ap-northeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-1.amazonaws.com",
    "RegionName": "ap-northeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.sa-east-1.amazonaws.com",
    "RegionName": "sa-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ca-central-1.amazonaws.com",
    "RegionName": "ca-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-southeast-1.amazonaws.com",
    "RegionName": "ap-southeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-southeast-2.amazonaws.com",
    "RegionName": "ap-southeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-central-1.amazonaws.com",
    "RegionName": "eu-central-1",
```

```

    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-1.amazonaws.com",
    "RegionName": "us-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-2.amazonaws.com",
    "RegionName": "us-east-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-west-1.amazonaws.com",
    "RegionName": "us-west-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-west-2.amazonaws.com",
    "RegionName": "us-west-2",
    "OptInStatus": "opt-in-not-required"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Wilayah dan Zona](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menjelaskan Wilayah yang diaktifkan dengan titik akhir yang namanya berisi string tertentu

Contoh `describe-regions` berikut menjelaskan semua Wilayah yang telah Anda aktifkan yang memiliki string “us” di titik akhir.

```

aws ec2 describe-regions \
  --filters "Name=endpoint,Values=*us*"

```

Output:

```

{
  "Regions": [
    {
      "Endpoint": "ec2.us-east-1.amazonaws.com",
      "RegionName": "us-east-1"
    }
  ]
}

```

```
    },
    {
      "Endpoint": "ec2.us-east-2.amazonaws.com",
      "RegionName": "us-east-2"
    },
    {
      "Endpoint": "ec2.us-west-1.amazonaws.com",
      "RegionName": "us-west-1"
    },
    {
      "Endpoint": "ec2.us-west-2.amazonaws.com",
      "RegionName": "us-west-2"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Wilayah dan Zona](#) di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk menjelaskan semua Wilayah

Contoh `describe-regions` berikut menjelaskan semua Wilayah yang tersedia, termasuk Wilayah yang dinonaktifkan.

```
aws ec2 describe-regions \
  --all-regions
```

Output:

```
{
  "Regions": [
    {
      "Endpoint": "ec2.eu-north-1.amazonaws.com",
      "RegionName": "eu-north-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.ap-south-1.amazonaws.com",
      "RegionName": "ap-south-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-3.amazonaws.com",
      "RegionName": "eu-west-3",

```



```
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-west-2.amazonaws.com",
    "RegionName": "eu-west-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-west-1.amazonaws.com",
    "RegionName": "eu-west-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-3.amazonaws.com",
    "RegionName": "ap-northeast-3",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.me-south-1.amazonaws.com",
    "RegionName": "me-south-1",
    "OptInStatus": "not-opted-in"
  },
  {
    "Endpoint": "ec2.ap-northeast-2.amazonaws.com",
    "RegionName": "ap-northeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-1.amazonaws.com",
    "RegionName": "ap-northeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.sa-east-1.amazonaws.com",
    "RegionName": "sa-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ca-central-1.amazonaws.com",
    "RegionName": "ca-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-east-1.amazonaws.com",
```

```
    "RegionName": "ap-east-1",
    "OptInStatus": "not-opted-in"
  },
  {
    "Endpoint": "ec2.ap-southeast-1.amazonaws.com",
    "RegionName": "ap-southeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-southeast-2.amazonaws.com",
    "RegionName": "ap-southeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-central-1.amazonaws.com",
    "RegionName": "eu-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-1.amazonaws.com",
    "RegionName": "us-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-2.amazonaws.com",
    "RegionName": "us-east-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-west-1.amazonaws.com",
    "RegionName": "us-west-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-west-2.amazonaws.com",
    "RegionName": "us-west-2",
    "OptInStatus": "opt-in-not-required"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Wilayah dan Zona](#) di Panduan Pengguna Amazon EC2.

Contoh 4: Untuk mencantumkan nama Wilayah saja

Contoh `describe-regions` berikut menggunakan parameter `--query` untuk memfilter output dan hanya mengembalikan nama Wilayah sebagai teks.

```
aws ec2 describe-regions \  
  --all-regions \  
  --query "Regions[].{Name:RegionName}" \  
  --output text
```

Output:

```
eu-north-1  
ap-south-1  
eu-west-3  
eu-west-2  
eu-west-1  
ap-northeast-3  
ap-northeast-2  
me-south-1  
ap-northeast-1  
sa-east-1  
ca-central-1  
ap-east-1  
ap-southeast-1  
ap-southeast-2  
eu-central-1  
us-east-1  
us-east-2  
us-west-1  
us-west-2
```

Untuk informasi selengkapnya, lihat [Wilayah dan Zona](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeRegions](#) di Referensi AWS CLI Perintah.

## **describe-replace-root-volume-tasks**

Contoh kode berikut menunjukkan cara menggunakan `describe-replace-root-volume-tasks`.

### AWS CLI

Contoh 1: Untuk melihat informasi tentang tugas penggantian volume root tertentu

`describe-replace-root-volume-tasks` Contoh berikut menjelaskan tugas penggantian volume root `replacevol-0111122223333abcd`.

```
aws ec2 describe-replace-root-volume-tasks \  
  --replace-root-volume-task-ids replacevol-0111122223333abcd
```

Output:

```
{  
  "ReplaceRootVolumeTasks": [  
    {  
      "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",  
      "Tags": [],  
      "InstanceId": "i-0123456789abcdefa",  
      "TaskState": "succeeded",  
      "StartTime": "2022-03-14T15:16:28Z",  
      "CompleteTime": "2022-03-14T15:16:52Z"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengganti volume root](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

Contoh 2: Untuk melihat informasi tentang semua tugas penggantian volume root untuk instance tertentu

`describe-replace-root-volume-tasks` Contoh berikut menjelaskan semua tugas penggantian volume root misalnya `i-0123456789abcdefa`.

```
aws ec2 describe-replace-root-volume-tasks \  
  --filters Name=instance-id,Values=i-0123456789abcdefa
```

Output:

```
{  
  "ReplaceRootVolumeTasks": [  
    {  
      "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",  
      "Tags": [],  
      "InstanceId": "i-0123456789abcdefa",  
      "TaskState": "succeeded",  
      "StartTime": "2022-03-14T15:16:28Z",  
      "CompleteTime": "2022-03-14T15:16:52Z"  
    }  
  ]  
}
```

```
    "StartTime": "2022-03-14T15:06:38Z",
    "CompleteTime": "2022-03-14T15:07:03Z"
  },
  {
    "ReplaceRootVolumeTaskId": "replacevol-044445555555abcd",
    "Tags": [],
    "InstanceId": "i-0123456789abcdefa",
    "TaskState": "succeeded",
    "StartTime": "2022-03-14T15:16:28Z",
    "CompleteTime": "2022-03-14T15:16:52Z"
  }
]
```

Untuk informasi selengkapnya, lihat [Mengganti volume root](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [DescribeReplaceRootVolumeTasks](#) di Referensi AWS CLI Perintah.

## **describe-reserved-instances-listings**

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-instances-listings`.

### AWS CLI

Untuk mendeskripsikan daftar Instans Cadangan

`describe-reserved-instances-listings` Contoh berikut mengambil informasi tentang daftar Instans Cadangan yang ditentukan.

```
aws ec2 describe-reserved-instances-listings \
  --reserved-instances-listing-id 5ec28771-05ff-4b9b-aa31-9e57dexample
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DescribeReservedInstancesListings](#) di Referensi AWS CLI Perintah.

## **describe-reserved-instances-modifications**

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-instances-modifications`.

## AWS CLI

Untuk menjelaskan modifikasi Instans Cadangan

Perintah contoh ini menjelaskan semua permintaan modifikasi Instans Cadangan yang telah dikirimkan untuk akun Anda.

Perintah:

```
aws ec2 describe-reserved-instances-modifications
```

Output:

```
{
  "ReservedInstancesModifications": [
    {
      "Status": "fulfilled",
      "ModificationResults": [
        {
          "ReservedInstancesId": "93bbbca2-62f1-4d9d-b225-16bada29e6c7",
          "TargetConfiguration": {
            "AvailabilityZone": "us-east-1b",
            "InstanceType": "m1.large",
            "InstanceCount": 3
          }
        },
        {
          "ReservedInstancesId": "1ba8e2e3-aabb-46c3-bcf5-3fe2fda922e6",
          "TargetConfiguration": {
            "AvailabilityZone": "us-east-1d",
            "InstanceType": "m1.xlarge",
            "InstanceCount": 1
          }
        }
      ],
      "EffectiveDate": "2015-08-12T17:00:00.000Z",
      "CreateDate": "2015-08-12T17:52:52.630Z",
      "UpdateDate": "2015-08-12T18:08:06.698Z",
      "ClientToken": "c9adb218-3222-4889-8216-0cf0e52dc37e",
      "ReservedInstancesModificationId": "rimod-d3ed4335-b1d3-4de6-ab31-0f13aaf46687",
      "ReservedInstancesIds": [
        {
```

```

    "ReservedInstancesId": "b847fa93-e282-4f55-b59a-1342f5bd7c02"
  }
]
}

```

- Untuk detail API, lihat [DescribeReservedInstancesModifications](#) di Referensi AWS CLI Perintah.

## describe-reserved-instances-offerings

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-instances-offerings`.

### AWS CLI

Untuk menggambarkan penawaran Instans Cadangan

Perintah contoh ini menjelaskan semua Instans Cadangan yang tersedia untuk pembelian di wilayah tersebut.

Perintah:

```
aws ec2 describe-reserved-instances-offerings
```

Output:

```

{
  "ReservedInstancesOfferings": [
    {
      "OfferingType": "Partial Upfront",
      "AvailabilityZone": "us-east-1b",
      "InstanceTenancy": "default",
      "PricingDetails": [],
      "ProductDescription": "Red Hat Enterprise Linux",
      "UsagePrice": 0.0,
      "RecurringCharges": [
        {
          "Amount": 0.088,
          "Frequency": "Hourly"
        }
      ],
    }
  ],
}

```

```

    "Marketplace": false,
    "CurrencyCode": "USD",
    "FixedPrice": 631.0,
    "Duration": 94608000,
    "ReservedInstancesOfferingId": "9a06095a-bdc6-47fe-a94a-2a382f016040",
    "InstanceType": "c1.medium"
  },
  {
    "OfferingType": "PartialUpfront",
    "AvailabilityZone": "us-east-1b",
    "InstanceTenancy": "default",
    "PricingDetails": [],
    "ProductDescription": "Linux/UNIX",
    "UsagePrice": 0.0,
    "RecurringCharges": [
      {
        "Amount": 0.028,
        "Frequency": "Hourly"
      }
    ],
    "Marketplace": false,
    "CurrencyCode": "USD",
    "FixedPrice": 631.0,
    "Duration": 94608000,
    "ReservedInstancesOfferingId": "bfbefc6c-0d10-418d-b144-7258578d329d",
    "InstanceType": "c1.medium"
  },
  ...
}

```

Untuk menjelaskan penawaran Instans Cadangan Anda menggunakan opsi

Contoh ini mencantumkan Instans Cadangan yang ditawarkan AWS dengan spesifikasi berikut: jenis instans t1.micro, produk Windows (Amazon VPC), dan penawaran Pemanfaatan Berat.

Perintah:

```
aws ec2 describe-reserved-instances-offerings --no-include-marketplace --instance-type "t1.micro" --product-description "Windows (Amazon VPC)" --offering-type "no upfront"
```

Output:



```
{
  "ReservedInstancesOfferings": [
    {
      "OfferingType": "No Upfront",
      "AvailabilityZone": "us-east-1b",
      "InstanceTenancy": "default",
      "PricingDetails": [],
      "ProductDescription": "Windows",
      "UsagePrice": 0.0,
      "RecurringCharges": [
        {
          "Amount": 0.015,
          "Frequency": "Hourly"
        }
      ],
      "Marketplace": false,
      "CurrencyCode": "USD",
      "FixedPrice": 0.0,
      "Duration": 31536000,
      "ReservedInstancesOfferingId": "c48ab04c-fe69-4f94-8e39-a23842292823",
      "InstanceType": "t1.micro"
    },
    ...
    {
      "OfferingType": "No Upfront",
      "AvailabilityZone": "us-east-1d",
      "InstanceTenancy": "default",
      "PricingDetails": [],
      "ProductDescription": "Windows (Amazon VPC)",
      "UsagePrice": 0.0,
      "RecurringCharges": [
        {
          "Amount": 0.015,
          "Frequency": "Hourly"
        }
      ],
      "Marketplace": false,
      "CurrencyCode": "USD",
      "FixedPrice": 0.0,
      "Duration": 31536000,
      "ReservedInstancesOfferingId": "3a98bf7d-2123-42d4-b4f5-8dbec4b06dc6",
      "InstanceType": "t1.micro"
    }
  ]
}
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeReservedInstancesOfferings](#) di Referensi AWS CLI Perintah.

## describe-reserved-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-instances`.

### AWS CLI

Untuk mendeskripsikan Instans Cadangan Anda

Perintah contoh ini menjelaskan Instans Cadangan yang Anda miliki.

Perintah:

```
aws ec2 describe-reserved-instances
```

Output:

```
{  
  "ReservedInstances": [  
    {  
      "ReservedInstancesId": "b847fa93-e282-4f55-b59a-1342fexample",  
      "OfferingType": "No Upfront",  
      "AvailabilityZone": "us-west-1c",  
      "End": "2016-08-14T21:34:34.000Z",  
      "ProductDescription": "Linux/UNIX",  
      "UsagePrice": 0.00,  
      "RecurringCharges": [  
        {  
          "Amount": 0.104,  
          "Frequency": "Hourly"  
        }  
      ],  
      "Start": "2015-08-15T21:34:35.086Z",  
      "State": "active",  
      "FixedPrice": 0.0,  
      "CurrencyCode": "USD",  
      "Duration": 31536000,  
    }  
  ]  
}
```

```

        "InstanceTenancy": "default",
        "InstanceType": "m3.medium",
        "InstanceCount": 2
    },
    ...
]
}

```

Untuk mendeskripsikan Instans Cadangan Anda menggunakan filter

Contoh ini memfilter respons untuk menyertakan hanya tiga tahun, t2.micro Linux/Unix Reserved Instances di us-west-1c.

Perintah:

```

aws ec2 describe-reserved-instances --filters Name=duration,Values=94608000
Name=instance-type,Values=t2.micro Name=product-description,Values=Linux/UNIX
Name=availability-zone,Values=us-east-1e

```

Output:

```

{
  "ReservedInstances": [
    {
      "ReservedInstancesId": "f127bd27-edb7-44c9-a0eb-0d7e09259af0",
      "OfferingType": "All Upfront",
      "AvailabilityZone": "us-east-1e",
      "End": "2018-03-26T21:34:34.000Z",
      "ProductDescription": "Linux/UNIX",
      "UsagePrice": 0.00,
      "RecurringCharges": [],
      "Start": "2015-03-27T21:34:35.848Z",
      "State": "active",
      "FixedPrice": 151.0,
      "CurrencyCode": "USD",
      "Duration": 94608000,
      "InstanceTenancy": "default",
      "InstanceType": "t2.micro",
      "InstanceCount": 1
    }
  ]
}

```

Untuk informasi selengkapnya, lihat Menggunakan Instans Amazon EC2 di Panduan Pengguna Antarmuka Baris Perintah AWS .

- Untuk detail API, lihat [DescribeReservedInstances](#) di Referensi AWS CLI Perintah.

## describe-route-tables

Contoh kode berikut menunjukkan cara menggunakan `describe-route-tables`.

### AWS CLI

Untuk menggambarkan tabel rute Anda

`describe-route-tables` Contoh berikut mengambil rincian tentang tabel rute Anda

```
aws ec2 describe-route-tables
```

Output:

```
{
  "RouteTables": [
    {
      "Associations": [
        {
          "Main": true,
          "RouteTableAssociationId": "rtbassoc-0df3f54e06EXAMPLE",
          "RouteTableId": "rtb-09ba434c1bEXAMPLE"
        }
      ],
      "PropagatingVgws": [],
      "RouteTableId": "rtb-09ba434c1bEXAMPLE",
      "Routes": [
        {
          "DestinationCidrBlock": "10.0.0.0/16",
          "GatewayId": "local",
          "Origin": "CreateRouteTable",
          "State": "active"
        },
        {
          "DestinationCidrBlock": "0.0.0.0/0",
          "NatGatewayId": "nat-06c018cbd8EXAMPLE",
          "Origin": "CreateRoute",
          "State": "blackhole"
        }
      ]
    }
  ]
}
```

```
    }
  ],
  "Tags": [],
  "VpcId": "vpc-0065acced4EXAMPLE",
  "OwnerId": "111122223333"
},
{
  "Associations": [
    {
      "Main": true,
      "RouteTableAssociationId": "rtbassoc-9EXAMPLE",
      "RouteTableId": "rtb-a1eec7de"
    }
  ],
  "PropagatingVgws": [],
  "RouteTableId": "rtb-a1eec7de",
  "Routes": [
    {
      "DestinationCidrBlock": "172.31.0.0/16",
      "GatewayId": "local",
      "Origin": "CreateRouteTable",
      "State": "active"
    },
    {
      "DestinationCidrBlock": "0.0.0.0/0",
      "GatewayId": "igw-fEXAMPLE",
      "Origin": "CreateRoute",
      "State": "active"
    }
  ],
  "Tags": [],
  "VpcId": "vpc-3EXAMPLE",
  "OwnerId": "111122223333"
},
{
  "Associations": [
    {
      "Main": false,
      "RouteTableAssociationId": "rtbassoc-0b100c28b2EXAMPLE",
      "RouteTableId": "rtb-07a98f76e5EXAMPLE",
      "SubnetId": "subnet-0d3d002af8EXAMPLE"
    }
  ],
  "PropagatingVgws": [],
```

```
"RouteTableId": "rtb-07a98f76e5EXAMPLE",
"Routes": [
  {
    "DestinationCidrBlock": "10.0.0.0/16",
    "GatewayId": "local",
    "Origin": "CreateRouteTable",
    "State": "active"
  },
  {
    "DestinationCidrBlock": "0.0.0.0/0",
    "GatewayId": "igw-06cf664d80EXAMPLE",
    "Origin": "CreateRoute",
    "State": "active"
  }
],
"Tags": [],
"VpcId": "vpc-0065acced4EXAMPLE",
"OwnerId": "111122223333"
}
]
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tabel Rute](#) di Panduan Pengguna AWS VPC.

- Untuk detail API, lihat [DescribeRouteTables](#) di Referensi AWS CLI Perintah.

## **describe-scheduled-instance-availability**

Contoh kode berikut menunjukkan cara menggunakan `describe-scheduled-instance-availability`.

### AWS CLI

Untuk menggambarkan jadwal yang tersedia

Contoh ini menjelaskan jadwal yang terjadi setiap minggu pada hari Minggu, dimulai pada tanggal yang ditentukan.

Perintah:

```
aws ec2 describe-scheduled-instance-availability --recurrence
Frequency=Weekly,Interval=1,OccurrenceDays=[1] --first-slot-start-time-range
EarliestTime=2016-01-31T00:00:00Z,LatestTime=2016-01-31T04:00:00Z
```

### Output:

```
{
  "ScheduledInstanceAvailabilitySet": [
    {
      "AvailabilityZone": "us-west-2b",
      "TotalScheduledInstanceHours": 1219,
      "PurchaseToken": "eyJ2IjoiMSIsInMiOiJEsImMiOi...",
      "MinTermDurationInDays": 366,
      "AvailableInstanceCount": 20,
      "Recurrence": {
        "OccurrenceDaySet": [
          1
        ],
        "Interval": 1,
        "Frequency": "Weekly",
        "OccurrenceRelativeToEnd": false
      },
      "Platform": "Linux/UNIX",
      "FirstSlotStartTime": "2016-01-31T00:00:00Z",
      "MaxTermDurationInDays": 366,
      "SlotDurationInHours": 23,
      "NetworkPlatform": "EC2-VPC",
      "InstanceType": "c4.large",
      "HourlyPrice": "0.095"
    },
    ...
  ]
}
```

Untuk mempersempit hasil, Anda dapat menambahkan filter yang menentukan sistem operasi, jaringan, dan jenis instance.

### Perintah:

```
--filter name=Platform, nilai = Linux/UNIX name=Network-platform, nilai = EC2-VPC
name=Instance-type, nilai = C4.large
```

- Untuk detail API, lihat [DescribeScheduledInstanceAvailability](#) di Referensi AWS CLI Perintah.

## describe-scheduled-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-scheduled-instances`.

### AWS CLI

Untuk menggambarkan Instans Terjadwal

Contoh ini menjelaskan Instance Terjadwal yang ditentukan.

Perintah:

```
aws ec2 describe-scheduled-instances --scheduled-instance-ids
sci-1234-1234-1234-1234-123456789012
```

Output:

```
{
  "ScheduledInstanceSet": [
    {
      "AvailabilityZone": "us-west-2b",
      "ScheduledInstanceId": "sci-1234-1234-1234-1234-123456789012",
      "HourlyPrice": "0.095",
      "CreateDate": "2016-01-25T21:43:38.612Z",
      "Recurrence": {
        "OccurrenceDaySet": [
          1
        ],
        "Interval": 1,
        "Frequency": "Weekly",
        "OccurrenceRelativeToEnd": false,
        "OccurrenceUnit": ""
      },
      "Platform": "Linux/UNIX",
      "TermEndDate": "2017-01-31T09:00:00Z",
      "InstanceCount": 1,
      "SlotDurationInHours": 32,
      "TermStartDate": "2016-01-31T09:00:00Z",
      "NetworkPlatform": "EC2-VPC",
      "TotalScheduledInstanceHours": 1696,
      "NextSlotStartTime": "2016-01-31T09:00:00Z",
      "InstanceType": "c4.large"
    }
  ]
}
```



```
]
}
```

Contoh ini menjelaskan semua Instans Terjadwal Anda.

Perintah:

```
aws ec2 describe-scheduled-instances
```

- Untuk detail API, lihat [DescribeScheduledInstances](#) di Referensi AWS CLI Perintah.

## **describe-security-group-references**

Contoh kode berikut menunjukkan cara menggunakan `describe-security-group-references`.

AWS CLI

Untuk menggambarkan referensi grup keamanan

Contoh ini menjelaskan referensi grup keamanan untuk `sg-bbbb2222`. Respons menunjukkan bahwa kelompok keamanan `sg-bbbb2222` sedang direferensikan oleh grup keamanan di `VPCvpc-aaaaaaaa`.

Perintah:

```
aws ec2 describe-security-group-references --group-id sg-bbbbb22222
```

Output:

```
{
  "SecurityGroupsReferenceSet": [
    {
      "ReferencingVpcId": "vpc-aaaaaaaa ",
      "GroupId": "sg-bbbbb22222",
      "VpcPeeringConnectionId": "pcx-b04deed9"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeSecurityGroupReferences](#) di Referensi AWS CLI Perintah.

## describe-security-group-rules

Contoh kode berikut menunjukkan cara menggunakan `describe-security-group-rules`.

### AWS CLI

Contoh 1: Untuk menjelaskan aturan grup keamanan untuk grup keamanan

`describe-security-group-rules` Contoh berikut menjelaskan aturan grup keamanan dari grup keamanan tertentu. Gunakan `filters` opsi untuk cakupan hasil ke grup keamanan tertentu.

```
aws ec2 describe-security-group-rules \
  --filters Name="group-id",Values="sg-1234567890abcdef0"
```

Output:

```
{
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-abcdef01234567890",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "111122223333",
      "IsEgress": false,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "ReferencedGroupInfo": {
        "GroupId": "sg-1234567890abcdef0",
        "UserId": "111122223333"
      },
      "Tags": []
    },
    {
      "SecurityGroupRuleId": "sgr-bcdef01234567890a",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "111122223333",
      "IsEgress": true,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv6": ":::/0",
      "Tags": []
    }
  ],
}
```

```
{
  "SecurityGroupId": "sg-1234567890abcdef0",
  "GroupOwnerId": "111122223333",
  "IsEgress": true,
  "IpProtocol": "-1",
  "FromPort": -1,
  "ToPort": -1,
  "CidrIpv4": "0.0.0.0/0",
  "Tags": []
}
```

Contoh 2: Untuk mendeskripsikan aturan grup keamanan

`describe-security-group-rules` Contoh berikut menjelaskan aturan grup keamanan yang ditentukan.

```
aws ec2 describe-security-group-rules \
  --security-group-rule-ids sgr-cdef01234567890ab
```

Output:

```
{
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "111122223333",
      "IsEgress": true,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv4": "0.0.0.0/0",
      "Tags": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Aturan grup keamanan](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeSecurityGroupRules](#) di Referensi AWS CLI Perintah.

## describe-security-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-security-groups`.

### AWS CLI

Contoh 1: Untuk menjelaskan grup keamanan

Contoh `describe-security-groups` berikut menjelaskan grup keamanan tertentu.

```
aws ec2 describe-security-groups \
  --group-ids sg-903004f8
```

Output:

```
{
  "SecurityGroups": [
    {
      "IpPermissionsEgress": [
        {
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0"
            }
          ],
          "UserIdGroupPairs": [],
          "PrefixListIds": []
        }
      ],
      "Description": "My security group",
      "Tags": [
        {
          "Value": "SG1",
          "Key": "Name"
        }
      ],
      "IpPermissions": [
        {
          "IpProtocol": "-1",
          "IpRanges": [],
          "UserIdGroupPairs": [
            {
              "UserId": "123456789012",
```

```

        "GroupId": "sg-903004f8"
      }
    ],
    "PrefixListIds": []
  },
  {
    "PrefixListIds": [],
    "FromPort": 22,
    "IpRanges": [
      {
        "Description": "Access from NY office",
        "CidrIp": "203.0.113.0/24"
      }
    ],
    "ToPort": 22,
    "IpProtocol": "tcp",
    "UserIdGroupPairs": []
  }
],
"GroupName": "MySecurityGroup",
"VpcId": "vpc-1a2b3c4d",
"OwnerId": "123456789012",
"GroupId": "sg-903004f8",
}
]
}

```

Contoh 2: Untuk menjelaskan grup keamanan yang memiliki aturan khusus

`describe-security-groups` Contoh berikut menggunakan filter untuk lingkup hasil untuk kelompok keamanan yang memiliki aturan yang memungkinkan lalu lintas SSH (port 22) dan aturan yang memungkinkan lalu lintas dari semua alamat (`0.0.0.0/0`). Contoh tersebut menggunakan parameter `--query` untuk hanya menampilkan nama grup keamanan. Grup keamanan harus cocok dengan semua filter yang akan dikembalikan dalam hasil; namun, satu aturan tidak harus cocok dengan semua filter. Misalnya, output mengembalikan grup keamanan dengan aturan yang memungkinkan lalu lintas SSH dari alamat IP tertentu dan aturan lain yang memungkinkan lalu lintas HTTP dari semua alamat.

```

aws ec2 describe-security-groups \
  --filters Name=ip-permission.from-port,Values=22 Name=ip-permission.to-
port,Values=22 Name=ip-permission.cidr,Values='0.0.0.0/0' \
  --query "SecurityGroups[*].[GroupName]" \

```

```
--output text
```

Output:

```
default
my-security-group
web-servers
launch-wizard-1
```

Contoh 3: Untuk menjelaskan grup keamanan berdasarkan tanda

Contoh `describe-security-groups` berikut menggunakan filter guna mencakup hasil untuk grup keamanan yang menyertakan `test` dalam nama grup keamanan dan yang memiliki tanda `Test=To-delete`. Contoh tersebut menggunakan parameter `--query` untuk hanya menampilkan nama dan ID grup keamanan.

```
aws ec2 describe-security-groups \
  --filters Name=group-name,Values=*test* Name=tag:Test,Values=To-delete \
  --query "SecurityGroups[*].{Name:GroupName,ID:GroupId}"
```

Output:

```
[
  {
    "Name": "testfornewinstance",
    "ID": "sg-33bb22aa"
  },
  {
    "Name": "newgrouptest",
    "ID": "sg-1a2b3c4d"
  }
]
```

Untuk contoh tambahan menggunakan filter tanda, lihat [Bekerja dengan tanda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeSecurityGroups](#) di Referensi AWS CLI Perintah.

## describe-snapshot-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshot-attribute`.

## AWS CLI

Untuk mendeskripsikan atribut snapshot untuk snapshot

`describe-snapshot-attribute` Contoh berikut mencantumkan akun yang digunakan untuk berbagi snapshot.

```
aws ec2 describe-snapshot-attribute \  
  --snapshot-id snap-01234567890abcdef \  
  --attribute createVolumePermission
```

Output:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "CreateVolumePermissions": [  
    {  
      "UserId": "123456789012"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membagikan snapshot Amazon EBS di Panduan Pengguna Amazon Elastic Compute Cloud](#).

- Untuk detail API, lihat [DescribeSnapshotAttribute](#) di Referensi AWS CLI Perintah.

## `describe-snapshot-tier-status`

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshot-tier-status`.

## AWS CLI

Untuk melihat informasi arsip tentang snapshot yang diarsipkan

`describe-snapshot-tier-status` Contoh berikut memberikan informasi arsip tentang snapshot yang diarsipkan.

```
aws ec2 describe-snapshot-tier-status \  
  --filters "Name=snapshot-id, Values=snap-01234567890abcdef"
```

Output:

```
{
  "SnapshotTierStatuses": [
    {
      "Status": "completed",
      "ArchivalCompleteTime": "2021-09-15T17:33:16.147Z",
      "LastTieringProgress": 100,
      "Tags": [],
      "VolumeId": "vol-01234567890abcdef",
      "LastTieringOperationState": "archival-completed",
      "StorageTier": "archive",
      "OwnerId": "123456789012",
      "SnapshotId": "snap-01234567890abcdef",
      "LastTieringStartTime": "2021-09-15T16:44:37.574Z"
    }
  ]
}
```

Untuk informasi selengkapnya, buka [Lihat snapshot yang diarsipkan](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [DescribeSnapshotTierStatus](#) di Referensi AWS CLI Perintah.

## describe-snapshots

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshots`.

### AWS CLI

Contoh 1: Untuk menjelaskan snapshot

Contoh `describe-snapshots` berikut menjelaskan snapshot tertentu.

```
aws ec2 describe-snapshots \
  --snapshot-ids snap-1234567890abcdef0
```

Output:

```
{
  "Snapshots": [
    {
      "Description": "This is my snapshot",
      "Encrypted": false,
```



```

    "VolumeId": "vol-049df61146c4d7901",
    "State": "completed",
    "VolumeSize": 8,
    "StartTime": "2019-02-28T21:28:32.000Z",
    "Progress": "100%",
    "OwnerId": "012345678910",
    "SnapshotId": "snap-01234567890abcdef",
    "Tags": [
      {
        "Key": "Stack",
        "Value": "test"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Snapshot Amazon EBS](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menjelaskan snapshot berdasarkan filter

`describe-snapshots` Contoh berikut menggunakan filter untuk mencakupkan hasil ke snapshot yang dimiliki oleh AWS akun Anda yang berada dalam pending status. Contoh tersebut menggunakan parameter `--query` untuk hanya menampilkan ID snapshot dan waktu mulai snapshot.

```

aws ec2 describe-snapshots \
  --owner-ids self \
  --filters Name=status,Values=pending \
  --query "Snapshots[*].{ID:SnapshotId,Time:StartTime}"

```

Output:

```

[
  {
    "ID": "snap-1234567890abcdef0",
    "Time": "2019-08-04T12:48:18.000Z"
  },
  {
    "ID": "snap-066877671789bd71b",
    "Time": "2019-08-04T02:45:16.000Z"
  },
  ...
]

```

```
] ]
```

Contoh `describe-snapshots` berikut menggunakan filter guna mencakup hasil untuk snapshot yang dibuat dari volume tertentu. Contoh tersebut menggunakan parameter `--query` untuk hanya menampilkan ID snapshot.

```
aws ec2 describe-snapshots \  
  --filters Name=volume-id,Values=049df61146c4d7901 \  
  --query "Snapshots[*].[SnapshotId]" \  
  --output text
```

Output:

```
snap-1234567890abcdef0  
snap-08637175a712c3fb9  
...
```

Untuk contoh tambahan menggunakan filter, lihat [Mencantumkan dan memfilter sumber daya Anda](#) di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk menjelaskan snapshot berdasarkan tanda

Contoh `describe-snapshots` berikut menggunakan filter tanda guna mencakup hasil untuk snapshot yang memiliki tanda `Stack=Prod`.

```
aws ec2 describe-snapshots \  
  --filters Name=tag:Stack,Values=prod
```

Untuk contoh output `describe-snapshots`, lihat Contoh 1.

Untuk contoh tambahan menggunakan filter tanda, lihat [Bekerja dengan tanda](#) di Panduan Pengguna Amazon EC2.

Contoh 4: Untuk menjelaskan snapshot berdasarkan usia

Contoh `describe-snapshots` berikut menggunakan ekspresi JMESPath untuk menggambarkan semua snapshot yang dibuat oleh AWS akun Anda sebelum tanggal yang ditentukan. Contoh tersebut hanya menampilkan ID snapshot.

```
aws ec2 describe-snapshots \  
  --owner-ids 012345678910 \  
  --query "Snapshots[*].[SnapshotId]"
```

```
--query "Snapshots[?(StartTime<='2020-03-31')].[SnapshotId]"
```

Untuk contoh tambahan menggunakan filter, lihat [Mencantumkan dan memfilter sumber daya Anda](#) di Panduan Pengguna Amazon EC2.

Contoh 5: Untuk melihat snapshot yang diarsipkan saja

Contoh `describe-snapshots` berikut hanya mencantumkan snapshot yang disimpan di tingkat arsip.

```
aws ec2 describe-snapshots \  
--filters "Name=storage-tier,Values=archive"
```

Output:

```
{  
  "Snapshots": [  
    {  
      "Description": "Snap A",  
      "Encrypted": false,  
      "VolumeId": "vol-01234567890aaaaaa",  
      "State": "completed",  
      "VolumeSize": 8,  
      "StartTime": "2021-09-07T21:00:00.000Z",  
      "Progress": "100%",  
      "OwnerId": "123456789012",  
      "SnapshotId": "snap-01234567890aaaaaa",  
      "StorageTier": "archive",  
      "Tags": []  
    },  
  ]  
}
```

Untuk informasi selengkapnya, buka [Lihat snapshot yang diarsipkan](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [DescribeSnapshots](#) di Referensi AWS CLI Perintah.

## **describe-spot-datafeed-subscription**

Contoh kode berikut menunjukkan cara menggunakan `describe-spot-datafeed-subscription`.

## AWS CLI

Untuk mendeskripsikan langganan datafeed Instance Spot untuk sebuah akun

Perintah contoh ini menjelaskan umpan data untuk akun.

Perintah:

```
aws ec2 describe-spot-datafeed-subscription
```

Output:

```
{
  "SpotDatafeedSubscription": {
    "OwnerId": "123456789012",
    "Prefix": "spotdata",
    "Bucket": "my-s3-bucket",
    "State": "Active"
  }
}
```

- Untuk detail API, lihat [DescribeSpotDatafeedSubscription](#) di Referensi AWS CLI Perintah.

## describe-spot-fleet-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-spot-fleet-instances`.

### AWS CLI

Untuk menggambarkan Instans Spot yang terkait dengan armada Spot

Perintah contoh ini mencantumkan instance Spot yang terkait dengan armada Spot yang ditentukan.

Perintah:

```
aws ec2 describe-spot-fleet-instances --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
```

```
"ActiveInstances": [  
  {  
    "InstanceId": "i-1234567890abcdef0",  
    "InstanceType": "m3.medium",  
    "SpotInstanceRequestId": "sir-08b93456"  
  },  
  ...  
],  
"SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"  
}
```

- Untuk detail API, lihat [DescribeSpotFleetInstances](#) di Referensi AWS CLI Perintah.

## describe-spot-fleet-request-history

Contoh kode berikut menunjukkan cara menggunakan `describe-spot-fleet-request-history`.

### AWS CLI

Untuk menggambarkan sejarah armada Spot

Perintah contoh ini mengembalikan riwayat untuk armada Spot yang ditentukan mulai pada waktu yang ditentukan.

Perintah:

```
aws ec2 describe-spot-fleet-request-history --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE --start-time 2015-05-26T00:00:00Z
```

Contoh keluaran berikut menunjukkan keberhasilan peluncuran dua Instans Spot untuk armada Spot.

Output:

```
{  
  "HistoryRecords": [  
    {  
      "Timestamp": "2015-05-26T23:17:20.697Z",  
      "EventInformation": {  
        "EventSubType": "submitted"  
      }  
    }  
  ]  
}
```

```

    },
    "EventType": "fleetRequestChange"
  },
  {
    "Timestamp": "2015-05-26T23:17:20.873Z",
    "EventInformation": {
      "EventSubType": "active"
    },
    "EventType": "fleetRequestChange"
  },
  {
    "Timestamp": "2015-05-26T23:21:21.712Z",
    "EventInformation": {
      "InstanceId": "i-1234567890abcdef0",
      "EventSubType": "launched"
    },
    "EventType": "instanceChange"
  },
  {
    "Timestamp": "2015-05-26T23:21:21.816Z",
    "EventInformation": {
      "InstanceId": "i-1234567890abcdef1",
      "EventSubType": "launched"
    },
    "EventType": "instanceChange"
  }
],
"SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
"NextToken": "CpHNsscimcV5oH7bSbub03CI2Qms5+ypNpNm
+53MNLr0YcXAkp0xFlfKf91yVxSExmbtma3awYxMFzNA663ZskT0AhtJ6TCb2Z8bQC2EnZgyELbymtWPfpZ1ZbauVg
+P+TfG1WxWwB/Vr5dk5d4LfdgA/DRAHUrYgxzrEXAMPLE=",
"StartTime": "2015-05-26T00:00:00Z"
}

```

- Untuk detail API, lihat [DescribeSpotFleetRequestHistory](#) di Referensi AWS CLI Perintah.

## describe-spot-fleet-requests

Contoh kode berikut menunjukkan cara menggunakan `describe-spot-fleet-requests`.

### AWS CLI

Untuk menjelaskan permintaan armada Spot Anda

Contoh ini menjelaskan semua permintaan armada Spot Anda.

Perintah:

```
aws ec2 describe-spot-fleet-requests
```

Output:

```
{
  "SpotFleetRequestConfigs": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
      "SpotFleetRequestConfig": {
        "TargetCapacity": 20,
        "LaunchSpecifications": [
          {
            "EbsOptimized": false,
            "NetworkInterfaces": [
              {
                "SubnetId": "subnet-a61dafcf",
                "DeviceIndex": 0,
                "DeleteOnTermination": false,
                "AssociatePublicIpAddress": true,
                "SecondaryPrivateIpAddressCount": 0
              }
            ],
            "InstanceType": "cc2.8xlarge",
            "ImageId": "ami-1a2b3c4d"
          },
          {
            "EbsOptimized": false,
            "NetworkInterfaces": [
              {
                "SubnetId": "subnet-a61dafcf",
                "DeviceIndex": 0,
                "DeleteOnTermination": false,
                "AssociatePublicIpAddress": true,
                "SecondaryPrivateIpAddressCount": 0
              }
            ],
            "InstanceType": "r3.8xlarge",
            "ImageId": "ami-1a2b3c4d"
          }
        ]
      }
    }
  ]
}
```

```

    ],
    "SpotPrice": "0.05",
    "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
  },
  "SpotFleetRequestState": "active"
},
{
  "SpotFleetRequestId": "sfr-306341ed-9739-402e-881b-ce47bEXAMPLE",
  "SpotFleetRequestConfig": {
    "TargetCapacity": 20,
    "LaunchSpecifications": [
      {
        "EbsOptimized": false,
        "NetworkInterfaces": [
          {
            "SubnetId": "subnet-6e7f829e",
            "DeviceIndex": 0,
            "DeleteOnTermination": false,
            "AssociatePublicIpAddress": true,
            "SecondaryPrivateIpAddressCount": 0
          }
        ],
        "InstanceType": "m3.medium",
        "ImageId": "ami-1a2b3c4d"
      }
    ],
    "SpotPrice": "0.05",
    "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
  },
  "SpotFleetRequestState": "active"
}
]
}

```

Untuk menjelaskan permintaan armada Spot

Contoh ini menjelaskan permintaan armada Spot yang ditentukan.

Perintah:

```
aws ec2 describe-spot-fleet-requests --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```



## Output:

```
{
  "SpotFleetRequestConfigs": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
      "SpotFleetRequestConfig": {
        "TargetCapacity": 20,
        "LaunchSpecifications": [
          {
            "EbsOptimized": false,
            "NetworkInterfaces": [
              {
                "SubnetId": "subnet-a61dafcf",
                "DeviceIndex": 0,
                "DeleteOnTermination": false,
                "AssociatePublicIpAddress": true,
                "SecondaryPrivateIpAddressCount": 0
              }
            ],
            "InstanceType": "cc2.8xlarge",
            "ImageId": "ami-1a2b3c4d"
          },
          {
            "EbsOptimized": false,
            "NetworkInterfaces": [
              {
                "SubnetId": "subnet-a61dafcf",
                "DeviceIndex": 0,
                "DeleteOnTermination": false,
                "AssociatePublicIpAddress": true,
                "SecondaryPrivateIpAddressCount": 0
              }
            ],
            "InstanceType": "r3.8xlarge",
            "ImageId": "ami-1a2b3c4d"
          }
        ],
        "SpotPrice": "0.05",
        "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
      },
      "SpotFleetRequestState": "active"
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [DescribeSpotFleetRequests](#) di Referensi AWS CLI Perintah.

## describe-spot-instance-requests

Contoh kode berikut menunjukkan cara menggunakan `describe-spot-instance-requests`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan permintaan Instans Spot

`describe-spot-instance-requests` Contoh berikut menjelaskan permintaan Instans Spot yang ditentukan.

```
aws ec2 describe-spot-instance-requests \
  --spot-instance-request-ids sir-08b93456
```

Output:

```
{
  "SpotInstanceRequests": [
    {
      "CreateTime": "2018-04-30T18:14:55.000Z",
      "InstanceId": "i-1234567890abcdef1",
      "LaunchSpecification": {
        "InstanceType": "t2.micro",
        "ImageId": "ami-003634241a8fcdec0",
        "KeyName": "my-key-pair",
        "SecurityGroups": [
          {
            "GroupName": "default",
            "GroupId": "sg-e38f24a7"
          }
        ],
        "BlockDeviceMappings": [
          {
            "DeviceName": "/dev/sda1",
            "Ebs": {
              "DeleteOnTermination": true,
              "SnapshotId": "snap-0e54a519c999adbbd",
              "VolumeSize": 8,

```

```

        "VolumeType": "standard",
        "Encrypted": false
    }
  ],
  "NetworkInterfaces": [
    {
      "DeleteOnTermination": true,
      "DeviceIndex": 0,
      "SubnetId": "subnet-049df61146c4d7901"
    }
  ],
  "Placement": {
    "AvailabilityZone": "us-east-2b",
    "Tenancy": "default"
  },
  "Monitoring": {
    "Enabled": false
  }
},
"LaunchedAvailabilityZone": "us-east-2b",
"ProductDescription": "Linux/UNIX",
"SpotInstanceRequestId": "sir-08b93456",
"SpotPrice": "0.010000"
"State": "active",
"Status": {
  "Code": "fulfilled",
  "Message": "Your Spot request is fulfilled.",
  "UpdateTime": "2018-04-30T18:16:21.000Z"
},
"Tags": [],
"Type": "one-time",
"InstanceInterruptionBehavior": "terminate"
}
]
}

```

Contoh 2: Untuk menjelaskan permintaan Instans Spot berdasarkan filter

`describe-spot-instance-requests` Contoh berikut menggunakan filter untuk membuat cakupan hasil ke permintaan Instance Spot dengan jenis instans yang ditentukan di Availability Zone yang ditentukan. Contoh menggunakan `--query` parameter untuk menampilkan hanya ID instance.

```
aws ec2 describe-spot-instance-requests \  
  --filters Name=launch.instance-type,Values=m3.medium Name=launched-availability-  
zone,Values=us-east-2a \  
  --query "SpotInstanceRequests[*].[InstanceId]" \  
  --output text
```

Output:

```
i-057750d42936e468a  
i-001efd250faaa6ffa  
i-027552a73f021f3bd  
...
```

Untuk contoh tambahan menggunakan filter, lihat [Mencantumkan dan memfilter sumber daya Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

Contoh 3: Untuk mendeskripsikan permintaan Instans Spot berdasarkan tag

`describe-spot-instance-requests` Contoh berikut menggunakan filter tag untuk cakupan hasil ke permintaan Spot Instance yang memiliki `tag:cost-center=cc123`.

```
aws ec2 describe-spot-instance-requests \  
  --filters Name=tag:cost-center,Values=cc123
```

Untuk contoh output `describe-spot-instance-requests`, lihat Contoh 1.

Untuk contoh tambahan menggunakan filter tanda, lihat [Bekerja dengan tanda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeSpotInstanceRequests](#) di Referensi AWS CLI Perintah.

## **describe-spot-price-history**

Contoh kode berikut menunjukkan cara menggunakan `describe-spot-price-history`.

### AWS CLI

Untuk menggambarkan sejarah harga Spot

Perintah contoh ini mengembalikan riwayat Harga Spot untuk instance `m1.xlarge` untuk hari tertentu di bulan Januari.

**Perintah:**

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --start-time
2014-01-06T07:08:09 --end-time 2014-01-06T08:09:10
```

**Output:**

```
{
  "SpotPriceHistory": [
    {
      "Timestamp": "2014-01-06T07:10:55.000Z",
      "ProductDescription": "SUSE Linux",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.087000",
      "AvailabilityZone": "us-west-1b"
    },
    {
      "Timestamp": "2014-01-06T07:10:55.000Z",
      "ProductDescription": "SUSE Linux",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.087000",
      "AvailabilityZone": "us-west-1c"
    },
    {
      "Timestamp": "2014-01-06T05:42:36.000Z",
      "ProductDescription": "SUSE Linux (Amazon VPC)",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.087000",
      "AvailabilityZone": "us-west-1a"
    },
    ...
  ]
}
```

Untuk menggambarkan sejarah harga Spot untuk Linux/Unix Amazon VPC

Perintah contoh ini mengembalikan histori Harga Spot untuk instans VPC Amazon Linux/Unix m1.xlarge untuk hari tertentu di bulan Januari.

**Perintah:**

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --product-
description "Linux/UNIX (Amazon VPC)" --start-time 2014-01-06T07:08:09 --end-time
2014-01-06T08:09:10
```

Output:

```
{
  "SpotPriceHistory": [
    {
      "Timestamp": "2014-01-06T04:32:53.000Z",
      "ProductDescription": "Linux/UNIX (Amazon VPC)",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.080000",
      "AvailabilityZone": "us-west-1a"
    },
    {
      "Timestamp": "2014-01-05T11:28:26.000Z",
      "ProductDescription": "Linux/UNIX (Amazon VPC)",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.080000",
      "AvailabilityZone": "us-west-1c"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeSpotPriceHistory](#) di Referensi AWS CLI Perintah.

## describe-stale-security-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-stale-security-groups`.

### AWS CLI

Untuk menggambarkan kelompok keamanan basi

Contoh ini menjelaskan aturan grup keamanan basi untuk `vpc-11223344`. Tanggapan menunjukkan bahwa `sg-5fa68d3a` di akun Anda memiliki aturan SSH ingress basi yang mereferensikan di VPC rekan, dan bahwa di `sg-279ab042` akun Anda memiliki aturan SSH keluar basi yang merujuk `sg-fe6fba9a` dalam VPC rekan. `sg-ef6fba8b`

Perintah:

```
aws ec2 describe-stale-security-groups --vpc-id vpc-11223344
```

**Output:**

```
{
  "StaleSecurityGroupSet": [
    {
      "VpcId": "vpc-11223344",
      "StaleIpPermissionsEgress": [
        {
          "ToPort": 22,
          "FromPort": 22,
          "UserIdGroupPairs": [
            {
              "VpcId": "vpc-7a20e51f",
              "GroupId": "sg-ef6fba8b",
              "VpcPeeringConnectionId": "pcx-b04deed9",
              "PeeringStatus": "active"
            }
          ],
          "IpProtocol": "tcp"
        }
      ],
      "GroupName": "MySG1",
      "StaleIpPermissions": [],
      "GroupId": "sg-fe6fba9a",
      "Description": "MySG1"
    },
    {
      "VpcId": "vpc-11223344",
      "StaleIpPermissionsEgress": [],
      "GroupName": "MySG2",
      "StaleIpPermissions": [
        {
          "ToPort": 22,
          "FromPort": 22,
          "UserIdGroupPairs": [
            {
              "VpcId": "vpc-7a20e51f",
              "GroupId": "sg-279ab042",
              "Description": "Access from pcx-b04deed9",
              "VpcPeeringConnectionId": "pcx-b04deed9",
              "PeeringStatus": "active"
            }
          ]
        }
      ]
    }
  ]
}
```

```

        }
      ],
      "IpProtocol": "tcp"
    }
  ],
  "GroupId": "sg-5fa68d3a",
  "Description": "MySG2"
}
]
}

```

- Untuk detail API, lihat [DescribeStaleSecurityGroups](#) di Referensi AWS CLI Perintah.

## describe-store-image-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-store-image-tasks`.

### AWS CLI

Untuk menggambarkan kemajuan tugas penyimpanan AMI

`describe-store-image-tasks` Contoh berikut menjelaskan kemajuan tugas penyimpanan AMI.

```
aws ec2 describe-store-image-tasks
```

Output:

```

{
  "AmiId": "ami-1234567890abcdef0",
  "Bucket": "my-ami-bucket",
  "ProgressPercentage": 17,
  "S3ObjectKey": "ami-1234567890abcdef0.bin",
  "StoreTaskState": "InProgress",
  "StoreTaskFailureReason": null,
  "TaskStartTime": "2022-01-01T01:01:01.001Z"
}

```

Untuk informasi selengkapnya tentang menyimpan dan memulihkan AMI menggunakan S3, lihat [Menyimpan dan memulihkan AMI menggunakan S3 < https://docs.aws.amazon.com/AWS EC2/latest//.html >](https://docs.aws.amazon.com/AWS EC2/latest//.html) di Panduan Pengguna Amazon EC2 User Guide. `ami-store-restore`



- Untuk detail API, lihat [DescribeStoreImageTasks](#) di Referensi AWS CLI Perintah.

## describe-subnets

Contoh kode berikut menunjukkan cara menggunakan `describe-subnets`.

### AWS CLI

Contoh 1: Untuk menjelaskan semua subnet Anda

Contoh `describe-subnets` berikut menampilkan detail subnet Anda.

```
aws ec2 describe-subnets
```

Output:

```
{
  "Subnets": [
    {
      "AvailabilityZone": "us-east-1d",
      "AvailabilityZoneId": "use1-az2",
      "AvailableIpAddressCount": 4089,
      "CidrBlock": "172.31.80.0/20",
      "DefaultForAz": true,
      "MapPublicIpOnLaunch": false,
      "MapCustomerOwnedIpOnLaunch": true,
      "State": "available",
      "SubnetId": "subnet-0bb1c79de3EXAMPLE",
      "VpcId": "vpc-0ee975135dEXAMPLE",
      "OwnerId": "111122223333",
      "AssignIpv6AddressOnCreation": false,
      "Ipv6CidrBlockAssociationSet": [],
      "CustomerOwnedIpv4Pool": "pool-2EXAMPLE",
      "SubnetArn": "arn:aws:ec2:us-east-2:111122223333:subnet/
subnet-0bb1c79de3EXAMPLE",
      "EnableDns64": false,
      "Ipv6Native": false,
      "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
        "EnableResourceNameDnsARecord": false,
        "EnableResourceNameDnsAAAARecord": false
      }
    }
  ]
}
```

```
    },
    {
      "AvailabilityZone": "us-east-1d",
      "AvailabilityZoneId": "use1-az2",
      "AvailableIpAddressCount": 4089,
      "CidrBlock": "172.31.80.0/20",
      "DefaultForAz": true,
      "MapPublicIpOnLaunch": true,
      "MapCustomerOwnedIpOnLaunch": false,
      "State": "available",
      "SubnetId": "subnet-8EXAMPLE",
      "VpcId": "vpc-3EXAMPLE",
      "OwnerId": "1111222233333",
      "AssignIpv6AddressOnCreation": false,
      "Ipv6CidrBlockAssociationSet": [],
      "Tags": [
        {
          "Key": "Name",
          "Value": "MySubnet"
        }
      ],
      "SubnetArn": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-8EXAMPLE",
      "EnableDns64": false,
      "Ipv6Native": false,
      "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
        "EnableResourceNameDnsARecord": false,
        "EnableResourceNameDnsAAAARecord": false
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan VPC dan Subnet](#) di Panduan Pengguna VPC AWS .

Contoh 2: Untuk menjelaskan subnet dari VPC tertentu

Contoh `describe-subnets` berikut menggunakan filter guna mengambil detail untuk subnet dari VPC tertentu.

```
aws ec2 describe-subnets \
```

```
--filters "Name=vpc-id,Values=vpc-3EXAMPLE"
```

**Output:**

```
{
  "Subnets": [
    {
      "AvailabilityZone": "us-east-1d",
      "AvailabilityZoneId": "use1-az2",
      "AvailableIpAddressCount": 4089,
      "CidrBlock": "172.31.80.0/20",
      "DefaultForAz": true,
      "MapPublicIpOnLaunch": true,
      "MapCustomerOwnedIpOnLaunch": false,
      "State": "available",
      "SubnetId": "subnet-8EXAMPLE",
      "VpcId": "vpc-3EXAMPLE",
      "OwnerId": "1111222233333",
      "AssignIpv6AddressOnCreation": false,
      "Ipv6CidrBlockAssociationSet": [],
      "Tags": [
        {
          "Key": "Name",
          "Value": "MySubnet"
        }
      ],
      "SubnetArn": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-8EXAMPLE",
      "EnableDns64": false,
      "Ipv6Native": false,
      "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
        "EnableResourceNameDnsARecord": false,
        "EnableResourceNameDnsAAAARecord": false
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan VPC dan Subnet](#) di Panduan Pengguna VPC AWS .

Contoh 3: Untuk menjelaskan subnet dengan tanda tertentu

Contoh `describe-subnets` berikut menggunakan filter untuk mengambil detail subnet tersebut dengan tanda `CostCenter=123` dan parameter `--query` untuk menampilkan ID subnet dari subnet dengan tanda ini.

```
aws ec2 describe-subnets \
  --filters "Name=tag:CostCenter,Values=123" \
  --query "Subnets[*].SubnetId" \
  --output text
```

Output:

```
subnet-0987a87c8b37348ef
subnet-02a95061c45f372ee
subnet-03f720e7de2788d73
```

Untuk informasi selengkapnya, lihat [Bekerja dengan VPC dan Subnet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeSubnets](#) di Referensi AWS CLI Perintah.

## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan semua tag untuk satu sumber daya

`describe-tags` Contoh berikut menjelaskan tag untuk contoh yang ditentukan.

```
aws ec2 describe-tags \
  --filters "Name=resource-id,Values=i-1234567890abcdef8"
```

Output:

```
{
  "Tags": [
    {
      "ResourceType": "instance",
      "ResourceId": "i-1234567890abcdef8",
      "Value": "Test",
```

```
        "Key": "Stack"
      },
      {
        "ResourceType": "instance",
        "ResourceId": "i-1234567890abcdef8",
        "Value": "Beta Server",
        "Key": "Name"
      }
    ]
  }
}
```

Contoh 2: Untuk mendeskripsikan semua tag untuk jenis sumber daya

`describe-tags` Contoh berikut menjelaskan tag untuk volume Anda.

```
aws ec2 describe-tags \
  --filters "Name=resource-type,Values=volume"
```

Output:

```
{
  "Tags": [
    {
      "ResourceType": "volume",
      "ResourceId": "vol-1234567890abcdef0",
      "Value": "Project1",
      "Key": "Purpose"
    },
    {
      "ResourceType": "volume",
      "ResourceId": "vol-049df61146c4d7901",
      "Value": "Logs",
      "Key": "Purpose"
    }
  ]
}
```

Contoh 3: Untuk mendeskripsikan semua tag Anda

`describe-tags` Contoh berikut menjelaskan tag untuk semua sumber daya Anda.

```
aws ec2 describe-tags
```

Contoh 4: Untuk mendeskripsikan tag untuk sumber daya Anda berdasarkan kunci tag

`describe-tags` Contoh berikut menjelaskan tag untuk sumber daya Anda yang memiliki tag dengan kunci `Stack`.

```
aws ec2 describe-tags \  
  --filters Name=key,Values=Stack
```

Output:

```
{  
  "Tags": [  
    {  
      "ResourceType": "volume",  
      "ResourceId": "vol-027552a73f021f3b",  
      "Value": "Production",  
      "Key": "Stack"  
    },  
    {  
      "ResourceType": "instance",  
      "ResourceId": "i-1234567890abcdef8",  
      "Value": "Test",  
      "Key": "Stack"  
    }  
  ]  
}
```

Contoh 5: Untuk mendeskripsikan tag untuk sumber daya Anda berdasarkan kunci tag dan nilai tag

`describe-tags` Contoh berikut menjelaskan tag untuk sumber daya Anda yang memiliki tag `Stack=Test`.

```
aws ec2 describe-tags \  
  --filters Name=key,Values=Stack Name=value,Values=Test
```

Output:

```
{  
  "Tags": [  
    {
```

```

    "ResourceType": "image",
    "ResourceId": "ami-3ac336533f021f3bd",
    "Value": "Test",
    "Key": "Stack"
  },
  {
    "ResourceType": "instance",
    "ResourceId": "i-1234567890abcdef8",
    "Value": "Test",
    "Key": "Stack"
  }
]
}

```

`describe-tags` Contoh berikut menggunakan sintaks alternatif untuk menggambarkan sumber daya dengan tag `Stack=Test`.

```

aws ec2 describe-tags \
  --filters "Name=tag:Stack,Values=Test"

```

`describe-tags` Contoh berikut menjelaskan tag untuk semua instance Anda yang memiliki tag dengan kunci `Purpose` dan tanpa nilai.

```

aws ec2 describe-tags \
  --filters "Name=resource-type,Values=instance" "Name=key,Values=Purpose"
  "Name=value,Values="

```

Output:

```

{
  "Tags": [
    {
      "ResourceType": "instance",
      "ResourceId": "i-1234567890abcdef5",
      "Value": null,
      "Key": "Purpose"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## describe-traffic-mirror-filters

Contoh kode berikut menunjukkan cara menggunakan `describe-traffic-mirror-filters`.

### AWS CLI

Untuk melihat filter cermin lalu lintas Anda

`describe-traffic-mirror-filters` Contoh berikut menampilkan detail untuk semua filter cermin lalu lintas Anda.

```
aws ec2 describe-traffic-mirror-filters
```

Output:

```
{
  "TrafficMirrorFilters": [
    {
      "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
      "IngressFilterRules": [
        {
          "TrafficMirrorFilterRuleId": "tmfr-0ca76e0e08EXAMPLE",
          "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
          "TrafficDirection": "ingress",
          "RuleNumber": 100,
          "RuleAction": "accept",
          "Protocol": 6,
          "DestinationCidrBlock": "10.0.0.0/24",
          "SourceCidrBlock": "10.0.0.0/24",
          "Description": "TCP Rule"
        }
      ],
      "EgressFilterRules": [],
      "NetworkServices": [],
      "Description": "Example filter",
      "Tags": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat filter cermin lalu lintas Anda](#) di Panduan Pencermian Lalu Lintas.



- Untuk detail API, lihat [DescribeTrafficMirrorFilters](#) di Referensi AWS CLI Perintah.

## describe-traffic-mirror-sessions

Contoh kode berikut menunjukkan cara menggunakan `describe-traffic-mirror-sessions`.

### AWS CLI

Untuk menggambarkan Sesi Cermin Lalu Lintas

`describe-traffic-mirror-sessions` Contoh berikut menampilkan detail sesi Traffic Mirror Anda.

```
aws ec2 describe-traffic-mirror-sessions
```

Output:

```
{
  "TrafficMirrorSessions": [
    {
      "Tags": [],
      "VirtualNetworkId": 42,
      "OwnerId": "111122223333",
      "Description": "TCP Session",
      "NetworkInterfaceId": "eni-0a471a5cf3EXAMPLE",
      "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE",
      "TrafficMirrorFilterId": "tmf-083e18f985EXAMPLE",
      "PacketLength": 20,
      "SessionNumber": 1,
      "TrafficMirrorSessionId": "tms-0567a4c684EXAMPLE"
    },
    {
      "Tags": [
        {
          "Key": "Name",
          "Value": "tag test"
        }
      ],
      "VirtualNetworkId": 13314501,
      "OwnerId": "111122223333",
      "Description": "TCP Session",
      "NetworkInterfaceId": "eni-0a471a5cf3EXAMPLE",
```

```

        "TrafficMirrorTargetId": "tmt-03665551cbEXAMPLE",
        "TrafficMirrorFilterId": "tmf-06c787846cEXAMPLE",
        "SessionNumber": 2,
        "TrafficMirrorSessionId": "tms-0060101cf8EXAMPLE"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Lihat Rincian Sesi Cermin Lalu Lintas](#) di Panduan Pencermian AWS Lalu Lintas.

- Untuk detail API, lihat [DescribeTrafficMirrorSessions](#) di Referensi AWS CLI Perintah.

## describe-traffic-mirror-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-traffic-mirror-targets`.

### AWS CLI

Untuk menggambarkan target cermin lalu lintas

`describe-traffic-mirror-targets` Contoh berikut menampilkan informasi tentang target cermin lalu lintas yang ditentukan.

```

aws ec2 describe-traffic-mirror-targets \
  --traffic-mirror-target-ids tmt-0dabe9b0a6EXAMPLE

```

Output:

```

{
  "TrafficMirrorTargets": [
    {
      "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE",
      "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/net/NLB/7cdec873fEXAMPLE",
      "Type": "network-load-balancer",
      "Description": "Example Network Load Balancer target",
      "OwnerId": "111122223333",
      "Tags": []
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Target cermin lalu lintas](#) di Panduan Pencerminan Lalu Lintas VPC Amazon.

- Untuk detail API, lihat [DescribeTrafficMirrorTargets](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-attachments

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-attachments`.

### AWS CLI

Untuk melihat lampiran gateway transit

`describe-transit-gateway-attachments` Contoh berikut menampilkan detail untuk lampiran gateway transit Anda.

```
aws ec2 describe-transit-gateway-attachments
```

Output:

```
{
  "TransitGatewayAttachments": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-01f8100bc7EXAMPLE",
      "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
      "TransitGatewayOwnerId": "123456789012",
      "ResourceOwnerId": "123456789012",
      "ResourceType": "vpc",
      "ResourceId": "vpc-3EXAMPLE",
      "State": "available",
      "Association": {
        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "State": "associated"
      },
      "CreationTime": "2019-08-26T14:59:25.000Z",
      "Tags": [
        {
          "Key": "Name",
          "Value": "Example"
        }
      ]
    }
  ],
}
```

```
{
  "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
  "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
  "TransitGatewayOwnerId": "123456789012",
  "ResourceOwnerId": "123456789012",
  "ResourceType": "vpc",
  "ResourceId": "vpc-0065acced4EXAMPLE",
  "State": "available",
  "Association": {
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
    "State": "associated"
  },
  "CreationTime": "2019-08-07T17:03:07.000Z",
  "Tags": []
},
{
  "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
  "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
  "TransitGatewayOwnerId": "123456789012",
  "ResourceOwnerId": "123456789012",
  "ResourceType": "direct-connect-gateway",
  "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
  "State": "available",
  "Association": {
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
    "State": "associated"
  },
  "CreationTime": "2019-08-14T20:27:44.000Z",
  "Tags": []
},
{
  "TransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
  "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
  "TransitGatewayOwnerId": "123456789012",
  "ResourceOwnerId": "123456789012",
  "ResourceType": "direct-connect-gateway",
  "ResourceId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
  "State": "available",
  "Association": {
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
    "State": "associated"
  },
  "CreationTime": "2019-08-14T20:33:02.000Z",
  "Tags": []
}
```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan gateway transit di Panduan Gerbang Transit](#).

- Untuk detail API, lihat [DescribeTransitGatewayAttachments](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-connect-peers

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-connect-peers`.

### AWS CLI

Untuk menggambarkan rekan Transit Gateway Connect

`describe-transit-gateway-connect-peers` Contoh berikut menjelaskan Connect peer yang ditentukan.

```

aws ec2 describe-transit-gateway-connect-peers \
  --transit-gateway-connect-peer-ids tgw-connect-peer-0666adbac4EXAMPLE

```

### Output:

```

{
  "TransitGatewayConnectPeers": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
      "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
      "State": "available",
      "CreationTime": "2021-10-13T03:35:17.000Z",
      "ConnectPeerConfiguration": {
        "TransitGatewayAddress": "10.0.0.234",
        "PeerAddress": "172.31.1.11",
        "InsideCidrBlocks": [
          "169.254.6.0/29"
        ],
        "Protocol": "gre",
        "BgpConfigurations": [
          {
            "TransitGatewayAsn": 64512,
            "PeerAsn": 64512,

```

```

    "TransitGatewayAddress": "169.254.6.2",
    "PeerAddress": "169.254.6.1",
    "BgpStatus": "down"
  },
  {
    "TransitGatewayAsn": 64512,
    "PeerAsn": 64512,
    "TransitGatewayAddress": "169.254.6.3",
    "PeerAddress": "169.254.6.1",
    "BgpStatus": "down"
  }
]
},
"Tags": []
}
]
}

```

Untuk informasi selengkapnya, lihat [Lampiran Transit gateway Connect dan rekan Transit Gateway Connect](#) di Panduan Transit Gateway.

- Untuk detail API, lihat [DescribeTransitGatewayConnectPeers](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-connects

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-connects`.

### AWS CLI

Untuk mendeskripsikan lampiran Connect gateway transit

`describe-transit-gateway-connects` Contoh berikut menjelaskan lampiran Connect yang ditentukan.

```

aws ec2 describe-transit-gateway-connects \
  --transit-gateway-attachment-ids tgw-attach-037012e5dcEXAMPLE

```

### Output:

```

{
  "TransitGatewayConnects": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",

```

```

    "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "State": "available",
    "CreationTime": "2021-03-09T19:59:17+00:00",
    "Options": {
      "Protocol": "gre"
    },
    "Tags": []
  }
]
}

```

Untuk informasi selengkapnya, lihat [Lampiran Transit gateway Connect dan rekan Transit Gateway Connect](#) di Panduan Transit Gateway.

- Untuk detail API, lihat [DescribeTransitGatewayConnects](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-multicast-domains

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-multicast-domains`.

### AWS CLI

Untuk mendeskripsikan domain multicast gateway transit Anda

`describe-transit-gateway-multicast-domains` Contoh berikut menampilkan detail untuk semua domain multicast gateway transit Anda.

```
aws ec2 describe-transit-gateway-multicast-domains
```

Output:

```

{
  "TransitGatewayMulticastDomains": [
    {
      "TransitGatewayMulticastDomainId": "tgw-mcast-domain-000fb24d04EXAMPLE",
      "TransitGatewayId": "tgw-0bf0bfffefaEXAMPLE",
      "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-east-1:123456789012:transit-gateway-multicast-domain/tgw-mcast-domain-000fb24d04EXAMPLE",
    }
  ]
}

```

```

    "OwnerId": "123456789012",
    "Options": {
      "Icmpv2Support": "disable",
      "StaticSourcesSupport": "enable",
      "AutoAcceptSharedAssociations": "disable"
    },
    "State": "available",
    "CreationTime": "2019-12-10T18:32:50+00:00",
    "Tags": [
      {
        "Key": "Name",
        "Value": "mc1"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Transit Gateways.

- Untuk detail API, lihat [DescribeTransitGatewayMulticastDomains](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-peering-attachments

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-peering-attachments`.

### AWS CLI

Untuk menggambarkan lampiran peering gateway transit Anda

`describe-transit-gateway-peering-attachments` Contoh berikut menampilkan detail untuk semua lampiran peering gateway transit Anda.

```
aws ec2 describe-transit-gateway-peering-attachments
```

Output:

```

{
  "TransitGatewayPeeringAttachments": [
    {

```



```

    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
    "RequesterTgwInfo": {
      "TransitGatewayId": "tgw-123abc05e04123abc",
      "OwnerId": "123456789012",
      "Region": "us-west-2"
    },
    "AcceptorTgwInfo": {
      "TransitGatewayId": "tgw-11223344aabbcc112",
      "OwnerId": "123456789012",
      "Region": "us-east-2"
    },
    "State": "pendingAcceptance",
    "CreationTime": "2019-12-09T11:38:05.000Z",
    "Tags": []
  }
]
}

```

Untuk informasi selengkapnya, lihat [Lampiran Pengintip Transit Gateway](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DescribeTransitGatewayPeeringAttachments](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-policy-tables

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-policy-tables`.

### AWS CLI

Untuk menjelaskan tabel kebijakan gateway transit

`describe-transit-gateway-policy-tables` Contoh berikut menjelaskan tabel kebijakan gateway transit yang ditentukan.

```

aws ec2 describe-transit-gateway-policy-tables \
  --transit-gateway-policy-table-ids tgw-ptb-0a16f134b78668a81

```

Output:

```
{
```

```

    "TransitGatewayPolicyTables": [
      {
        "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
        "TransitGatewayId": "tgw-067f8505c18f0bd6e",
        "State": "available",
        "CreationTime": "2023-11-28T16:36:43+00:00",
        "Tags": []
      }
    ]
  }

```

Untuk informasi selengkapnya, lihat [Tabel kebijakan gateway transit](#) di Panduan Pengguna Gateway Transit.

- Untuk detail API, lihat [DescribeTransitGatewayPolicyTables](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-route-tables

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-route-tables`.

### AWS CLI

Untuk mendeskripsikan tabel rute gateway transit Anda

`describe-transit-gateway-route-tables` Contoh berikut menampilkan detail untuk tabel rute gateway transit Anda.

```
aws ec2 describe-transit-gateway-route-tables
```

Output:

```

{
  "TransitGatewayRouteTables": [
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0ca78a549EXAMPLE",
      "TransitGatewayId": "tgw-0bc994abffEXAMPLE",
      "State": "available",
      "DefaultAssociationRouteTable": true,
      "DefaultPropagationRouteTable": true,
      "CreationTime": "2018-11-28T14:24:49.000Z",
      "Tags": []
    }
  ]
}

```

```

    },
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0e8f48f148EXAMPLE",
      "TransitGatewayId": "tgw-0043d72bb4EXAMPLE",
      "State": "available",
      "DefaultAssociationRouteTable": true,
      "DefaultPropagationRouteTable": true,
      "CreationTime": "2018-11-28T14:24:00.000Z",
      "Tags": []
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DescribeTransitGatewayRouteTables](#) di Referensi AWS CLI Perintah.

## describe-transit-gateway-vpc-attachments

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateway-vpc-attachments`.

### AWS CLI

Untuk menjelaskan lampiran VPC gateway transit Anda

`describe-transit-gateway-vpc-attachments` Contoh berikut menampilkan detail untuk lampiran VPC gateway transit Anda.

```
aws ec2 describe-transit-gateway-vpc-attachments
```

Output:

```

{
  "TransitGatewayVpcAttachments": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-0a08e88308EXAMPLE",
      "TransitGatewayId": "tgw-0043d72bb4EXAMPLE",
      "VpcId": "vpc-0f501f7ee8EXAMPLE",
      "VpcOwnerId": "111122223333",
      "State": "available",
    }
  ]
}

```

```

    "SubnetIds": [
      "subnet-045d586432EXAMPLE",
      "subnet-0a0ad478a6EXAMPLE"
    ],
    "CreationTime": "2019-02-13T11:04:02.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    },
    "Tags": [
      {
        "Key": "Name",
        "Value": "attachment name"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat lampiran VPC Anda](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DescribeTransitGatewayVpcAttachments](#) di Referensi AWS CLI Perintah.

## describe-transit-gateways

Contoh kode berikut menunjukkan cara menggunakan `describe-transit-gateways`.

### AWS CLI

Untuk menggambarkan gateway transit Anda

`describe-transit-gateways` Contoh berikut mengambil detail tentang gateway transit Anda.

```
aws ec2 describe-transit-gateways
```

Output:

```

{
  "TransitGateways": [
    {
      "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
      "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-
gateway/tgw-0262a0e521EXAMPLE",

```

```

    "State": "available",
    "OwnerId": "111122223333",
    "Description": "MyTGW",
    "CreationTime": "2019-07-10T14:02:12.000Z",
    "Options": {
      "AmazonSideAsn": 64516,
      "AutoAcceptSharedAttachments": "enable",
      "DefaultRouteTableAssociation": "enable",
      "AssociationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
      "DefaultRouteTablePropagation": "enable",
      "PropagationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
      "VpnEcmpSupport": "enable",
      "DnsSupport": "enable"
    },
    "Tags": []
  },
  {
    "TransitGatewayId": "tgw-0fb8421e2dEXAMPLE",
    "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-
gateway/tgw-0fb8421e2da853bf3",
    "State": "available",
    "OwnerId": "111122223333",
    "CreationTime": "2019-03-15T22:57:33.000Z",
    "Options": {
      "AmazonSideAsn": 65412,
      "AutoAcceptSharedAttachments": "disable",
      "DefaultRouteTableAssociation": "enable",
      "AssociationDefaultRouteTableId": "tgw-rtb-06a241a3d8EXAMPLE",
      "DefaultRouteTablePropagation": "enable",
      "PropagationDefaultRouteTableId": "tgw-rtb-06a241a3d8EXAMPLE",
      "VpnEcmpSupport": "enable",
      "DnsSupport": "enable"
    },
    "Tags": [
      {
        "Key": "Name",
        "Value": "TGW1"
      }
    ]
  }
]
}

```

- Untuk detail API, lihat [DescribeTransitGateways](#) di Referensi AWS CLI Perintah.

## describe-verified-access-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-verified-access-endpoints`.

### AWS CLI

Untuk mendeskripsikan titik akhir Akses Terverifikasi

`delete-verified-access-endpoints` Contoh berikut menjelaskan titik akhir Akses Terverifikasi yang ditentukan.

```
aws ec2 describe-verified-access-endpoints \
  --verified-access-endpoint-ids vae-066fac616d4d546f2
```

Output:

```
{
  "VerifiedAccessEndpoints": [
    {
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
      "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
      "ApplicationDomain": "example.com",
      "EndpointType": "network-interface",
      "AttachmentType": "vpc",
      "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
      "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
      "SecurityGroupIds": [
        "sg-004915970c4c8f13a"
      ],
      "NetworkInterfaceOptions": {
        "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
        "Protocol": "https",
        "Port": 443
      },
      "Status": {
        "Code": "active"
      },
      "Description": "",
      "CreationTime": "2023-08-25T20:54:43",
```

```
    "LastUpdatedTime": "2023-08-25T22:17:26",
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-va-endpoint"
      }
    ]
  }
]
```

Untuk informasi selengkapnya, lihat [Titik akhir Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DescribeVerifiedAccessEndpoints](#) di Referensi AWS CLI Perintah.

## describe-verified-access-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-verified-access-groups`.

### AWS CLI

Untuk mendeskripsikan grup Akses Terverifikasi

`describe-verified-access-groups` Contoh berikut menjelaskan grup Akses Terverifikasi yang ditentukan.

```
aws ec2 describe-verified-access-groups \
  --verified-access-group-ids vagr-0dbe967baf14b7235
```

Output:

```
{
  "VerifiedAccessGroups": [
    {
      "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "Description": "Testing Verified Access",
      "Owner": "123456789012",
      "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
      "CreationTime": "2023-08-25T19:55:19",
```

```

        "LastUpdatedTime": "2023-08-25T22:17:25",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-va-group"
            }
        ]
    }
]
}

```

Untuk informasi selengkapnya, lihat [Grup Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DescribeVerifiedAccessGroups](#) di Referensi AWS CLI Perintah.

## describe-verified-access-instance-logging-configurations

Contoh kode berikut menunjukkan cara menggunakan `describe-verified-access-instance-logging-configurations`.

### AWS CLI

Untuk menjelaskan konfigurasi logging untuk instance Akses Terverifikasi

`describe-verified-access-instance-logging-configurations` Contoh berikut menjelaskan konfigurasi logging untuk instance Akses Terverifikasi yang ditentukan.

```

aws ec2 describe-verified-access-instance-logging-configurations \
  --verified-access-instance-ids vai-0ce000c0b7643abea

```

Output:

```

{
  "LoggingConfigurations": [
    {
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "AccessLogs": {
        "S3": {
          "Enabled": false
        }
      },
    },
  ],
}

```



```

        "CloudWatchLogs": {
            "Enabled": true,
            "DeliveryStatus": {
                "Code": "success"
            },
            "LogGroup": "my-log-group"
        },
        "KinesisDataFirehose": {
            "Enabled": false
        },
        "LogVersion": "ocsf-1.0.0-rc.2",
        "IncludeTrustContext": false
    }
}
]
}

```

Untuk informasi selengkapnya, lihat [log Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DescribeVerifiedAccessInstanceLoggingConfigurations](#) di Referensi AWS CLI Perintah.

## describe-verified-access-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-verified-access-instances`.

### AWS CLI

Untuk mendeskripsikan instance Akses Terverifikasi

`describe-verified-access-instances` Contoh berikut menjelaskan contoh Akses Terverifikasi yang ditentukan.

```
aws ec2 describe-verified-access-instances \
  --verified-access-instance-ids vai-0ce000c0b7643abea
```

Output:

```
{
  "VerifiedAccessInstances": [
    {
```

```

    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "VerifiedAccessTrustProviders": [
      {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center"
      }
    ],
    "CreationTime": "2023-08-25T18:27:56",
    "LastUpdatedTime": "2023-08-25T19:03:32",
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-ava-instance"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Instans Akses Terverifikasi](#) di Panduan Pengguna Akses AWS Terverifikasi.

- Untuk detail API, lihat [DescribeVerifiedAccessInstances](#) di Referensi AWS CLI Perintah.

## describe-verified-access-trust-providers

Contoh kode berikut menunjukkan cara menggunakan `describe-verified-access-trust-providers`.

### AWS CLI

Untuk mendeskripsikan penyedia kepercayaan Akses Terverifikasi

`describe-verified-access-trust-providers` Contoh berikut menjelaskan penyedia kepercayaan Akses Terverifikasi yang ditentukan.

```

aws ec2 describe-verified-access-trust-providers \
  --verified-access-trust-provider-ids vatp-0bb32de759a3e19e7

```

Output:

```
{
  "VerifiedAccessTrustProviders": [
    {
      "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
      "Description": "Testing Verified Access",
      "TrustProviderType": "user",
      "UserTrustProviderType": "iam-identity-center",
      "PolicyReferenceName": "idc",
      "CreationTime": "2023-08-25T19:00:38",
      "LastUpdatedTime": "2023-08-25T19:03:32",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-va-trust-provider"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Penyedia kepercayaan untuk Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [DescribeVerifiedAccessTrustProviders](#) di Referensi AWS CLI Perintah.

## describe-volume-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-volume-attribute`.

### AWS CLI

Untuk menggambarkan atribut volume

Perintah contoh ini menjelaskan `autoEnableIo` atribut volume dengan ID `vol-049df61146c4d7901`.

Perintah:

```
aws ec2 describe-volume-attribute --volume-id vol-049df61146c4d7901 --attribute autoEnableIO
```

Output:

```
{
  "AutoEnableIO": {
    "Value": false
  },
  "VolumeId": "vol-049df61146c4d7901"
}
```

- Untuk detail API, lihat [DescribeVolumeAttribute](#) di Referensi AWS CLI Perintah.

## describe-volume-status

Contoh kode berikut menunjukkan cara menggunakan `describe-volume-status`.

### AWS CLI

Untuk menggambarkan status dari satu volume

Perintah contoh ini menjelaskan status untuk volume `vol-1234567890abcdef0`.

Perintah:

```
aws ec2 describe-volume-status --volume-ids vol-1234567890abcdef0
```

Output:

```
{
  "VolumeStatuses": [
    {
      "VolumeStatus": {
        "Status": "ok",
        "Details": [
          {
            "Status": "passed",
            "Name": "io-enabled"
          },
          {
            "Status": "not-applicable",
            "Name": "io-performance"
          }
        ]
      },
      "AvailabilityZone": "us-east-1a",
    }
  ]
}
```

```
        "VolumeId": "vol-1234567890abcdef0",
        "Actions": [],
        "Events": []
    }
]
}
```

Untuk menggambarkan status volume yang terganggu

Perintah contoh ini menjelaskan status untuk semua volume yang terganggu. Dalam contoh output ini, tidak ada volume yang terganggu.

Perintah:

```
aws ec2 describe-volume-status --filters Name=volume-status.status,Values=impaired
```

Output:

```
{
  "VolumeStatuses": []
}
```

Jika Anda memiliki volume dengan pemeriksaan status gagal (status terganggu), lihat [Bekerja dengan Volume yang Terganggu](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeVolumeStatus](#) di Referensi AWS CLI Perintah.

## describe-volumes-modifications

Contoh kode berikut menunjukkan cara menggunakan `describe-volumes-modifications`.

AWS CLI

Untuk menggambarkan status modifikasi untuk volume

`describe-volumes-modifications` Contoh berikut menjelaskan status modifikasi volume volume volume yang ditentukan.

```
aws ec2 describe-volumes-modifications \
  --volume-ids vol-1234567890abcdef0
```

**Output:**

```
{
  "VolumeModification": {
    "TargetSize": 150,
    "TargetVolumeType": "io1",
    "ModificationState": "optimizing",
    "VolumeId": "vol-1234567890abcdef0",
    "TargetIops": 100,
    "StartTime": "2019-05-17T11:27:19.000Z",
    "Progress": 70,
    "OriginalVolumeType": "io1",
    "OriginalIops": 100,
    "OriginalSize": 100
  }
}
```

- Untuk detail API, lihat [DescribeVolumesModifications](#) di Referensi AWS CLI Perintah.

**describe-volumes**

Contoh kode berikut menunjukkan cara menggunakan `describe-volumes`.

**AWS CLI**

Contoh 1: Untuk mendeskripsikan volume

`describe-volumes` Contoh berikut menjelaskan volume yang ditentukan di Wilayah saat ini.

```
aws ec2 describe-volumes \
  --volume-ids vol-049df61146c4d7901 vol-1234567890abcdef0
```

**Output:**

```
{
  "Volumes": [
    {
      "AvailabilityZone": "us-east-1a",
      "Attachments": [
        {
          "AttachTime": "2013-12-18T22:35:00.000Z",
          "InstanceId": "i-1234567890abcdef0",
```

```

        "VolumeId": "vol-049df61146c4d7901",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
    }
],
"Encrypted": true,
"KmsKeyId": "arn:aws:kms:us-east-2a:123456789012:key/8c5b2c63-b9bc-45a3-
a87a-5513eEXAMPLE,
"VolumeType": "gp2",
"VolumeId": "vol-049df61146c4d7901",
"State": "in-use",
"Iops": 100,
"SnapshotId": "snap-1234567890abcdef0",
"CreateTime": "2019-12-18T22:35:00.084Z",
"Size": 8
},
{
    "AvailabilityZone": "us-east-1a",
    "Attachments": [],
    "Encrypted": false,
    "VolumeType": "gp2",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "available",
    "Iops": 300,
    "SnapshotId": "",
    "CreateTime": "2020-02-27T00:02:41.791Z",
    "Size": 100
}
]
}

```

Contoh 2: Untuk menggambarkan volume yang dilampirkan ke instance tertentu

`describe-volumes` Contoh berikut menjelaskan semua volume yang keduanya dilampirkan ke instance tertentu dan diatur untuk menghapus ketika instance berakhir.

```

aws ec2 describe-volumes \
  --region us-east-1 \
  --filters Name=attachment.instance-id,Values=i-1234567890abcdef0
Name=attachment.delete-on-termination,Values=true

```

Untuk contoh output `describe-volumes`, lihat Contoh 1.

### Contoh 3: Untuk menjelaskan volume yang tersedia di Availability Zone tertentu

`describe-volumes` Contoh berikut menjelaskan semua volume yang memiliki status `available` dan berada di Availability Zone yang ditentukan.

```
aws ec2 describe-volumes \  
  --filters Name=status,Values=available Name=availability-zone,Values=us-east-1a
```

Untuk contoh output `describe-volumes`, lihat Contoh 1.

### Contoh 4: Untuk menggambarkan volume berdasarkan tag

`describe-volumes` Contoh berikut menjelaskan semua volume yang memiliki kunci tag `Name` dan nilai yang dimulai dengan `Test`. Output kemudian disaring dengan kueri yang hanya menampilkan tag dan ID volume.

```
aws ec2 describe-volumes \  
  --filters Name=tag:Name,Values=Test* \  
  --query "Volumes[*].{ID:VolumeId,Tag:Tags}"
```

### Output:

```
[  
  {  
    "Tag": [  
      {  
        "Value": "Test2",  
        "Key": "Name"  
      }  
    ],  
    "ID": "vol-1234567890abcdef0"  
  },  
  {  
    "Tag": [  
      {  
        "Value": "Test1",  
        "Key": "Name"  
      }  
    ],  
    "ID": "vol-049df61146c4d7901"  
  }  
]
```



```
] ]
```

Untuk contoh tambahan menggunakan filter tanda, lihat [Bekerja dengan tanda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DescribeVolumes](#) di Referensi AWS CLI Perintah.

## describe-vpc-attribute

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-attribute`.

### AWS CLI

Untuk menggambarkan `enableDnsSupport` atribut

Contoh ini menjelaskan `enableDnsSupport` atribut. Atribut ini menunjukkan apakah resolusi DNS diaktifkan untuk VPC. Jika atribut `inittrue`, server DNS Amazon menyelesaikan nama host DNS untuk instance Anda ke alamat IP yang sesuai; jika tidak, tidak.

Perintah:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-a01106c2 --attribute enableDnsSupport
```

Output:

```
{
  "VpcId": "vpc-a01106c2",
  "EnableDnsSupport": {
    "Value": true
  }
}
```

Untuk menggambarkan `enableDnsHostnames` atribut

Contoh ini menjelaskan `enableDnsHostnames` atribut. Atribut ini menunjukkan apakah instance yang diluncurkan di VPC mendapatkan nama host DNS. Jika atribut `inittrue`, instance di VPC mendapatkan nama host DNS; jika tidak, mereka tidak.

Perintah:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-a01106c2 --attribute enableDnsHostnames
```

**Output:**

```
{
  "VpcId": "vpc-a01106c2",
  "EnableDnsHostnames": {
    "Value": true
  }
}
```

- Untuk detail API, lihat [DescribeVpcAttribute](#) di Referensi AWS CLI Perintah.

**describe-vpc-classic-link-dns-support**

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-classic-link-dns-support`.

**AWS CLI**

Untuk menggambarkan dukungan ClassicLink DNS untuk VPC Anda

Contoh ini menjelaskan status dukungan ClassicLink DNS dari semua VPC Anda.

**Perintah:**

```
aws ec2 describe-vpc-classic-link-dns-support
```

**Output:**

```
{
  "Vpcs": [
    {
      "VpcId": "vpc-88888888",
      "ClassicLinkDnsSupported": true
    },
    {
      "VpcId": "vpc-1a2b3c4d",
      "ClassicLinkDnsSupported": false
    }
  ]
}
```

- Untuk detail API, lihat [DescribeVpcClassicLinkDnsSupport](#) di Referensi AWS CLI Perintah.

## describe-vpc-classic-link

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-classic-link`.

### AWS CLI

Untuk menggambarkan ClassicLink status VPC Anda

Contoh ini mencantumkan ClassicLink status `vpc-88888888`.

Perintah:

```
aws ec2 describe-vpc-classic-link --vpc-id vpc-88888888
```

Output:

```
{
  "Vpcs": [
    {
      "ClassicLinkEnabled": true,
      "VpcId": "vpc-88888888",
      "Tags": [
        {
          "Value": "classiclinkvpc",
          "Key": "Name"
        }
      ]
    }
  ]
}
```

Contoh ini hanya mencantumkan VPC yang diaktifkan untuk Classiclink (nilai filter `is-classic-link-enabled` disetel ke). `true`

Perintah:

```
aws ec2 describe-vpc-classic-link --filter "Name=is-classic-link-enabled,Values=true"
```

- Untuk detail API, lihat [DescribeVpcClassicLink](#) di Referensi AWS CLI Perintah.

## describe-vpc-endpoint-connection-notifications

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-endpoint-connection-notifications`.

### AWS CLI

Untuk menjelaskan pemberitahuan koneksi titik akhir

`describe-vpc-endpoint-connection-notifications` Contoh berikut menjelaskan semua pemberitahuan koneksi titik akhir Anda.

```
aws ec2 describe-vpc-endpoint-connection-notifications
```

Output:

```
{
  "ConnectionNotificationSet": [
    {
      "ConnectionNotificationState": "Enabled",
      "ConnectionNotificationType": "Topic",
      "ConnectionEvents": [
        "Accept",
        "Reject",
        "Delete",
        "Connect"
      ],
      "ConnectionNotificationId": "vpce-nfn-04bcb952bc8af7abc",
      "ConnectionNotificationArn": "arn:aws:sns:us-east-1:123456789012:VpceNotification",
      "VpcEndpointId": "vpce-0324151a02f327123"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeVpcEndpointConnectionNotifications](#) di Referensi AWS CLI Perintah.

## describe-vpc-endpoint-connections

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-endpoint-connections`.

## AWS CLI

Untuk menggambarkan koneksi titik akhir VPC

Contoh ini menjelaskan koneksi titik akhir antarmuka ke layanan titik akhir Anda dan memfilter hasilnya untuk menampilkan titik akhir yang ada. PendingAcceptance

Perintah:

```
aws ec2 describe-vpc-endpoint-connections --filters Name=vpc-endpoint-  
state,Values=pendingAcceptance
```

Output:

```
{  
  "VpcEndpointConnections": [  
    {  
      "VpcEndpointId": "vpce-0abed31004e618123",  
      "ServiceId": "vpce-svc-0abced088d20def56",  
      "CreationTimestamp": "2017-11-30T10:00:24.350Z",  
      "VpcEndpointState": "pendingAcceptance",  
      "VpcEndpointOwner": "123456789012"  
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeVpcEndpointConnections](#) di Referensi AWS CLI Perintah.

## describe-vpc-endpoint-service-configurations

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-endpoint-service-configurations`.

## AWS CLI

Untuk menjelaskan konfigurasi layanan endpoint

`describe-vpc-endpoint-service-configurations` Contoh berikut menjelaskan konfigurasi layanan endpoint Anda.

```
aws ec2 describe-vpc-endpoint-service-configurations
```

## Output:

```
{
  "ServiceConfigurations": [
    {
      "ServiceType": [
        {
          "ServiceType": "GatewayLoadBalancer"
        }
      ],
      "ServiceId": "vpce-svc-012d33a1c4321cab",
      "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-
svc-012d33a1c4321cab",
      "ServiceState": "Available",
      "AvailabilityZones": [
        "us-east-1d"
      ],
      "AcceptanceRequired": false,
      "ManagesVpcEndpoints": false,
      "GatewayLoadBalancerArns": [
        "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
gwy/GWLBService/123210844e429123"
      ],
      "Tags": []
    },
    {
      "ServiceType": [
        {
          "ServiceType": "Interface"
        }
      ],
      "ServiceId": "vpce-svc-123cab125efa123",
      "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123cab125efa123",
      "ServiceState": "Available",
      "AvailabilityZones": [
        "us-east-1a"
      ],
      "AcceptanceRequired": true,
      "ManagesVpcEndpoints": false,
      "NetworkLoadBalancerArns": [
        "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
net/NLBforService/1238753950b25123"
      ],
      "BaseEndpointDnsNames": [
```

```

        "vpce-svc-123cab125efa123.us-east-1.vpce.amazonaws.com"
    ],
    "PrivateDnsName": "example.com",
    "PrivateDnsNameConfiguration": {
        "State": "failed",
        "Type": "TXT",
        "Value": "vpce:qUAtH3FdeABCpUiXabc",
        "Name": "_1d367jvbg34znqvyefrj"
    },
    "Tags": []
}
]
}

```

Untuk informasi selengkapnya, lihat [layanan titik akhir VPC di Panduan](#) Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeVpcEndpointServiceConfigurations](#) di Referensi AWS CLI Perintah.

## describe-vpc-endpoint-service-permissions

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-endpoint-service-permissions`.

### AWS CLI

Untuk menjelaskan izin layanan titik akhir

Contoh ini menjelaskan izin untuk layanan endpoint yang ditentukan.

Perintah:

```
aws ec2 describe-vpc-endpoint-service-permissions --service-id vpce-svc-03d5ebb7d9579a2b3
```

Output:

```
{
  "AllowedPrincipals": [
    {
      "PrincipalType": "Account",
      "Principal": "arn:aws:iam::123456789012:root"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [DescribeVpcEndpointServicePermissions](#) di Referensi AWS CLI Perintah.

## describe-vpc-endpoint-services

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-endpoint-services`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan semua layanan titik akhir VPC

Contoh "describe-vpc-endpoint-services" berikut mencantumkan semua layanan titik akhir VPC untuk suatu Wilayah. AWS

```
aws ec2 describe-vpc-endpoint-services
```

Output:

```
{
  "ServiceDetails": [
    {
      "ServiceType": [
        {
          "ServiceType": "Gateway"
        }
      ],
      "AcceptanceRequired": false,
      "ServiceName": "com.amazonaws.us-east-1.dynamodb",
      "VpcEndpointPolicySupported": true,
      "Owner": "amazon",
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",
        "us-east-1d",
        "us-east-1e",
        "us-east-1f"
      ],
      "BaseEndpointDnsNames": [
        "dynamodb.us-east-1.amazonaws.com"
      ]
    }
  ]
}
```



```
]
},
{
  "ServiceType": [
    {
      "ServiceType": "Interface"
    }
  ],
  "PrivateDnsName": "ec2.us-east-1.amazonaws.com",
  "ServiceName": "com.amazonaws.us-east-1.ec2",
  "VpcEndpointPolicySupported": false,
  "Owner": "amazon",
  "AvailabilityZones": [
    "us-east-1a",
    "us-east-1b",
    "us-east-1c",
    "us-east-1d",
    "us-east-1e",
    "us-east-1f"
  ],
  "AcceptanceRequired": false,
  "BaseEndpointDnsNames": [
    "ec2.us-east-1.vpce.amazonaws.com"
  ]
},
{
  "ServiceType": [
    {
      "ServiceType": "Interface"
    }
  ],
  "PrivateDnsName": "ssm.us-east-1.amazonaws.com",
  "ServiceName": "com.amazonaws.us-east-1.ssm",
  "VpcEndpointPolicySupported": true,
  "Owner": "amazon",
  "AvailabilityZones": [
    "us-east-1a",
    "us-east-1b",
    "us-east-1c",
    "us-east-1d",
    "us-east-1e"
  ],
  "AcceptanceRequired": false,
  "BaseEndpointDnsNames": [
```

```

        "ssm.us-east-1.vpce.amazonaws.com"
    ]
}
],
"ServiceNames": [
    "com.amazonaws.us-east-1.dynamodb",
    "com.amazonaws.us-east-1.ec2",
    "com.amazonaws.us-east-1.ec2messages",
    "com.amazonaws.us-east-1.elasticloadbalancing",
    "com.amazonaws.us-east-1.kinesis-streams",
    "com.amazonaws.us-east-1.s3",
    "com.amazonaws.us-east-1.ssm"
]
}

```

Untuk informasi selengkapnya, [lihat Melihat nama AWS layanan yang tersedia](#) di Panduan Pengguna untuk AWS PrivateLink.

Contoh 2: Untuk menjelaskan detail tentang layanan endpoint

Contoh "describe-vpc-endpoint-services" berikut mencantumkan rincian service endpoint antarmuka Amazon S3

```

aws ec2 describe-vpc-endpoint-services \
  --filter "Name=service-type,Values=Interface" Name=service-
name,Values=com.amazonaws.us-east-1.s3

```

Output:

```

{
  "ServiceDetails": [
    {
      "ServiceName": "com.amazonaws.us-east-1.s3",
      "ServiceId": "vpce-svc-081d84efcdEXAMPLE",
      "ServiceType": [
        {
          "ServiceType": "Interface"
        }
      ],
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",

```

```

        "us-east-1d",
        "us-east-1e",
        "us-east-1f"
    ],
    "Owner": "amazon",
    "BaseEndpointDnsNames": [
        "s3.us-east-1.vpce.amazonaws.com"
    ],
    "VpcEndpointPolicySupported": true,
    "AcceptanceRequired": false,
    "ManagesVpcEndpoints": false,
    "Tags": []
  }
],
"ServiceNames": [
  "com.amazonaws.us-east-1.s3"
]
}

```

Untuk informasi selengkapnya, [lihat Melihat nama AWS layanan yang tersedia](#) di Panduan Pengguna untuk AWS PrivateLink.

- Untuk detail API, lihat [DescribeVpcEndpointServices](#) di Referensi AWS CLI Perintah.

## describe-vpc-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-endpoints`.

### AWS CLI

Untuk mendeskripsikan titik akhir VPC Anda

`describe-vpc-endpoints` Contoh berikut menampilkan detail untuk semua titik akhir VPC Anda.

```
aws ec2 describe-vpc-endpoints
```

Output:

```
{
  "VpcEndpoints": [
    {
```

```

    "PolicyDocument": "{\\"Version\\":\\"2008-10-17\\",\\"Statement\\":[{\\"Effect
\\":\\"Allow\\",\\"Principal\\":\\"*\\" ,\\"Action\\":\\"*\\" ,\\"Resource\\":\\"*\\"}]}",
    "VpcId": "vpc-aabb1122",
    "NetworkInterfaceIds": [],
    "SubnetIds": [],
    "PrivateDnsEnabled": true,
    "State": "available",
    "ServiceName": "com.amazonaws.us-east-1.dynamodb",
    "RouteTableIds": [
        "rtb-3d560345"
    ],
    "Groups": [],
    "VpcEndpointId": "vpce-032a826a",
    "VpcEndpointType": "Gateway",
    "CreationTimestamp": "2017-09-05T20:41:28Z",
    "DnsEntries": [],
    "OwnerId": "123456789012"
},
{
    "PolicyDocument": "{\n  \\"Statement\\": [\n    {\n      \\"Action\\": \\"*
\\", \n      \\"Effect\\": \\"Allow\\", \n      \\"Principal\\": \\"*\\" , \n      \\"Resource
\\": \\"*\\" \n    }]\n}",
    "VpcId": "vpc-1a2b3c4d",
    "NetworkInterfaceIds": [
        "eni-2ec2b084",
        "eni-1b4a65cf"
    ],
    "SubnetIds": [
        "subnet-d6fcaa8d",
        "subnet-7b16de0c"
    ],
    "PrivateDnsEnabled": false,
    "State": "available",
    "ServiceName": "com.amazonaws.us-east-1.elasticloadbalancing",
    "RouteTableIds": [],
    "Groups": [
        {
            "GroupName": "default",
            "GroupId": "sg-54e8bf31"
        }
    ],
    "VpcEndpointId": "vpce-0f89a33420c1931d7",
    "VpcEndpointType": "Interface",
    "CreationTimestamp": "2017-09-05T17:55:27.583Z",

```

```

    "DnsEntries": [
      {
        "HostedZoneId": "Z7HUB22UULQXV",
        "DnsName": "vpce-0f89a33420c1931d7-
bluzidnv.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
      },
      {
        "HostedZoneId": "Z7HUB22UULQXV",
        "DnsName": "vpce-0f89a33420c1931d7-bluzidnv-us-
east-1b.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
      },
      {
        "HostedZoneId": "Z7HUB22UULQXV",
        "DnsName": "vpce-0f89a33420c1931d7-bluzidnv-us-
east-1a.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
      }
    ],
    "OwnerId": "123456789012"
  },
  {
    "VpcEndpointId": "vpce-aabbaabbaabbaabba",
    "VpcEndpointType": "GatewayLoadBalancer",
    "VpcId": "vpc-111122223333aabb",
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-
svc-123123a1c43abc123",
    "State": "available",
    "SubnetIds": [
      "subnet-0011aabbcc2233445"
    ],
    "RequesterManaged": false,
    "NetworkInterfaceIds": [
      "eni-01010120203030405"
    ],
    "CreationTimestamp": "2020-11-11T08:06:03.522Z",
    "Tags": [],
    "OwnerId": "123456789012"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Titik akhir VPC](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DescribeVpcEndpoints](#) di Referensi AWS CLI Perintah.

## describe-vpc-peering-connections

Contoh kode berikut menunjukkan cara menggunakan `describe-vpc-peering-connections`.

### AWS CLI

Untuk menggambarkan koneksi peering VPC Anda

Contoh ini menjelaskan semua koneksi peering VPC Anda.

Perintah:

```
aws ec2 describe-vpc-peering-connections
```

Output:

```
{
  "VpcPeeringConnections": [
    {
      "Status": {
        "Message": "Active",
        "Code": "active"
      },
      "Tags": [
        {
          "Value": "Peering-1",
          "Key": "Name"
        }
      ],
      "AccepterVpcInfo": {
        "OwnerId": "111122223333",
        "VpcId": "vpc-1a2b3c4d",
        "CidrBlock": "10.0.1.0/28"
      },
      "VpcPeeringConnectionId": "pcx-11122233",
      "RequesterVpcInfo": {
        "PeeringOptions": {
          "AllowEgressFromLocalVpcToRemoteClassicLink": false,
          "AllowEgressFromLocalClassicLinkToRemoteVpc": false
        },
        "OwnerId": "444455556666",
        "VpcId": "vpc-123abc45",
        "CidrBlock": "192.168.0.0/16"
      }
    }
  ]
}
```

```

    }
  },
  {
    "Status": {
      "Message": "Pending Acceptance by 444455556666",
      "Code": "pending-acceptance"
    },
    "Tags": [],
    "RequesterVpcInfo": {
      "PeeringOptions": {
        "AllowEgressFromLocalVpcToRemoteClassicLink": false,
        "AllowEgressFromLocalClassicLinkToRemoteVpc": false
      },
      "OwnerId": "444455556666",
      "VpcId": "vpc-11aa22bb",
      "CidrBlock": "10.0.0.0/28"
    },
    "VpcPeeringConnectionId": "pcx-abababab",
    "ExpirationTime": "2014-04-03T09:12:43.000Z",
    "AccepterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-33cc44dd"
    }
  }
]
}

```

Untuk menggambarkan koneksi peering VPC tertentu

Contoh ini menjelaskan semua koneksi peering VPC Anda yang berada dalam status penerimaan tertunda.

Perintah:

```
aws ec2 describe-vpc-peering-connections --filters Name=status-code,Values=pending-acceptance
```

Contoh ini menjelaskan semua koneksi peering VPC Anda yang memiliki tag Owner=Finance.

Perintah:

```
aws ec2 describe-vpc-peering-connections --filters Name=tag:Owner,Values=Finance
```

Contoh ini menjelaskan semua koneksi peering VPC yang Anda minta untuk VPC yang ditentukan, vpc-1a2b3c4d.

Perintah:

```
aws ec2 describe-vpc-peering-connections --filters Name=requester-vpc-info.vpc-id,Values=vpc-1a2b3c4d
```

- Untuk detail API, lihat [DescribeVpcPeeringConnections](#) di Referensi AWS CLI Perintah.

## describe-vpcs

Contoh kode berikut menunjukkan cara menggunakan `describe-vpcs`.

AWS CLI

Contoh 1: Untuk menjelaskan semua VPC Anda

Contoh `describe-vpcs` berikut mengambil detail tentang VPC Anda.

```
aws ec2 describe-vpcs
```

Output:

```
{
  "Vpcs": [
    {
      "CidrBlock": "30.1.0.0/16",
      "DhcpOptionsId": "dopt-19edf471",
      "State": "available",
      "VpcId": "vpc-0e9801d129EXAMPLE",
      "OwnerId": "111122223333",
      "InstanceTenancy": "default",
      "CidrBlockAssociationSet": [
        {
          "AssociationId": "vpc-cidr-assoc-062c64cfafEXAMPLE",
          "CidrBlock": "30.1.0.0/16",
          "CidrBlockState": {
            "State": "associated"
          }
        }
      ]
    }
  ],
}
```



```

    "IsDefault": false,
    "Tags": [
      {
        "Key": "Name",
        "Value": "Not Shared"
      }
    ]
  },
  {
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-19edf471",
    "State": "available",
    "VpcId": "vpc-06e4ab6c6cEXAMPLE",
    "OwnerId": "222222222222",
    "InstanceTenancy": "default",
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-00b17b4eddEXAMPLE",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ]
  },
  "IsDefault": false,
  "Tags": [
    {
      "Key": "Name",
      "Value": "Shared VPC"
    }
  ]
}
]
}

```

Contoh 2: Untuk menjelaskan VPC tertentu

Contoh `describe-vpcs` berikut mengambil detail untuk VPC tertentu.

```

aws ec2 describe-vpcs \
  --vpc-ids vpc-06e4ab6c6cEXAMPLE

```

Output:

```

{
  "Vpcs": [
    {
      "CidrBlock": "10.0.0.0/16",
      "DhcpOptionsId": "dopt-19edf471",
      "State": "available",
      "VpcId": "vpc-06e4ab6c6cEXAMPLE",
      "OwnerId": "111122223333",
      "InstanceTenancy": "default",
      "CidrBlockAssociationSet": [
        {
          "AssociationId": "vpc-cidr-assoc-00b17b4eddEXAMPLE",
          "CidrBlock": "10.0.0.0/16",
          "CidrBlockState": {
            "State": "associated"
          }
        }
      ],
      "IsDefault": false,
      "Tags": [
        {
          "Key": "Name",
          "Value": "Shared VPC"
        }
      ]
    }
  ]
}

```

- Untuk detail API, lihat [DescribeVpcs](#) di Referensi AWS CLI Perintah.

## describe-vpn-connections

Contoh kode berikut menunjukkan cara menggunakan `describe-vpn-connections`.

### AWS CLI

Contoh 1: Untuk menggambarkan koneksi VPN Anda

`describe-vpn-connections` Contoh berikut menjelaskan semua koneksi VPN Site-to-Site Anda.

```
aws ec2 describe-vpn-connections
```

**Output:**

```
{
  "VpnConnections": [
    {
      "CustomerGatewayConfiguration": "...configuration information...",
      "CustomerGatewayId": "cgw-01234567abcde1234",
      "Category": "VPN",
      "State": "available",
      "Type": "ipsec.1",
      "VpnConnectionId": "vpn-1122334455aabbccd",
      "TransitGatewayId": "tgw-00112233445566aab",
      "Options": {
        "EnableAcceleration": false,
        "StaticRoutesOnly": true,
        "LocalIpv4NetworkCidr": "0.0.0.0/0",
        "RemoteIpv4NetworkCidr": "0.0.0.0/0",
        "TunnelInsideIpVersion": "ipv4"
      },
      "Routes": [],
      "Tags": [
        {
          "Key": "Name",
          "Value": "CanadaVPN"
        }
      ],
      "VgwTelemetry": [
        {
          "AcceptedRouteCount": 0,
          "LastStatusChange": "2020-07-29T10:35:11.000Z",
          "OutsideIpAddress": "203.0.113.3",
          "Status": "DOWN",
          "StatusMessage": ""
        },
        {
          "AcceptedRouteCount": 0,
          "LastStatusChange": "2020-09-02T09:09:33.000Z",
          "OutsideIpAddress": "203.0.113.5",
          "Status": "UP",
          "StatusMessage": ""
        }
      ]
    }
  ]
}
```

```
    ]
  }
]
}
```

Untuk informasi selengkapnya, lihat [Cara kerja AWS VPN Site-to-Site di Panduan Pengguna VPN AWS Site-to-Site](#).

Contoh 2: Untuk menggambarkan koneksi VPN Anda yang tersedia

`describe-vpn-connections` Contoh berikut menjelaskan koneksi VPN Site-to-Site Anda dengan status `available`

```
aws ec2 describe-vpn-connections \
  --filters "Name=state,Values=available"
```

Untuk informasi selengkapnya, lihat [Cara kerja AWS VPN Site-to-Site di Panduan Pengguna VPN AWS Site-to-Site](#).

- Untuk detail API, lihat [DescribeVpnConnections](#) di Referensi AWS CLI Perintah.

## **describe-vpn-gateways**

Contoh kode berikut menunjukkan cara menggunakan `describe-vpn-gateways`.

### AWS CLI

Untuk menggambarkan gateway pribadi virtual Anda

Contoh ini menjelaskan gateway pribadi virtual Anda.

Perintah:

```
aws ec2 describe-vpn-gateways
```

Output:

```
{
  "VpnGateways": [
    {
      "State": "available",
      "Type": "ipsec.1",
```

```
    "VpnGatewayId": "vgw-f211f09b",
    "VpcAttachments": [
      {
        "State": "attached",
        "VpcId": "vpc-98eb5ef5"
      }
    ]
  },
  {
    "State": "available",
    "Type": "ipsec.1",
    "VpnGatewayId": "vgw-9a4cacf3",
    "VpcAttachments": [
      {
        "State": "attaching",
        "VpcId": "vpc-a01106c2"
      }
    ]
  }
]
```

- Untuk detail API, lihat [DescribeVpnGateways](#) di Referensi AWS CLI Perintah.

## detach-classic-link-vpc

Contoh kode berikut menunjukkan cara menggunakan `detach-classic-link-vpc`.

### AWS CLI

Untuk memutuskan tautan (melepaskan) instance EC2-Classic dari VPC

Contoh ini membatalkan tautan instance `i-0598c7d356eba48d7` dari VPC `vpc-88888888`.

Perintah:

```
aws ec2 detach-classic-link-vpc --instance-id i-0598c7d356eba48d7 --vpc-id
vpc-88888888
```

Output:

```
{
```

```
"Return": true
}
```

- Untuk detail API, lihat [DetachClassicLinkVpc](#) di Referensi AWS CLI Perintah.

## detach-internet-gateway

Contoh kode berikut menunjukkan cara menggunakan `detach-internet-gateway`.

### AWS CLI

Untuk melepaskan gateway internet dari VPC Anda

`detach-internet-gateway` Contoh berikut melepaskan gateway internet yang ditentukan dari VPC tertentu.

```
aws ec2 detach-internet-gateway \
  --internet-gateway-id igw-0d0fb496b3EXAMPLE \
  --vpc-id vpc-0a60eb65b4EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Gateway internet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DetachInternetGateway](#) di Referensi AWS CLI Perintah.

## detach-network-interface

Contoh kode berikut menunjukkan cara menggunakan `detach-network-interface`.

### AWS CLI

Untuk melepaskan antarmuka jaringan dari instans Anda

Contoh ini melepaskan antarmuka jaringan yang ditentukan dari contoh yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 detach-network-interface --attachment-id eni-attach-66c4350a
```

- Untuk detail API, lihat [DetachNetworkInterfacedi Referensi AWS CLI Perintah](#).

## detach-verified-access-trust-provider

Contoh kode berikut menunjukkan cara menggunakan `detach-verified-access-trust-provider`.

### AWS CLI

Untuk melepaskan penyedia kepercayaan dari sebuah instance

`detach-verified-access-trust-provider` Contoh berikut melepaskan penyedia kepercayaan Akses Terverifikasi yang ditentukan dari instance Akses Terverifikasi yang ditentukan.

```
aws ec2 detach-verified-access-trust-provider \  
  --verified-access-instance-id vai-0ce000c0b7643abea \  
  --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

### Output:

```
{  
  "VerifiedAccessTrustProvider": {  
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",  
    "Description": "Testing Verified Access",  
    "TrustProviderType": "user",  
    "UserTrustProviderType": "iam-identity-center",  
    "PolicyReferenceName": "idc",  
    "CreationTime": "2023-08-25T19:00:38",  
    "LastUpdatedTime": "2023-08-25T19:00:38"  
  },  
  "VerifiedAccessInstance": {  
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",  
    "Description": "Testing Verified Access",  
    "VerifiedAccessTrustProviders": [],  
    "CreationTime": "2023-08-25T18:27:56",  
    "LastUpdatedTime": "2023-08-25T18:27:56"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Instans Akses Terverifikasi](#) di Panduan Pengguna Akses AWS Terverifikasi.

- Untuk detail API, lihat [DetachVerifiedAccessTrustProvider](#) di Referensi AWS CLI Perintah.

## detach-volume

Contoh kode berikut menunjukkan cara menggunakan `detach-volume`.

### AWS CLI

Untuk melepaskan volume dari sebuah instance

Perintah contoh ini melepaskan volume (`vol-049df61146c4d7901`) dari instance yang dilampirkan.

Perintah:

```
aws ec2 detach-volume --volume-id vol-1234567890abcdef0
```

Output:

```
{
  "AttachTime": "2014-02-27T19:23:06.000Z",
  "InstanceId": "i-1234567890abcdef0",
  "VolumeId": "vol-049df61146c4d7901",
  "State": "detaching",
  "Device": "/dev/sdb"
}
```

- Untuk detail API, lihat [DetachVolume](#) di Referensi AWS CLI Perintah.

## detach-vpn-gateway

Contoh kode berikut menunjukkan cara menggunakan `detach-vpn-gateway`.

### AWS CLI

Untuk melepaskan gateway pribadi virtual dari VPC Anda

Contoh ini melepaskan gateway pribadi virtual yang ditentukan dari VPC yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:



```
aws ec2 detach-vpn-gateway --vpn-gateway-id vgw-9a4cacf3 --vpc-id vpc-a01106c2
```

- Untuk detail API, lihat [DetachVpnGateway](#) di Referensi AWS CLI Perintah.

## disable-address-transfer

Contoh kode berikut menunjukkan cara menggunakan `disable-address-transfer`.

### AWS CLI

Untuk menonaktifkan transfer alamat IP Elastis

`disable-address-transfer` Contoh berikut menonaktifkan transfer alamat IP Elastis untuk alamat IP Elastis yang ditentukan.

```
aws ec2 disable-address-transfer \  
  --allocation-id eipalloc-09ad461b0d03f6aaf
```

Output:

```
{  
  "AddressTransfer": {  
    "PublicIp": "100.21.184.216",  
    "AllocationId": "eipalloc-09ad461b0d03f6aaf",  
    "AddressTransferStatus": "disabled"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mentransfer alamat IP Elastis](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [DisableAddressTransfer](#) di Referensi AWS CLI Perintah.

## disable-aws-network-performance-metric-subscription

Contoh kode berikut menunjukkan cara menggunakan `disable-aws-network-performance-metric-subscription`.

### AWS CLI

Untuk menonaktifkan langganan metrik

`disable-aws-network-performance-metric-subscription` Contoh berikut menonaktifkan pemantauan latensi jaringan agregat antara wilayah sumber dan tujuan yang ditentukan.

```
aws ec2 disable-aws-network-performance-metric-subscription \  
  --source us-east-1 \  
  --destination eu-west-1 \  
  --metric aggregate-latency \  
  --statistic p50
```

Output:

```
{  
  "Output": true  
}
```

Untuk informasi selengkapnya, lihat [Mengelola langganan](#) di Panduan Pengguna Kinerja Infrastruktur.

- Untuk detail API, lihat [DisableAwsNetworkPerformanceMetricSubscription](#) di Referensi AWS CLI Perintah.

## `disable-ebs-encryption-by-default`

Contoh kode berikut menunjukkan cara menggunakan `disable-ebs-encryption-by-default`.

AWS CLI

Untuk menonaktifkan enkripsi EBS secara default

`disable-ebs-encryption-by-default` Contoh berikut menonaktifkan enkripsi EBS secara default untuk AWS akun Anda di Wilayah saat ini.

```
aws ec2 disable-ebs-encryption-by-default
```

Output:

```
{  
  "EbsEncryptionByDefault": false  
}
```

```
}
```

- Untuk detail API, lihat [DisableEbsEncryptionByDefault](#) di Referensi AWS CLI Perintah.

## disable-fast-launch

Contoh kode berikut menunjukkan cara menggunakan `disable-fast-launch`.

### AWS CLI

Untuk menghentikan peluncuran cepat untuk sebuah gambar

`disable-fast-launch` Contoh berikut menghentikan peluncuran cepat pada AMI yang ditentukan, dan membersihkan snapshot yang sudah disediakan sebelumnya.

```
aws ec2 disable-fast-launch \  
  --image-id ami-01234567890abcdef
```

Output:

```
{  
  "ImageId": "ami-01234567890abcdef",  
  "ResourceType": "snapshot",  
  "SnapshotConfiguration": {},  
  "LaunchTemplate": {  
    "LaunchTemplateId": "lt-01234567890abcdef",  
    "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-  
a8c6215d-94e6-441b-9272-dbd1f87b07e2",  
    "Version": "1"  
  },  
  "MaxParallelLaunches": 6,  
  "OwnerId": "0123456789123",  
  "State": "disabling",  
  "StateTransitionReason": "Client.UserInitiated",  
  "StateTransitionTime": "2022-01-27T22:47:29.265000+00:00"  
}
```

Untuk informasi selengkapnya tentang mengonfigurasi AMI Windows untuk peluncuran yang lebih cepat, lihat [Mengonfigurasi AMI Anda untuk peluncuran lebih cepat](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DisableFastLaunch](#) di Referensi AWS CLI Perintah.

## disable-fast-snapshot-restores

Contoh kode berikut menunjukkan cara menggunakan `disable-fast-snapshot-restores`.

### AWS CLI

Untuk menonaktifkan pemulihan snapshot cepat

`disable-fast-snapshot-restores` Contoh berikut menonaktifkan pemulihan snapshot cepat untuk snapshot yang ditentukan di Availability Zone yang ditentukan.

```
aws ec2 disable-fast-snapshot-restores \
  --availability-zones us-east-2a \
  --source-snapshot-ids snap-1234567890abcdef0
```

### Output:

```
{
  "Successful": [
    {
      "SnapshotId": "snap-1234567890abcdef0"
      "AvailabilityZone": "us-east-2a",
      "State": "disabling",
      "StateTransitionReason": "Client.UserInitiated",
      "OwnerId": "123456789012",
      "EnablingTime": "2020-01-25T23:57:49.602Z"
    }
  ],
  "Unsuccessful": []
}
```

- Untuk detail API, lihat [DisableFastSnapshotRestores](#) di Referensi AWS CLI Perintah.

## disable-image-block-public-access

Contoh kode berikut menunjukkan cara menggunakan `disable-image-block-public-access`.

### AWS CLI

Untuk menonaktifkan blokir akses publik untuk AMI di Wilayah yang ditentukan

`disable-image-block-public-access` Contoh berikut menonaktifkan memblokir akses publik untuk AMI di tingkat akun di Wilayah yang ditentukan.

```
aws ec2 disable-image-block-public-access \  
  --region us-east-1
```

Output:

```
{  
  "ImageBlockPublicAccessState": "unblocked"  
}
```

Untuk informasi selengkapnya, lihat [Memblokir akses publik ke AMI Anda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DisableImageBlockPublicAccess](#) di Referensi AWS CLI Perintah.

## disable-image-deprecation

Contoh kode berikut menunjukkan cara menggunakan `disable-image-deprecation`.

AWS CLI

Untuk membatalkan penghentian AMI

`disable-image-deprecation` Contoh berikut membatalkan penghentian AMI, yang menghapus `DeprecationTime` bidang dari output. `describe-images` Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

```
aws ec2 disable-image-deprecation \  
  --image-id ami-1234567890abcdef0
```

Output:

```
{  
  "RequestID": "11aabb229-4eac-35bd-99ed-be587EXAMPLE",  
  "Return": "true"  
}
```

Untuk informasi selengkapnya, lihat Menghentikan AMI < <https://docs.aws.amazon.com/AWS-EC2/latest/ami-deprecate.html#deprecate> UserGuide -ami > di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DisableImageDeprecation](#) di Referensi AWS CLI Perintah.

## **disable-image**

Contoh kode berikut menunjukkan cara menggunakan `disable-image`.

### AWS CLI

Untuk menonaktifkan AMI

`disable-image` Contoh berikut menonaktifkan AMI yang ditentukan.

```
aws ec2 disable-image \  
  --image-id ami-1234567890abcdef0
```

Output:

```
{  
  "Return": "true"  
}
```

Untuk informasi selengkapnya, lihat [Menonaktifkan AMI](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DisableImage](#) di Referensi AWS CLI Perintah.

## **disable-ipam-organization-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `disable-ipam-organization-admin-account`.

### AWS CLI

Untuk menonaktifkan admin IPAM yang didelegasikan

Dalam skenario tertentu, Anda akan mengintegrasikan IPAM dengan AWS Organizations. Ketika Anda melakukannya, akun manajemen AWS Organizations mendelegasikan akun anggota AWS Organizations sebagai admin IPAM.

Dalam contoh ini, Anda adalah akun manajemen AWS Organizations yang mendelegasikan akun admin IPAM dan Anda ingin menonaktifkan akun tersebut dari menjadi admin IPAM.

Anda dapat menggunakan AWS Wilayah mana pun untuk `--region` saat membuat permintaan ini. Anda tidak harus menggunakan Wilayah tempat Anda awalnya mendelegasikan admin,

tempat IPAM dibuat, atau Wilayah operasi IPAM. Jika Anda menonaktifkan akun admin yang didelegasikan, Anda dapat mengaktifkannya kembali kapan saja atau mendelegasikan akun baru sebagai admin IPAM.

`disable-ipam-organization-admin-account` Contoh berikut menonaktifkan admin IPAM yang didelegasikan di akun Anda. AWS

```
aws ec2 disable-ipam-organization-admin-account \  
  --delegated-admin-account-id 320805250157 \  
  --region ap-south-1
```

Output:

```
{  
  "Success": true  
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di AWS Organisasi](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DisableIpamOrganizationAdminAccount](#) di Referensi AWS CLI Perintah.

## **disable-serial-console-access**

Contoh kode berikut menunjukkan cara menggunakan `disable-serial-console-access`.

AWS CLI

Untuk menonaktifkan akses ke konsol serial EC2 untuk akun Anda

`disable-serial-console-access` Contoh berikut menonaktifkan akses akun ke konsol serial.

```
aws ec2 disable-serial-console-access
```

Output:

```
{  
  "SerialConsoleAccessEnabled": false  
}
```

Untuk informasi selengkapnya, lihat [Konsol Serial EC2](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DisableSerialConsoleAccess](#) di Referensi AWS CLI Perintah.

## **disable-transit-gateway-route-table-propagation**

Contoh kode berikut menunjukkan cara menggunakan `disable-transit-gateway-route-table-propagation`.

### AWS CLI

Untuk menonaktifkan lampiran gateway transit untuk menyebarkan rute ke tabel rute propagasi yang ditentukan

`disable-transit-gateway-route-table-propagation` Contoh berikut menonaktifkan lampiran yang ditentukan untuk menyebarkan rute ke tabel rute propagasi yang ditentukan.

```
aws ec2 disable-transit-gateway-route-table-propagation \  
  --transit-gateway-route-table-id tgw-rtb-0a823eddbdeEXAMPLE \  
  --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE
```

### Output:

```
{  
  "Propagation": {  
    "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",  
    "ResourceId": "vpc-4d7de228",  
    "ResourceType": "vpc",  
    "TransitGatewayRouteTableId": "tgw-rtb-0a823eddbdeEXAMPLE",  
    "State": "disabled"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DisableTransitGatewayRouteTablePropagation](#) di Referensi AWS CLI Perintah.

## **disable-vgw-route-propagation**

Contoh kode berikut menunjukkan cara menggunakan `disable-vgw-route-propagation`.



## AWS CLI

Untuk menonaktifkan propagasi rute

Contoh ini menonaktifkan gateway pribadi virtual yang ditentukan dari menyebarkan rute statis ke tabel rute yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 disable-vgw-route-propagation --route-table-id rtb-22574640 --gateway-id vgw-9a4cacf3
```

- Untuk detail API, lihat [DisableVgwRoutePropagation](#) di Referensi AWS CLI Perintah.

## **disable-vpc-classic-link-dns-support**

Contoh kode berikut menunjukkan cara menggunakan `disable-vpc-classic-link-dns-support`.

## AWS CLI

Untuk menonaktifkan dukungan ClassicLink DNS untuk VPC

Contoh ini menonaktifkan dukungan ClassicLink DNS untuk `vpc-88888888`

Perintah:

```
aws ec2 disable-vpc-classic-link-dns-support --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [DisableVpcClassicLinkDnsSupport](#) di Referensi AWS CLI Perintah.

## **disable-vpc-classic-link**

Contoh kode berikut menunjukkan cara menggunakan `disable-vpc-classic-link`.

## AWS CLI

### ClassicLink Untuk menonaktifkan VPC

Contoh ini menonaktifkan ClassicLink untuk vpc-88888888.

Perintah:

```
aws ec2 disable-vpc-classic-link --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [DisableVpcClassicLink](#) di Referensi AWS CLI Perintah.

## disassociate-address

Contoh kode berikut menunjukkan cara menggunakan `disassociate-address`.

### AWS CLI

Untuk melepaskan kaitan alamat IP Elastis di EC2-Classic

Contoh ini melepaskan kaitan alamat IP Elastis dari instans di EC2-Classic. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 disassociate-address --public-ip 198.51.100.0
```

Untuk melepaskan kaitan alamat IP Elastis di EC2-VPC

Contoh ini melepaskan kaitan alamat IP Elastis dari instans di VPC. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 disassociate-address --association-id eipassoc-2bebb745
```

- Untuk detail API, lihat [DisassociateAddress](#) di Referensi AWS CLI Perintah.

## disassociate-client-vpn-target-network

Contoh kode berikut menunjukkan cara menggunakan `disassociate-client-vpn-target-network`.

### AWS CLI

Untuk memisahkan jaringan dari titik akhir Client VPN

`disassociate-client-vpn-target-network` Contoh berikut memisahkan jaringan target yang terkait dengan ID `cvpn-assoc-12312312312312312` asosiasi untuk titik akhir Client VPN yang ditentukan.

```
aws ec2 disassociate-client-vpn-target-network \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --association-id cvpn-assoc-12312312312312312
```

Output:

```
{
  "AssociationId": "cvpn-assoc-12312312312312312",
  "Status": {
    "Code": "disassociating"
  }
}
```

Untuk informasi selengkapnya, lihat [Jaringan Target](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [DisassociateClientVpnTargetNetwork](#) di Referensi AWS CLI Perintah.

## disassociate-iam-instance-profile

Contoh kode berikut menunjukkan cara menggunakan `disassociate-iam-instance-profile`.

### AWS CLI

Untuk memisahkan profil instans IAM

Contoh ini memisahkan profil instans IAM dengan ID asosiasi. `iip-  
assoc-05020b59952902f5f`

Perintah:

```
aws ec2 disassociate-iam-instance-profile --association-id iip-  
assoc-05020b59952902f5f
```

Output:

```
{  
  "IamInstanceProfileAssociation": {  
    "InstanceId": "i-123456789abcde123",  
    "State": "disassociating",  
    "AssociationId": "iip-assoc-05020b59952902f5f",  
    "IamInstanceProfile": {  
      "Id": "AIPAI5IVIHMFFYY2DKV5Y",  
      "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"  
    }  
  }  
}
```

- Untuk detail API, lihat [DisassociateIamInstanceProfile](#) di Referensi AWS CLI Perintah.

## **disassociate-instance-event-window**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-instance-event-window`.

AWS CLI

Contoh 1: Untuk memisahkan satu atau beberapa instance dari jendela acara

`disassociate-instance-event-window` Contoh berikut memisahkan satu atau lebih instance dari jendela acara. Tentukan `instance-event-window-id` parameter untuk menentukan jendela acara. Untuk memisahkan instans, tentukan parameter `association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa ID instans.

```
aws ec2 disassociate-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --association-target i-1234567890
```

```
--association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

### Output:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating"
  }
}
```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 2: Untuk memisahkan tag instance dari jendela acara

`disassociate-instance-event-window` Contoh berikut memisahkan tag instance dari jendela acara. Tentukan `instance-event-window-id` parameter untuk menentukan jendela acara. Untuk memisahkan tanda instans, tentukan parameter `association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa tanda.

```
aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

### Output:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
```

```

        "InstanceIds": [],
        "Tags": [],
        "DedicatedHostIds": []
    },
    "State": "creating"
}
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 3: Untuk memisahkan Host Khusus dari jendela acara

`disassociate-instance-event-window` Contoh berikut memisahkan Host Khusus dari jendela acara. Tentukan `instance-event-window-id` parameter untuk menentukan jendela acara. Untuk memisahkan Host Khusus, tentukan parameter `association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa ID Host Khusus.

```

aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target DedicatedHostIds=h-029fa35a02b99801d

```

Output:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating"
  }
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DisassociateInstanceEventWindow](#) di Referensi AWS CLI Perintah.

## disassociate-ipam-resource-discovery

Contoh kode berikut menunjukkan cara menggunakan `disassociate-ipam-resource-discovery`.

### AWS CLI

Untuk memisahkan penemuan sumber daya dari IPAM

Dalam contoh ini, Anda adalah akun admin yang didelegasikan IPAM dan Anda ingin memisahkan penemuan sumber daya IPAM dari IPAM Anda. Anda menjalankan perintah `describe` dan memperhatikan bahwa `"ResourceDiscoveryStatus": "not-found"` dan Anda ingin memisahkannya dari IPAM Anda untuk memberi ruang bagi asosiasi lain.

`disassociate-ipam-resource-discovery` Contoh berikut memisahkan penemuan sumber daya IPAM di akun Anda. AWS

```
aws ec2 disassociate-ipam-resource-discovery \  
  --ipam-resource-discovery-association-id ipam-res-disco-assoc-04382a6346357cf82 \  
  --region us-east-1
```

Output:

```
{  
  "IpamResourceDiscoveryAssociation": {  
    "OwnerId": "320805250157",  
    "IpamResourceDiscoveryAssociationId": "ipam-res-disco-  
assoc-04382a6346357cf82",  
    "IpamResourceDiscoveryAssociationArn":  
"arn:aws:ec2::320805250157:ipam-resource-discovery-association/ipam-res-disco-  
assoc-04382a6346357cf82",  
    "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",  
    "IpamId": "ipam-005f921c17ebd5107",  
    "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",  
    "IpamRegion": "us-east-1",  
    "IsDefault": false,  
    "ResourceDiscoveryStatus": "not-found",  
    "State": "disassociate-in-progress"
```

```
}
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [DisassociateIpamResourceDiscovery](#) di Referensi AWS CLI Perintah.

## disassociate-nat-gateway-address

Contoh kode berikut menunjukkan cara menggunakan `disassociate-nat-gateway-address`.

### AWS CLI

Untuk memisahkan alamat IP Elastis dari gateway NAT publik

`disassociate-nat-gateway-address` Contoh berikut memisahkan alamat IP Elastis yang ditentukan dari gateway NAT publik yang ditentukan.

```
aws ec2 disassociate-nat-gateway-address \
  --nat-gateway-id nat-1234567890abcdef0 \
  --association-ids eipassoc-0f96bdca17EXAMPLE
```

### Output:

```
{
  "NatGatewayId": "nat-1234567890abcdef0",
  "NatGatewayAddresses": [
    {
      "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
      "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
      "PrivateIp": "10.0.0.74",
      "PublicIp": "3.211.231.218",
      "AssociationId": "eipassoc-0f96bdca17EXAMPLE",
      "IsPrimary": false,
      "Status": "disassociating"
    }
  ]
}
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.



- Untuk detail API, lihat [DisassociateNatGatewayAddress](#) di Referensi AWS CLI Perintah.

## **disassociate-route-table**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-route-table`.

### AWS CLI

Untuk memisahkan tabel rute

Contoh ini memisahkan tabel rute yang ditentukan dari subnet yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 disassociate-route-table --association-id rtbassoc-781d0d1a
```

- Untuk detail API, lihat [DisassociateRouteTable](#) di Referensi AWS CLI Perintah.

## **disassociate-subnet-cidr-block**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-subnet-cidr-block`.

### AWS CLI

Untuk memisahkan blok CIDR IPv6 dari subnet

Contoh ini memisahkan blok IPv6 CIDR dari subnet menggunakan ID asosiasi untuk blok CIDR.

Perintah:

```
aws ec2 disassociate-subnet-cidr-block --association-id subnet-cidr-assoc-3aa54053
```

Output:

```
{
  "SubnetId": "subnet-5f46ec3b",
  "Ipv6CidrBlockAssociation": {
    "Ipv6CidrBlock": "2001:db8:1234:1a00::/64",
```

```

    "AssociationId": "subnet-cidr-assoc-3aa54053",
    "Ipv6CidrBlockState": {
      "State": "disassociating"
    }
  }
}

```

- Untuk detail API, lihat [DisassociateSubnetCidrBlock](#) di Referensi AWS CLI Perintah.

## disassociate-transit-gateway-multicast-domain

Contoh kode berikut menunjukkan cara menggunakan `disassociate-transit-gateway-multicast-domain`.

### AWS CLI

Untuk memisahkan subnet dari domain multicast

`disassociate-transit-gateway-multicast-domain` Contoh berikut memisahkan subnet dari domain multicast yang ditentukan.

```

aws ec2 disassociate-transit-gateway-multicast-domain \
  --transit-gateway-attachment-id tgw-attach-070e571cd1EXAMPLE \
  --subnet-id subnet-000de86e3bEXAMPLE \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE

```

Output:

```

{
  "Associations": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef7EXAMPLE",
    "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
    "ResourceId": "vpc-7EXAMPLE",
    "ResourceType": "vpc",
    "Subnets": [
      {
        "SubnetId": "subnet-000de86e3bEXAMPLE",
        "State": "disassociating"
      }
    ]
  }
}

```

```
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan multicast di Panduan Gerbang Transit](#) '.

- Untuk detail API, lihat [DisassociateTransitGatewayMulticastDomain](#) di Referensi AWS CLI Perintah.

## disassociate-transit-gateway-route-table

Contoh kode berikut menunjukkan cara menggunakan `disassociate-transit-gateway-route-table`.

### AWS CLI

Untuk memisahkan tabel rute gateway transit dari lampiran sumber daya

`disassociate-transit-gateway-route-table` Contoh berikut memisahkan lampiran yang ditentukan dari tabel rute gateway transit.

```
aws ec2 disassociate-transit-gateway-route-table \
  --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE \
  --transit-gateway-attachment-id tgw-attach-08e0bc912cEXAMPLE
```

### Output:

```
{
  "Association": {
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
    "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
    "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "ResourceType": "direct-connect-gateway",
    "State": "disassociating"
  }
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [DisassociateTransitGatewayRouteTable](#) di Referensi AWS CLI Perintah.

## disassociate-vpc-cidr-block

Contoh kode berikut menunjukkan cara menggunakan `disassociate-vpc-cidr-block`.

## AWS CLI

Untuk memisahkan blok CIDR IPv6 dari VPC

Contoh ini memisahkan blok IPv6 CIDR dari VPC menggunakan ID asosiasi untuk blok CIDR.

Perintah:

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-eca54085
```

Output:

```
{
  "Ipv6CidrBlockAssociation": {
    "Ipv6CidrBlock": "2001:db8:1234:1a00::/56",
    "AssociationId": "vpc-cidr-assoc-eca54085",
    "Ipv6CidrBlockState": {
      "State": "disassociating"
    }
  },
  "VpcId": "vpc-a034d6c4"
}
```

Untuk memisahkan blok CIDR IPv4 dari VPC

Contoh ini memisahkan blok CIDR IPv4 dari VPC.

Perintah:

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-0287ac6b
```

Output:

```
{
  "CidrBlockAssociation": {
    "AssociationId": "vpc-cidr-assoc-0287ac6b",
    "CidrBlock": "172.18.0.0/16",
    "CidrBlockState": {
      "State": "disassociating"
    }
  },
}
```

```
"VpcId": "vpc-27621243"
}
```

- Untuk detail API, lihat [DisassociateVpcCidrBlock](#) di Referensi AWS CLI Perintah.

## **enable-address-transfer**

Contoh kode berikut menunjukkan cara menggunakan `enable-address-transfer`.

### AWS CLI

Untuk mengaktifkan transfer alamat IP Elastis

`enable-address-transfer` Contoh berikut memungkinkan transfer alamat IP Elastis untuk alamat IP Elastis yang ditentukan ke akun yang ditentukan.

```
aws ec2 enable-address-transfer \
  --allocation-id eipalloc-09ad461b0d03f6aaf \
  --transfer-account-id 123456789012
```

Output:

```
{
  "AddressTransfer": {
    "PublicIp": "100.21.184.216",
    "AllocationId": "eipalloc-09ad461b0d03f6aaf",
    "TransferAccountId": "123456789012",
    "TransferOfferExpirationTimestamp": "2023-02-22T20:51:01.000Z",
    "AddressTransferStatus": "pending"
  }
}
```

Untuk informasi selengkapnya, lihat [Mentransfer alamat IP Elastis](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [EnableAddressTransfer](#) di Referensi AWS CLI Perintah.

## **enable-aws-network-performance-metric-subscription**

Contoh kode berikut menunjukkan cara menggunakan `enable-aws-network-performance-metric-subscription`.

## AWS CLI

Untuk mengaktifkan langganan metrik

`enable-aws-network-performance-metric-subscription` Contoh berikut memungkinkan pemantauan latensi jaringan agregat antara sumber yang ditentukan dan daerah tujuan.

```
aws ec2 enable-aws-network-performance-metric-subscription \
  --source us-east-1 \
  --destination eu-west-1 \
  --metric aggregate-latency \
  --statistic p50
```

Output:

```
{
  "Output": true
}
```

Untuk informasi selengkapnya, lihat [Mengelola langganan](#) di Panduan Pengguna Kinerja Infrastruktur.

- Untuk detail API, lihat [EnableAwsNetworkPerformanceMetricSubscription](#) di Referensi AWS CLI Perintah.

## **enable-ebs-encryption-by-default**

Contoh kode berikut menunjukkan cara menggunakan `enable-ebs-encryption-by-default`.

## AWS CLI

Untuk mengaktifkan enkripsi EBS secara default

`enable-ebs-encryption-by-default` Contoh berikut memungkinkan enkripsi EBS secara default untuk AWS akun Anda di Wilayah saat ini.

```
aws ec2 enable-ebs-encryption-by-default
```

Output:

```
{
```

```
"EbsEncryptionByDefault": true
}
```

- Untuk detail API, lihat [EnableEbsEncryptionByDefault](#) di Referensi AWS CLI Perintah.

## enable-fast-launch

Contoh kode berikut menunjukkan cara menggunakan `enable-fast-launch`.

### AWS CLI

Untuk memulai peluncuran cepat untuk sebuah gambar

`enable-fast-launch` Contoh berikut memulai peluncuran cepat pada AMI yang ditentukan dan menetapkan jumlah maksimum instance paralel untuk diluncurkan ke 6. Jenis sumber daya yang akan digunakan untuk pra-penyediaan AMI diatur `snapshot`, yang juga merupakan nilai default.

```
aws ec2 enable-fast-launch \
  --image-id ami-01234567890abcdef \
  --max-parallel-launches 6 \
  --resource-type snapshot
```

### Output:

```
{
  "ImageId": "ami-01234567890abcdef",
  "ResourceType": "snapshot",
  "SnapshotConfiguration": {
    "TargetResourceCount": 10
  },
  "LaunchTemplate": {},
  "MaxParallelLaunches": 6,
  "OwnerId": "0123456789123",
  "State": "enabling",
  "StateTransitionReason": "Client.UserInitiated",
  "StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
}
```

Untuk informasi selengkapnya tentang mengonfigurasi AMI Windows untuk peluncuran yang lebih cepat, lihat [Mengonfigurasi AMI Anda untuk peluncuran lebih cepat](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [EnableFastLaunch](#) di Referensi AWS CLI Perintah.

## enable-fast-snapshot-restores

Contoh kode berikut menunjukkan cara menggunakan `enable-fast-snapshot-restores`.

### AWS CLI

Untuk mengaktifkan pemulihan snapshot cepat

`enable-fast-snapshot-restores` Contoh berikut memungkinkan pemulihan snapshot cepat untuk snapshot yang ditentukan di Availability Zones yang ditentukan.

```
aws ec2 enable-fast-snapshot-restores \
  --availability-zones us-east-2a us-east-2b \
  --source-snapshot-ids snap-1234567890abcdef0
```

Output:

```
{
  "Successful": [
    {
      "SnapshotId": "snap-1234567890abcdef0"
      "AvailabilityZone": "us-east-2a",
      "State": "enabling",
      "StateTransitionReason": "Client.UserInitiated",
      "OwnerId": "123456789012",
      "EnablingTime": "2020-01-25T23:57:49.602Z"
    },
    {
      "SnapshotId": "snap-1234567890abcdef0"
      "AvailabilityZone": "us-east-2b",
      "State": "enabling",
      "StateTransitionReason": "Client.UserInitiated",
      "OwnerId": "123456789012",
      "EnablingTime": "2020-01-25T23:57:49.596Z"
    }
  ],
  "Unsuccessful": []
}
```

- Untuk detail API, lihat [EnableFastSnapshotRestores](#) di Referensi AWS CLI Perintah.



## **enable-image-block-public-access**

Contoh kode berikut menunjukkan cara menggunakan `enable-image-block-public-access`.

### AWS CLI

Untuk mengaktifkan blokir akses publik untuk AMI di Wilayah yang ditentukan

`enable-image-block-public-access` Contoh berikut memungkinkan blokir akses publik untuk AMI di tingkat akun di Wilayah yang ditentukan.

```
aws ec2 enable-image-block-public-access \
  --region us-east-1 \
  --image-block-public-access-state block-new-sharing
```

Output:

```
{
  "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Untuk informasi selengkapnya, lihat [Memblokir akses publik ke AMI Anda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [EnableImageBlockPublicAccess](#) di Referensi AWS CLI Perintah.

## **enable-image-deprecation**

Contoh kode berikut menunjukkan cara menggunakan `enable-image-deprecation`.

### AWS CLI

Contoh 1: Untuk menghentikan AMI

`enable-image-deprecation` Contoh berikut menghentikan AMI pada tanggal dan waktu tertentu. Jika Anda menentukan nilai untuk detik, Amazon EC2 membulatkan detik ke menit terdekat. Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

```
aws ec2 enable-image-deprecation \
  --image-id ami-1234567890abcdef0 \
  --deprecate-at "2022-10-15T13:17:12.000Z"
```

**Output:**

```
{
  "RequestID": "59dbff89-35bd-4eac-99ed-be587EXAMPLE",
  "Return": "true"
}
```

Untuk informasi selengkapnya, lihat Menghentikan AMI < <https://docs.aws.amazon.com/AWS-EC2/latest/UserGuide/ami-deprecate.html> > di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [EnableImageDeprecation](#) di Referensi AWS CLI Perintah.

**enable-image**

Contoh kode berikut menunjukkan cara menggunakan `enable-image`.

**AWS CLI**

Untuk mengaktifkan AMI

`enable-image` Contoh berikut memungkinkan AMI yang ditentukan.

```
aws ec2 enable-image \
  --image-id ami-1234567890abcdef0
```

**Output:**

```
{
  "Return": "true"
}
```

Untuk informasi selengkapnya, lihat [Menonaktifkan AMI](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [EnableImage](#) di Referensi AWS CLI Perintah.

**enable-ipam-organization-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `enable-ipam-organization-admin-account`.

## AWS CLI

Untuk mengintegrasikan dengan AWS Organizations dan mendelegasikan akun anggota sebagai akun IPAM

`enable-ipam-organization-admin-account` Contoh berikut mengintegrasikan IPAM dengan AWS Organizations dan mendelegasikan akun anggota sebagai akun IPAM.

```
aws ec2 enable-ipam-organization-admin-account \  
  --delegated-admin-account-id 320805250157
```

Output:

```
{  
  "Success": true  
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan AWS Organizations](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [EnableIpamOrganizationAdminAccount](#) di Referensi AWS CLI Perintah.

## **enable-reachability-analyzer-organization-sharing**

Contoh kode berikut menunjukkan cara menggunakan `enable-reachability-analyzer-organization-sharing`.

### AWS CLI

Untuk mengaktifkan akses tepercaya untuk Reachability Analyzer

`enable-reachability-analyzer-organization-sharing` Contoh berikut memungkinkan akses tepercaya untuk Reachability Analyzer.

```
aws ec2 enable-reachability-analyzer-organization-sharing
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Analisis lintas akun](#) di Panduan Pengguna Reachability Analyzer.

- Untuk detail API, lihat [EnableReachabilityAnalyzerOrganizationSharing](#) di Referensi AWS CLI Perintah.

## **enable-serial-console-access**

Contoh kode berikut menunjukkan cara menggunakan `enable-serial-console-access`.

### AWS CLI

Untuk mengaktifkan akses ke konsol serial untuk akun Anda

`enable-serial-console-access` Contoh berikut memungkinkan akses akun ke konsol serial.

```
aws ec2 enable-serial-console-access
```

Output:

```
{
  "SerialConsoleAccessEnabled": true
}
```

Untuk informasi selengkapnya, lihat [Konsol Serial EC2](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [EnableSerialConsoleAccess](#) di Referensi AWS CLI Perintah.

## **enable-transit-gateway-route-table-propagation**

Contoh kode berikut menunjukkan cara menggunakan `enable-transit-gateway-route-table-propagation`.

### AWS CLI

Untuk mengaktifkan lampiran gateway transit untuk menyebarkan rute ke tabel rute propagasi yang ditentukan

`enable-transit-gateway-route-table-propagation` Contoh berikut memungkinkan lampiran yang ditentukan untuk menyebarkan rute ke tabel rute propagasi yang ditentukan.

```
aws ec2 enable-transit-gateway-route-table-propagation \
  --transit-gateway-route-table-id tgw-rtb-0a823eddbdEXAMPLE \
  --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE
```

**Output:**

```
{
  "Propagation": {
    "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
    "ResourceId": "vpc-4d7de228",
    "ResourceType": "vpc",
    "TransitGatewayRouteTableId": "tgw-rtb-0a823edbbeEXAMPLE",
    "State": "disabled"
  }
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [EnableTransitGatewayRouteTablePropagation](#) di Referensi AWS CLI Perintah.

**enable-vgw-route-propagation**

Contoh kode berikut menunjukkan cara menggunakan `enable-vgw-route-propagation`.

**AWS CLI**

Untuk mengaktifkan propagasi rute

Contoh ini memungkinkan gateway pribadi virtual yang ditentukan untuk menyebarkan rute statis ke tabel rute yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 enable-vgw-route-propagation --route-table-id rtb-22574640 --gateway-id
vgw-9a4cacf3
```

- Untuk detail API, lihat [EnableVgwRoutePropagation](#) di Referensi AWS CLI Perintah.

**enable-volume-io**

Contoh kode berikut menunjukkan cara menggunakan `enable-volume-io`.

**AWS CLI**

Untuk mengaktifkan I/O untuk volume

Contoh ini memungkinkan I/O pada volumevol-1234567890abcdef0.

Perintah:

```
aws ec2 enable-volume-io --volume-id vol-1234567890abcdef0
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [EnableVolumeIo](#) di Referensi AWS CLI Perintah.

### **enable-vpc-classic-link-dns-support**

Contoh kode berikut menunjukkan cara menggunakanenable-vpc-classic-link-dns-support.

AWS CLI

Untuk mengaktifkan dukungan ClassicLink DNS untuk VPC

Contoh ini memungkinkan dukungan ClassicLink DNS untukvpc-88888888.

Perintah:

```
aws ec2 enable-vpc-classic-link-dns-support --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [EnableVpcClassicLinkDnsSupport](#) di Referensi AWS CLI Perintah.

### **enable-vpc-classic-link**

Contoh kode berikut menunjukkan cara menggunakanenable-vpc-classic-link.

## AWS CLI

Untuk mengaktifkan VPC untuk ClassicLink

Contoh ini memungkinkan vpc-8888888 untuk ClassicLink

Perintah:

```
aws ec2 enable-vpc-classic-link --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [EnableVpcClassicLink](#) di Referensi AWS CLI Perintah.

## export-client-vpn-client-certificate-revocation-list

Contoh kode berikut menunjukkan cara menggunakan `export-client-vpn-client-certificate-revocation-list`.

### AWS CLI

Untuk mengekspor daftar pencabutan sertifikat klien

`export-client-vpn-client-certificate-revocation-list` Contoh berikut mengekspor daftar pencabutan sertifikat klien untuk titik akhir Client VPN yang ditentukan. Dalam contoh ini, output dikembalikan dalam format teks agar lebih mudah dibaca.

```
aws ec2 export-client-vpn-client-certificate-revocation-list \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --output text
```

Output:

```
-----BEGIN X509 CRL-----
MIICiTCCAFICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAGGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBASTC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
```

```
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCMVVMxCzAJBgNVBAGTA1dBMRwDgYD
VQHQEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END X509 CRL-----
STATUS      pending
```

Untuk informasi selengkapnya, lihat [Daftar Pencabutan Sertifikat Klien di Panduan Administrator AWS Client VPN](#).

- Untuk detail API, lihat [ExportClientVpnClientCertificateRevocationList](#) di Referensi AWS CLI Perintah.

## export-client-vpn-client-configuration

Contoh kode berikut menunjukkan cara menggunakan `export-client-vpn-client-configuration`.

### AWS CLI

Untuk mengekspor konfigurasi klien

`export-client-vpn-client-configuration` Contoh berikut mengekspor konfigurasi klien untuk titik akhir Client VPN yang ditentukan. Dalam contoh ini, output dikembalikan dalam format teks agar lebih mudah dibaca.

```
aws ec2 export-client-vpn-client-configuration \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --output text
```

### Output:

```
client
dev tun
proto udp
remote cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-south-1.amazonaws.com 443
```



```

remote-random-hostname
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
cipher AES-256-GCM
verb 3
<ca>
-----BEGIN CERTIFICATE-----
MIICiTCcAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBASTC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBASTC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGFT
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----
</ca>
reneg-sec 0

```

Untuk informasi selengkapnya, lihat [Titik Akhir Client VPN](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [ExportClientVpnClientConfiguration](#) di Referensi AWS CLI Perintah.

## export-image

Contoh kode berikut menunjukkan cara menggunakan `export-image`.

### AWS CLI

Untuk mengekspor VM dari AMI

`export-image` Contoh berikut mengekspor AMI yang ditentukan ke bucket yang ditentukan dalam format yang ditentukan.

```
aws ec2 export-image \  
  --image-id ami-1234567890abcdef0 \  
  --disk-image-format VMDK \  
  --s3-export-location S3Bucket=my-export-bucket,S3Prefix=exports/
```

Output:

```
{  
  "DiskImageFormat": "vmdk",  
  "ExportImageTaskId": "export-ami-1234567890abcdef0"  
  "ImageId": "ami-1234567890abcdef0",  
  "RoleName": "vmimport",  
  "Progress": "0",  
  "S3ExportLocation": {  
    "S3Bucket": "my-export-bucket",  
    "S3Prefix": "exports/"  
  },  
  "Status": "active",  
  "StatusMessage": "validating"  
}
```

- Untuk detail API, lihat [ExportImage](#) di Referensi AWS CLI Perintah.

## get-associated-ipv6-pool-cidrs

Contoh kode berikut menunjukkan cara menggunakan `get-associated-ipv6-pool-cidrs`.

AWS CLI

Untuk mendapatkan asosiasi untuk kumpulan alamat IPv6

`get-associated-ipv6-pool-cidrs` Contoh berikut mendapatkan asosiasi untuk kumpulan alamat IPv6 yang ditentukan.

```
aws ec2 get-associated-ipv6-pool-cidrs \  
  --pool-id ipv6pool-ec2-012345abc12345abc
```

Output:

```
{  
  "Ipv6CidrAssociations": [  

```

```

    {
      "Ipv6Cidr": "2001:db8:1234:1a00::/56",
      "AssociatedResource": "vpc-111111222222333ab"
    }
  ]
}

```

- Untuk detail API, lihat [GetAssociatedIpv6 PoolCidrs](#) di Referensi AWS CLI Perintah.

## get-aws-network-performance-data

Contoh kode berikut menunjukkan cara menggunakan `get-aws-network-performance-data`.

### AWS CLI

Untuk mendapatkan data kinerja jaringan

`get-aws-network-performance-data` Contoh berikut mengambil data tentang kinerja jaringan antara Wilayah tertentu dalam periode waktu yang ditentukan.

```

aws ec2 get-aws-network-performance-data \
  --start-time 2022-10-26T12:00:00.000Z \
  --end-time 2022-10-26T12:30:00.000Z \
  --data-queries Id=my-query,Source=us-east-1,Destination=eu-
west-1,Metric=aggregate-latency,Statistic=p50,Period=five-minutes

```

### Output:

```

{
  "DataResponses": [
    {
      "Id": "my-query",
      "Source": "us-east-1",
      "Destination": "eu-west-1",
      "Metric": "aggregate-latency",
      "Statistic": "p50",
      "Period": "five-minutes",
      "MetricPoints": [
        {
          "StartDate": "2022-10-26T12:00:00+00:00",
          "EndDate": "2022-10-26T12:05:00+00:00",
          "Value": 62.44349,

```

```
    "Status": "OK"
  },
  {
    "StartDate": "2022-10-26T12:05:00+00:00",
    "EndDate": "2022-10-26T12:10:00+00:00",
    "Value": 62.483498,
    "Status": "OK"
  },
  {
    "StartDate": "2022-10-26T12:10:00+00:00",
    "EndDate": "2022-10-26T12:15:00+00:00",
    "Value": 62.51248,
    "Status": "OK"
  },
  {
    "StartDate": "2022-10-26T12:15:00+00:00",
    "EndDate": "2022-10-26T12:20:00+00:00",
    "Value": 62.635475,
    "Status": "OK"
  },
  {
    "StartDate": "2022-10-26T12:20:00+00:00",
    "EndDate": "2022-10-26T12:25:00+00:00",
    "Value": 62.733974,
    "Status": "OK"
  },
  {
    "StartDate": "2022-10-26T12:25:00+00:00",
    "EndDate": "2022-10-26T12:30:00+00:00",
    "Value": 62.773975,
    "Status": "OK"
  },
  {
    "StartDate": "2022-10-26T12:30:00+00:00",
    "EndDate": "2022-10-26T12:35:00+00:00",
    "Value": 62.75349,
    "Status": "OK"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Memantau kinerja jaringan](#) di Panduan Pengguna Kinerja Infrastruktur.

- Untuk detail API, lihat [GetAwsNetworkPerformanceData](#) di Referensi AWS CLI Perintah.

## get-capacity-reservation-usage

Contoh kode berikut menunjukkan cara menggunakan `get-capacity-reservation-usage`.

### AWS CLI

Untuk melihat penggunaan reservasi kapasitas di seluruh AWS akun

`get-capacity-reservation-usage` Contoh berikut menampilkan informasi penggunaan untuk reservasi kapasitas yang ditentukan.

```
aws ec2 get-capacity-reservation-usage \
  --capacity-reservation-id cr-1234abcd56EXAMPLE
```

Output:

```
{
  "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
  "InstanceUsages": [
    {
      "UsedInstanceCount": 1,
      "AccountId": "123456789012"
    }
  ],
  "AvailableInstanceCount": 4,
  "TotalInstanceCount": 5,
  "State": "active",
  "InstanceType": "t2.medium"
}
```

Untuk informasi selengkapnya, lihat [Melihat Penggunaan Reservasi Kapasitas Bersama](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [GetCapacityReservationUsage](#) di Referensi AWS CLI Perintah.

## get-coip-pool-usage

Contoh kode berikut menunjukkan cara menggunakan `get-coip-pool-usage`.

### AWS CLI

Untuk mendapatkan penggunaan kumpulan alamat IP milik pelanggan

`get-coip-pool-usage` Contoh berikut mendapatkan rincian penggunaan untuk kumpulan alamat IP milik pelanggan yang ditentukan.

```
aws ec2 get-coip-pool-usage \
  --pool-id ipv4pool-coip-123a45678bEXAMPLE
```

### Output:

```
{
  "CoipPoolId": "ipv4pool-coip-123a45678bEXAMPLE",
  "CoipAddressUsages": [
    {
      "CoIp": "0.0.0.0"
    },
    {
      "AllocationId": "eipalloc-123ab45c6dEXAMPLE",
      "AwsAccountId": "123456789012",
      "CoIp": "0.0.0.0"
    },
    {
      "AllocationId": "eipalloc-123ab45c6dEXAMPLE",
      "AwsAccountId": "123456789111",
      "CoIp": "0.0.0.0"
    }
  ],
  "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Alamat IP milik pelanggan](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [GetCoipPoolUsage](#) di Referensi AWS CLI Perintah.

## get-console-output

Contoh kode berikut menunjukkan cara menggunakan `get-console-output`.

### AWS CLI

Contoh 1: Untuk mendapatkan output konsol

`get-console-output` Contoh berikut mendapatkan output konsol untuk instance Linux tertentu.

```
aws ec2 get-console-output \  
  --instance-id i-1234567890abcdef0
```

Output:

```
{  
  "InstanceId": "i-1234567890abcdef0",  
  "Timestamp": "2013-07-25T21:23:53.000Z",  
  "Output": "..."  
}
```

Untuk informasi selengkapnya, lihat [Output konsol instans](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk mendapatkan keluaran konsol terbaru

`get-console-output` Contoh berikut mendapatkan output konsol terbaru untuk instance Linux yang ditentukan.

```
aws ec2 get-console-output \  
  --instance-id i-1234567890abcdef0 \  
  --latest \  
  --output text
```

Output:

```
i-1234567890abcdef0 [ 0.000000] Command line: root=LABEL=/ console=tty1  
console=ttyS0 selinux=0 nvme_core.io_timeout=4294967295  
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point  
registers'  
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'  
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'  
...
```

```
Cloud-init v. 0.7.6 finished at Wed, 09 May 2018 19:01:13 +0000. DataSource
DataSourceEc2. Up 21.50 seconds
Amazon Linux AMI release 2018.03
Kernel 4.14.26-46.32.amzn1.x
```

Untuk informasi selengkapnya, lihat [Output konsol instans](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [GetConsoleOutput](#) di Referensi AWS CLI Perintah.

## get-console-screenshot

Contoh kode berikut menunjukkan cara menggunakan `get-console-screenshot`.

### AWS CLI

Untuk mengambil tangkapan layar dari instance yang sedang berjalan

`get-console-screenshot` Contoh berikut mengambil screenshot dari contoh yang ditentukan dalam format `.jpg`. Tangkapan layar dikembalikan sebagai string yang dikodekan Base64.

```
aws ec2 get-console-screenshot \
  --instance-id i-1234567890abcdef0
```

Output:

```
{
  "ImageData": "997987/8kgj49ikjhewkww0008084EXAMPLE",
  "InstanceId": "i-1234567890abcdef0"
}
```

- Untuk detail API, lihat [GetConsoleScreenshot](#) di Referensi AWS CLI Perintah.

## get-default-credit-specification

Contoh kode berikut menunjukkan cara menggunakan `get-default-credit-specification`.

### AWS CLI

Untuk menjelaskan opsi kredit default

`get-default-credit-specification` Contoh berikut menjelaskan opsi kredit default untuk instance T2.



```
aws ec2 get-default-credit-specification \
  --instance-family t2
```

Output:

```
{
  "InstanceFamilyCreditSpecification": {
    "InstanceFamily": "t2",
    "CpuCredits": "standard"
  }
}
```

- Untuk detail API, lihat [GetDefaultCreditSpecification](#) di Referensi AWS CLI Perintah.

### get-ebs-default-kms-key-id

Contoh kode berikut menunjukkan cara menggunakan `get-ebs-default-kms-key-id`.

#### AWS CLI

Untuk mendeskripsikan CMK default Anda untuk enkripsi EBS

`get-ebs-default-kms-key-id` Contoh berikut menjelaskan CMK default untuk enkripsi EBS untuk akun Anda AWS .

```
aws ec2 get-ebs-default-kms-key-id
```

Output menunjukkan CMK default untuk enkripsi EBS, yang merupakan CMK AWS terkelola dengan alias. `alias/aws/ebs`

```
{
  "KmsKeyId": "alias/aws/ebs"
}
```

Output berikut menunjukkan CMK kustom untuk enkripsi EBS.

```
{
  "KmsKeyId": "arn:aws:kms:us-
west-2:123456789012:key/0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE"
}
```

- Untuk detail API, lihat [GetEbsDefaultKmsKeyId](#) di Referensi AWS CLI Perintah.

## **get-ebs-encryption-by-default**

Contoh kode berikut menunjukkan cara menggunakan `get-ebs-encryption-by-default`.

### AWS CLI

Untuk menjelaskan apakah enkripsi EBS secara default diaktifkan

`get-ebs-encryption-by-default` Contoh berikut menunjukkan apakah enkripsi EBS secara default diaktifkan untuk AWS akun Anda di Wilayah saat ini.

```
aws ec2 get-ebs-encryption-by-default
```

Output berikut menunjukkan bahwa enkripsi EBS secara default dinonaktifkan.

```
{
  "EbsEncryptionByDefault": false
}
```

Output berikut menunjukkan bahwa enkripsi EBS secara default diaktifkan.

```
{
  "EbsEncryptionByDefault": true
}
```

- Untuk detail API, lihat [GetEbsEncryptionByDefault](#) di Referensi AWS CLI Perintah.

## **get-flow-logs-integration-template**

Contoh kode berikut menunjukkan cara menggunakan `get-flow-logs-integration-template`.

### AWS CLI

Untuk membuat CloudFormation template untuk mengotomatiskan integrasi log aliran VPC dengan Amazon Athena

`get-flow-logs-integration-template` Contoh berikut membuat CloudFormation template untuk mengotomatiskan integrasi log aliran VPC dengan Amazon Athena.

**Linux:**

```
aws ec2 get-flow-logs-integration-template \
  --flow-log-id fl-1234567890abcdef0 \
  --config-delivery-s3-destination-arn arn:aws:s3::DOC-EXAMPLE-BUCKET \
  --integrate-services
  AthenaIntegrations='[{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
}{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
```

**Windows:**

```
aws ec2 get-flow-logs-integration-template ^
  --flow-log-id fl-1234567890abcdef0 ^
  --config-delivery-s3-destination-arn arn:aws:s3::DOC-EXAMPLE-BUCKET ^
  --integrate-services
  AthenaIntegrations=[{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
}{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
```

**Output:**

```
{
  "Result": "https://DOC-EXAMPLE-BUCKET.s3.us-east-2.amazonaws.com/
VPCFlowLogsIntegrationTemplate_fl-1234567890abcdef0_Wed%20Jul
%2021%2000%3A57%3A56%20UTC%202021.yml"
}
```

Untuk informasi tentang penggunaan CloudFormation templat, lihat [Bekerja dengan AWS CloudFormation templat](#) di Panduan AWS CloudFormation Pengguna.

Untuk informasi tentang penggunaan Amazon Athena dan log aliran, lihat [Log alur kueri menggunakan Amazon Athena di Panduan Pengguna Amazon Virtual Private Cloud](#).

- Untuk detail API, lihat [GetFlowLogsIntegrationTemplatedi](#) Referensi AWS CLI Perintah.

**get-groups-for-capacity-reservation**

Contoh kode berikut menunjukkan cara menggunakan `get-groups-for-capacity-reservation`.

## AWS CLI

Untuk mencantumkan grup sumber daya dengan Reservasi Kapasitas

`get-groups-for-capacity-reservation` Contoh berikut mencantumkan grup sumber daya yang ditambahkan Reservasi Kapasitas yang ditentukan.

```
aws ec2 get-groups-for-capacity-reservation \  
  --capacity-reservation-id cr-1234abcd56EXAMPLE
```

Output:

```
{  
  "CapacityReservationsGroup": [  
    {  
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/my-  
resource-group",  
      "OwnerId": "123456789012"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Reservasi Kapasitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [GetGroupsForCapacityReservation](#) di Referensi AWS CLI Perintah.

## **get-host-reservation-purchase-preview**

Contoh kode berikut menunjukkan cara menggunakan `get-host-reservation-purchase-preview`.

### AWS CLI

Untuk mendapatkan pratinjau pembelian untuk Reservasi Tuan Rumah Khusus

Contoh ini memberikan pratinjau biaya untuk Reservasi Host Khusus yang ditentukan untuk Host Khusus yang ditentukan di akun Anda.

Perintah:

```
aws ec2 get-host-reservation-purchase-preview --offering-id hro-03f707bf363b6b324 --
host-id-set h-013abcd2a00cbd123
```

Output:

```
{
  "TotalHourlyPrice": "1.499",
  "Purchase": [
    {
      "HourlyPrice": "1.499",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "HostIdSet": [
        "h-013abcd2a00cbd123"
      ],
      "UpfrontPrice": "0.000",
      "Duration": 31536000
    }
  ],
  "TotalUpfrontPrice": "0.000"
}
```

- Untuk detail API, lihat [GetHostReservationPurchasePreview](#) di Referensi AWS CLI Perintah.

## get-image-block-public-access-state

Contoh kode berikut menunjukkan cara menggunakan `get-image-block-public-access-state`.

### AWS CLI

Untuk mendapatkan status akses publik blok untuk AMI di Wilayah yang ditentukan

`get-image-block-public-access-state` Contoh berikut mendapatkan status blokir akses publik untuk AMI di tingkat akun di Wilayah tertentu.

```
aws ec2 get-image-block-public-access-state \
  --region us-east-1
```

Output:

```
{
  "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Untuk informasi selengkapnya, lihat [Memblokir akses publik ke AMI Anda](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [GetImageBlockPublicAccessState](#) di Referensi AWS CLI Perintah.

## get-instance-types-from-instance-requirements

Contoh kode berikut menunjukkan cara menggunakan `get-instance-types-from-instance-requirements`.

### AWS CLI

Untuk melihat pratinjau jenis instance yang cocok dengan atribut tertentu

`get-instance-types-from-instance-requirements` Contoh berikut pertama menghasilkan daftar semua atribut yang mungkin yang dapat ditentukan menggunakan `--generate-cli-skeleton` parameter, dan menyimpan daftar ke file JSON. Kemudian, file JSON digunakan untuk menyesuaikan atribut untuk melihat pratinjau jenis instance yang cocok.

Untuk menghasilkan semua atribut yang mungkin dan menyimpan output langsung ke file JSON, gunakan perintah berikut.

```
aws ec2 get-instance-types-from-instance-requirements \
  --region us-east-1 \
  --generate-cli-skeleton input > attributes.json
```

Output:

```
{
  "DryRun": true,
  "ArchitectureTypes": [
    "x86_64_mac"
  ],
  "VirtualizationTypes": [
    "paravirtual"
  ],
  "InstanceRequirements": {
    "VCpuCount": {
```

```
    "Min": 0,
    "Max": 0
  },
  "MemoryMiB": {
    "Min": 0,
    "Max": 0
  },
  "CpuManufacturers": [
    "intel"
  ],
  "MemoryGiBPerVCpu": {
    "Min": 0.0,
    "Max": 0.0
  },
  "ExcludedInstanceTypes": [
    ""
  ],
  "InstanceGenerations": [
    "current"
  ],
  "SpotMaxPricePercentageOverLowestPrice": 0,
  "OnDemandMaxPricePercentageOverLowestPrice": 0,
  "BareMetal": "included",
  "BurstablePerformance": "excluded",
  "RequireHibernateSupport": true,
  "NetworkInterfaceCount": {
    "Min": 0,
    "Max": 0
  },
  "LocalStorage": "required",
  "LocalStorageTypes": [
    "hdd"
  ],
  "TotalLocalStorageGB": {
    "Min": 0.0,
    "Max": 0.0
  },
  "BaselineEbsBandwidthMbps": {
    "Min": 0,
    "Max": 0
  },
  "AcceleratorTypes": [
    "inference"
  ],
],
```

```

    "AcceleratorCount": {
      "Min": 0,
      "Max": 0
    },
    "AcceleratorManufacturers": [
      "xilinx"
    ],
    "AcceleratorNames": [
      "t4"
    ],
    "AcceleratorTotalMemoryMiB": {
      "Min": 0,
      "Max": 0
    }
  },
  "MaxResults": 0,
  "NextToken": ""
}

```

Konfigurasi file JSON. Anda harus memberikan nilai untuk `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount`, dan `MemoryMiB`. Anda dapat menghilangkan atribut lainnya. Ketika dihilangkan, nilai default digunakan. Untuk deskripsi setiap atribut dan nilai defaultnya, lihat `get-instance-types-from-instance-requirements` < <https://docs.aws.amazon.com/cli/latest/reference/ec2/get-instance-types-from-instance-requirements.html> >.

Pratinjau jenis instance yang memiliki atribut yang ditentukan dalam `attributes.json`. Tentukan nama dan jalur ke file JSON Anda dengan menggunakan `--cli-input-json` parameter. Dalam permintaan berikut, output diformat sebagai tabel.

```

aws ec2 get-instance-types-from-instance-requirements \
  --cli-input-json file://attributes.json \
  --output table

```

Isi `attributes.json` file:

```

{
  "ArchitectureTypes": [
    "x86_64"
  ],
  "VirtualizationTypes": [
    "hvm"
  ]
}

```



```

    ],
    "InstanceRequirements": {
      "VCpuCount": {
        "Min": 4,
        "Max": 6
      },
      "MemoryMiB": {
        "Min": 2048
      },
      "InstanceGenerations": [
        "current"
      ]
    }
  }
}

```

Output:

```

-----
|GetInstanceTypesFromInstanceRequirements|
+-----+
||           InstanceTypes           ||
|+-----+|
||           InstanceType           ||
|+-----+|
||  c4.xlarge                        ||
||  c5.xlarge                        ||
||  c5a.xlarge                       ||
||  c5ad.xlarge                      ||
||  c5d.xlarge                       ||
||  c5n.xlarge                       ||
||  d2.xlarge                        ||
...

```

Untuk informasi selengkapnya tentang pemilihan jenis instans berbasis atribut, lihat [Cara kerja pemilihan jenis instans berbasis atribut di Panduan Pengguna](#) Amazon EC2.

- Untuk detail API, lihat [GetInstanceTypesFromInstanceRequirements](#) di Referensi AWS CLI Perintah.

## get-instance-uefi-data

Contoh kode berikut menunjukkan cara menggunakan `get-instance-uefi-data`.

## AWS CLI

Untuk mengambil data UEFI dari sebuah instance

`get-instance-uefi-data` Contoh berikut mengambil data UEFI dari sebuah instance. Jika output kosong, instance tidak berisi data UEFI.

```
aws ec2 get-instance-uefi-data \  
  --instance-id i-0123456789example
```

Output:

```
{  
  "InstanceId": "i-0123456789example",  
  "UefiData": "QU1aTlVFRkkf+uLXAAAAHj5a7fZ9+3dBzxXb/.  
  <snipped>  
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAD4L/J/A0Dshho="
```

Untuk informasi selengkapnya, lihat [Boot Aman UEFI](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [GetInstanceUefiData](#) di Referensi AWS CLI Perintah.

## `get-ipam-address-history`

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-address-history`.

### AWS CLI

Untuk mendapatkan sejarah CIDR

`get-ipam-address-history` Contoh berikut mendapatkan sejarah CIDR.

(Linux):

```
aws ec2 get-ipam-address-history \  
  --cidr 10.0.0.0/16 \  
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \  
  --start-time 2021-12-08T01:00:00.000Z \  
  --end-time 2021-12-10T01:00:00.000Z
```

(Windows):

```
aws ec2 get-ipam-address-history ^
--cidr 10.0.0.0/16 ^
--ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
--start-time 2021-12-08T01:00:00.000Z ^
--end-time 2021-12-10T01:00:00.000Z
```

## Output:

```
{
  "HistoryRecords": [
    {
      "ResourceOwnerId": "123456789012",
      "ResourceRegion": "us-west-1",
      "ResourceType": "vpc",
      "ResourceId": "vpc-06cbefa9ee907e1c0",
      "ResourceCidr": "10.0.0.0/16",
      "ResourceName": "Demo",
      "ResourceComplianceStatus": "unmanaged",
      "ResourceOverlapStatus": "overlapping",
      "VpcId": "vpc-06cbefa9ee907e1c0",
      "SampledStartTime": "2021-12-08T19:54:57.675000+00:00"
    },
    {
      "ResourceOwnerId": "123456789012",
      "ResourceRegion": "us-east-2",
      "ResourceType": "vpc",
      "ResourceId": "vpc-042702f474812c9ad",
      "ResourceCidr": "10.0.0.0/16",
      "ResourceName": "test",
      "ResourceComplianceStatus": "unmanaged",
      "ResourceOverlapStatus": "overlapping",
      "VpcId": "vpc-042702f474812c9ad",
      "SampledStartTime": "2021-12-08T19:54:59.019000+00:00"
    },
    {
      "ResourceOwnerId": "123456789012",
      "ResourceRegion": "us-east-2",
      "ResourceType": "vpc",
      "ResourceId": "vpc-042b8a44f64267d67",
      "ResourceCidr": "10.0.0.0/16",
      "ResourceName": "tester",
      "ResourceComplianceStatus": "unmanaged",
      "ResourceOverlapStatus": "overlapping",
```

```

        "VpcId": "vpc-042b8a44f64267d67",
        "SampledStartTime": "2021-12-08T19:54:59.019000+00:00"
    }
]
}

```

Untuk informasi selengkapnya, [lihat Melihat riwayat alamat IP](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [GetIpamAddressHistory](#) di Referensi AWS CLI Perintah.

## get-ipam-discovered-accounts

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-discovered-accounts`.

### AWS CLI

Untuk melihat akun yang ditemukan oleh IPAM

Dalam skenario ini, Anda adalah admin yang didelegasikan IPAM yang ingin melihat AWS akun yang memiliki sumber daya yang ditemukan IPAM.

`--discovery-region` ini adalah Wilayah operasi IPAM tempat Anda ingin melihat status akun yang dipantau. Misalnya, jika Anda memiliki tiga Wilayah operasi IPAM, Anda mungkin ingin membuat permintaan ini tiga kali untuk melihat stempel waktu khusus untuk penemuan di masing-masing Wilayah tertentu.

`get-ipam-discovered-accounts` Contoh berikut mencantumkan AWS akun yang memiliki sumber daya yang ditemukan IPAM.

```

aws ec2 get-ipam-discovered-accounts \
  --ipam-resource-discovery-id ipam-res-disco-0365d2977fc1672fe \
  --discovery-region us-east-1

```

### Output:

```

{
  "IpamDiscoveredAccounts": [
    {
      "AccountId": "149977607591",
      "DiscoveryRegion": "us-east-1",

```

```

        "LastAttemptedDiscoveryTime": "2024-02-09T19:04:31.379000+00:00",
        "LastSuccessfulDiscoveryTime": "2024-02-09T19:04:31.379000+00:00"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan akun di luar organisasi Anda](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [GetIpamDiscoveredAccounts](#) di Referensi AWS CLI Perintah.

## get-ipam-discovered-public-addresses

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-discovered-public-addresses`.

### AWS CLI

Untuk melihat alamat IP publik yang ditemukan

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM dan Anda ingin melihat alamat IP sumber daya yang ditemukan oleh IPAM. Anda bisa mendapatkan ID penemuan sumber daya dengan [describe-ipam-resource-discoveries](#).

`get-ipam-discovered-public-addresses` Contoh berikut menunjukkan alamat IP publik yang ditemukan untuk penemuan sumber daya.

```

aws ec2 get-ipam-discovered-public-addresses \
  --ipam-resource-discovery-id ipam-res-disco-0f4ef577a9f37a162 \
  --address-region us-east-1 \
  --region us-east-1

```

### Output:

```

{
  "IpamDiscoveredPublicAddresses": [
    {
      "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
      "AddressRegion": "us-east-1",
      "Address": "54.208.155.7",
      "AddressOwnerId": "320805250157",
      "AssociationStatus": "associated",
    }
  ]
}

```

```

    "AddressType": "ec2-public-ip",
    "VpcId": "vpc-073b294916198ce49",
    "SubnetId": "subnet-0b6c8a8839e9a4f15",
    "NetworkInterfaceId": "eni-081c446b5284a5e06",
    "NetworkInterfaceDescription": "",
    "InstanceId": "i-07459a6fca5b35823",
    "Tags": {},
    "NetworkBorderGroup": "us-east-1c",
    "SecurityGroups": [
      {
        "GroupName": "launch-wizard-2",
        "GroupId": "sg-0a489dd6a65c244ce"
      }
    ],
    "SampleTime": "2024-04-05T15:13:59.228000+00:00"
  },
  {
    "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
    "AddressRegion": "us-east-1",
    "Address": "44.201.251.218",
    "AddressOwnerId": "470889052923",
    "AssociationStatus": "associated",
    "AddressType": "ec2-public-ip",
    "VpcId": "vpc-6c31a611",
    "SubnetId": "subnet-062f47608b99834b1",
    "NetworkInterfaceId": "eni-024845359c2c3ae9b",
    "NetworkInterfaceDescription": "",
    "InstanceId": "i-04ef786d9c4e03f41",
    "Tags": {},
    "NetworkBorderGroup": "us-east-1a",
    "SecurityGroups": [
      {
        "GroupName": "launch-wizard-32",
        "GroupId": "sg-0ed1a426e96a68374"
      }
    ],
    "SampleTime": "2024-04-05T15:13:59.145000+00:00"
  }
}

```

Untuk informasi selengkapnya, [lihat Melihat wawasan IP publik](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [GetIpamDiscoveredPublicAddresses](#) di Referensi AWS CLI Perintah.

## get-ipam-discovered-resource-cidrs

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-discovered-resource-cidrs`.

### AWS CLI

Untuk melihat CIDR alamat IP yang ditemukan oleh IPAM

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM yang ingin melihat detail yang terkait dengan CIDR alamat IP untuk sumber daya yang ditemukan IPAM.

Untuk melengkapi permintaan ini:

Penemuan sumber daya yang Anda pilih harus dikaitkan dengan IPAM. `--resource-region` Adalah AWS Wilayah tempat sumber daya dibuat.

`get-ipam-discovered-resource-cidrs` Contoh berikut mencantumkan alamat IP untuk sumber daya yang ditemukan IPAM.

```
aws ec2 get-ipam-discovered-resource-cidrs \
  --ipam-resource-discovery-id ipam-res-disco-0365d2977fc1672fe \
  --resource-region us-east-1
```

### Output:

```
{
  {
    "IpamDiscoveredResourceCidrs": [
      {
        "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
        "ResourceRegion": "us-east-1",
        "ResourceId": "vpc-0c974c95ca7ceef4a",
        "ResourceOwnerId": "149977607591",
        "ResourceCidr": "172.31.0.0/16",
        "ResourceType": "vpc",
        "ResourceTags": [],
        "IpUsage": 0.375,
        "VpcId": "vpc-0c974c95ca7ceef4a",
        "SampleTime": "2024-02-09T19:15:16.529000+00:00"
      },
      {
        "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
        "ResourceRegion": "us-east-1",
```

```

    "ResourceId": "subnet-07fe028119082a8c1",
    "ResourceOwnerId": "149977607591",
    "ResourceCidr": "172.31.0.0/20",
    "ResourceType": "subnet",
    "ResourceTags": [],
    "IpUsage": 0.0012,
    "VpcId": "vpc-0c974c95ca7ceef4a",
    "SampleTime": "2024-02-09T19:15:16.529000+00:00"
  },
  {
    "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
    "ResourceRegion": "us-east-1",
    "ResourceId": "subnet-0a96893763984cc4e",
    "ResourceOwnerId": "149977607591",
    "ResourceCidr": "172.31.64.0/20",
    "ResourceType": "subnet",
    "ResourceTags": [],
    "IpUsage": 0.0012,
    "VpcId": "vpc-0c974c95ca7ceef4a",
    "SampleTime": "2024-02-09T19:15:16.529000+00:00"
  }
}

```

Untuk informasi selengkapnya, lihat [Memantau penggunaan CIDR berdasarkan sumber daya](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [GetIpamDiscoveredResourceCidrs](#) di Referensi AWS CLI Perintah.

## get-ipam-pool-allocations

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-pool-allocations`.

### AWS CLI

Untuk mendapatkan CIDR yang dialokasikan dari kolam IPAM

`get-ipam-pool-allocations` Contoh berikut mendapatkan CIDR yang dialokasikan dari kolam IPAM.

(Linux):

```
aws ec2 get-ipam-pool-allocations \
```



```
--ipam-pool-id ipam-pool-0533048da7d823723 \  
--filters Name=ipam-pool-allocation-id,Values=ipam-pool-  
alloc-0e6186d73999e47389266a5d6991e6220
```

(Windows):

```
aws ec2 get-ipam-pool-allocations ^  
--ipam-pool-id ipam-pool-0533048da7d823723 ^  
--filters Name=ipam-pool-allocation-id,Values=ipam-pool-  
alloc-0e6186d73999e47389266a5d6991e6220
```

Output:

```
{  
  "IpamPoolAllocations": [  
    {  
      "Cidr": "10.0.0.0/16",  
      "IpamPoolAllocationId": "ipam-pool-  
alloc-0e6186d73999e47389266a5d6991e6220",  
      "ResourceType": "custom",  
      "ResourceOwner": "123456789012"  
    }  
  ]  
}
```

- Untuk detail API, lihat [GetIpamPoolAllocations](#) di Referensi AWS CLI Perintah.

## get-ipam-pool-cidrs

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-pool-cidrs`.

AWS CLI

Untuk mendapatkan CIDR yang disediakan ke kolam IPAM

`get-ipam-pool-cidrs` Contoh berikut mendapatkan CIDR disediakan ke kolam IPAM.

(Linux):

```
aws ec2 get-ipam-pool-cidrs \  
--ipam-pool-id ipam-pool-0533048da7d823723 \  
--filters Name=ipam-pool-allocation-id,Values=ipam-pool-  
alloc-0e6186d73999e47389266a5d6991e6220
```

```
--filters 'Name=cidr,Values=10.*'
```

(Windows):

```
aws ec2 get-ipam-pool-cidrs ^
  --ipam-pool-id ipam-pool-0533048da7d823723 ^
  --filters Name=cidr,Values=10.*
```

Output:

```
{
  "IpamPoolCidr": {
    "Cidr": "10.0.0.0/24",
    "State": "provisioned"
  }
}
```

- Untuk detail API, lihat [GetIpamPoolCidrs](#) di Referensi AWS CLI Perintah.

## get-ipam-resource-cidrs

Contoh kode berikut menunjukkan cara menggunakan `get-ipam-resource-cidrs`.

### AWS CLI

Untuk mendapatkan CIDR yang dialokasikan ke sumber daya

`get-ipam-resource-cidrs` Contoh berikut mendapatkan CIDR dialokasikan ke sumber daya.

(Linux):

```
aws ec2 get-ipam-resource-cidrs \
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
  --filters Name=management-state,Values=unmanaged
```

(Windows):

```
aws ec2 get-ipam-resource-cidrs ^
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
```

```
--filters Name=management-state,Values=unmanaged
```

Output:

```
{
  "IpamResourceCidrs": [
    {
      "IpamId": "ipam-08440e7a3acde3908",
      "IpamScopeId": "ipam-scope-02fc38cd4c48e7d38",
      "ResourceRegion": "us-east-2",
      "ResourceOwnerId": "123456789012",
      "ResourceId": "vpc-621b8709",
      "ResourceName": "Default AWS VPC",
      "ResourceCidr": "172.33.0.0/16",
      "ResourceType": "vpc",
      "ResourceTags": [
        {
          "Key": "Environment",
          "Value": "Test"
        },
        {
          "Key": "Name",
          "Value": "Default AWS VPC"
        }
      ],
      "IpUsage": 0.0039,
      "ComplianceStatus": "unmanaged",
      "ManagementState": "unmanaged",
      "OverlapStatus": "nonoverlapping",
      "VpcId": "vpc-621b8709"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memantau penggunaan CIDR berdasarkan sumber daya](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [GetIpamResourceCidrs](#) di Referensi AWS CLI Perintah.

## get-launch-template-data

Contoh kode berikut menunjukkan cara menggunakan `get-launch-template-data`.

## AWS CLI

Untuk mendapatkan data instance untuk template peluncuran

Contoh ini mendapatkan data tentang instance yang ditentukan dan menggunakan `--query` opsi untuk mengembalikan konten `LaunchTemplateData`. Anda dapat menggunakan output sebagai basis untuk membuat templat peluncuran baru atau versi templat peluncuran.

Perintah:

```
aws ec2 get-launch-template-data --instance-id i-0123d646e8048babc --query
'LaunchTemplateData'
```

Output:

```
{
  "Monitoring": {},
  "ImageId": "ami-8c1be5f6",
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/xvda",
      "Ebs": {
        "DeleteOnTermination": true
      }
    }
  ],
  "EbsOptimized": false,
  "Placement": {
    "Tenancy": "default",
    "GroupName": "",
    "AvailabilityZone": "us-east-1a"
  },
  "InstanceType": "t2.micro",
  "NetworkInterfaces": [
    {
      "Description": "",
      "NetworkInterfaceId": "eni-35306abc",
      "PrivateIpAddresses": [
        {
          "Primary": true,
          "PrivateIpAddress": "10.0.0.72"
        }
      ]
    }
  ],
}
```

```

        "SubnetId": "subnet-7b16de0c",
        "Groups": [
            "sg-7c227019"
        ],
        "Ipv6Addresses": [
            {
                "Ipv6Address": "2001:db8:1234:1a00::123"
            }
        ],
        "PrivateIpAddress": "10.0.0.72"
    }
]
}

```

- Untuk detail API, lihat [GetLaunchTemplateData](#) di Referensi AWS CLI Perintah.

## get-managed-prefix-list-associations

Contoh kode berikut menunjukkan cara menggunakan `get-managed-prefix-list-associations`.

### AWS CLI

Untuk mendapatkan asosiasi daftar awalan

`get-managed-prefix-list-associations` Contoh berikut mendapatkan sumber daya yang terkait dengan daftar awalan yang ditentukan.

```
aws ec2 get-managed-prefix-list-associations \
  --prefix-list-id pl-0123456abcabc1
```

Output:

```

{
  "PrefixListAssociations": [
    {
      "ResourceId": "sg-0abc123456abc12345",
      "ResourceOwner": "123456789012"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Daftar awalan terkelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [GetManagedPrefixListAssociations](#) di Referensi AWS CLI Perintah.

## **get-managed-prefix-list-entries**

Contoh kode berikut menunjukkan cara menggunakan `get-managed-prefix-list-entries`.

### AWS CLI

Untuk mendapatkan entri untuk daftar awalan

Berikut ini `get-managed-prefix-list-entries` mendapatkan entri untuk daftar awalan yang ditentukan.

```
aws ec2 get-managed-prefix-list-entries \  
  --prefix-list-id pl-0123456abcabc1
```

Output:

```
{  
  "Entries": [  
    {  
      "Cidr": "10.0.0.0/16",  
      "Description": "vpc-a"  
    },  
    {  
      "Cidr": "10.2.0.0/16",  
      "Description": "vpc-b"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar awalan terkelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [GetManagedPrefixListEntries](#) di Referensi AWS CLI Perintah.

## **get-network-insights-access-scope-analysis-findings**

Contoh kode berikut menunjukkan cara menggunakan `get-network-insights-access-scope-analysis-findings`.

## AWS CLI

Untuk mendapatkan temuan analisis cakupan akses Network Insights

`get-network-insights-access-scope-analysis-findings` Contoh berikut mendapatkan temuan analisis lingkup yang dipilih di AWS akun Anda.

```
aws ec2 get-network-insights-access-scope-analysis-findings \
  --region us-east-1 \
  --network-insights-access-scope-analysis-id nis \
  --nis-123456789111
```

Output:

```
{
  "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
  "AnalysisFindings": [
    {
      "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
      "NetworkInsightsAccessScopeId": "nis-123456789111",
      "FindingComponents": [
        {
          "SequenceNumber": 1,
          "Component": {
            "Id": "eni-02e3d42d5cceca67d",
            "Arn": "arn:aws:ec2:us-east-1:936459623503:network-
interface/eni-02e3d32d9cceca17d"
          },
          "OutboundHeader": {
            "DestinationAddresses": [
              "0.0.0.0/5",
              "11.0.0.0/8",
              "12.0.0.0/6",
              "128.0.0.0/3",
              "16.0.0.0/4",
              "160.0.0.0/5",
              "168.0.0.0/6",
              "172.0.0.0/12"
              "8.0.0.0/7"
            ],
            "DestinationPortRanges": [
              {
                "From": 0,
```

```
        "To": 65535
      }
    ],
    "Protocol": "6",
    "SourceAddresses": [
      "10.0.2.253/32"
    ],
    "SourcePortRanges": [
      {
        "From": 0,
        "To": 65535
      }
    ]
  }, [etc]
]
}
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [GetNetworkInsightsAccessScopeAnalysisFindings](#) di Referensi AWS CLI Perintah.

## get-network-insights-access-scope-content

Contoh kode berikut menunjukkan cara menggunakan `get-network-insights-access-scope-content`.

### AWS CLI

Untuk mendapatkan konten cakupan akses Network Insights

`get-network-insights-access-scope-content` Contoh berikut mendapatkan konten ID analisis lingkup yang dipilih di AWS akun Anda.

```
aws ec2 get-network-insights-access-scope-content \
  --region us-east-1 \
  --network-insights-access-scope-id nis-123456789222
```



**Output:**

```
{
  "NetworkInsightsAccessScopeContent": {
    "NetworkInsightsAccessScopeId": "nis-123456789222",
    "MatchPaths": [
      {
        "Source": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::NetworkInterface"
            ]
          }
        },
        "Destination": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::InternetGateway"
            ]
          }
        }
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [GetNetworkInsightsAccessScopeContent](#) di Referensi AWS CLI Perintah.

**get-password-data**

Contoh kode berikut menunjukkan cara menggunakan `get-password-data`.

**AWS CLI**

Untuk mendapatkan kata sandi terenkripsi

Contoh ini mendapatkan kata sandi terenkripsi.

Perintah:

```
aws ec2 get-password-data --instance-id i-1234567890abcdef0
```

Output:

```
{
  "InstanceId": "i-1234567890abcdef0",
  "Timestamp": "2013-08-07T22:18:38.000Z",
  "PasswordData": "gSlJFq+VpcZXqy+iktXMF6NyxQ4qCrT4+ga0uN0enX1MmgXPTj7XEXAMPLE
UQ+YeFfb+L1U4C4AKv652Ux1iRB3CPTYp7WmU3TUnhsuBd+p6LVk7T2lKUm160Xbk6WPW1VYYm/TRPB1
e1DQ7PY4an/DgZT4mwcpRfigzhniQgDDe01InvSDcwoUTwNs0Y1S8ouri2W4n5GNlriM3Q0AnNve1Vz/
53TkDtxbNoU606M1gK9zUWSxqEgwvbV2j8c5rP0WCuaMWSF14ziDu4bd7q+4RSyi8NusVWnKZ4aEZffu
DPGzKrF5yL1f3etP2L4ZR6CvG7K1hx7VK0QVN32Dajw=="
}
```

Untuk mendapatkan kata sandi yang didekripsi

Contoh ini mendapatkan kata sandi yang didekripsi.

Perintah:

```
aws ec2 get-password-data --instance-id i-1234567890abcdef0 --priv-launch-key C:
\Keys\MyKeyPair.pem
```

Output:

```
{
  "InstanceId": "i-1234567890abcdef0",
  "Timestamp": "2013-08-30T23:18:05.000Z",
  "PasswordData": "&ViJ652e*u"
}
```

- Untuk detail API, lihat [GetPasswordData](#) di Referensi AWS CLI Perintah.

## get-reserved-instances-exchange-quote

Contoh kode berikut menunjukkan cara menggunakan `get-reserved-instances-exchange-quote`.

AWS CLI

Untuk mendapatkan penawaran untuk menukar Instans Cadangan Konvertibel

Contoh ini mendapatkan informasi pertukaran untuk Instans Cadangan Konvertibel yang ditentukan.

Perintah:

```
aws ec2 get-reserved-instances-exchange-quote --reserved-instance-ids
7b8750c3-397e-4da4-bbcb-a45ebexample --target-configurations OfferingId=6fea5434-
b379-434c-b07b-a7abexample
```

Output:

```
{
  "CurrencyCode": "USD",
  "ReservedInstanceValueSet": [
    {
      "ReservedInstanceId": "7b8750c3-397e-4da4-bbcb-a45ebexample",
      "ReservationValue": {
        "RemainingUpfrontValue": "0.000000",
        "HourlyPrice": "0.027800",
        "RemainingTotalValue": "730.556200"
      }
    }
  ],
  "PaymentDue": "424.983828",
  "TargetConfigurationValueSet": [
    {
      "TargetConfiguration": {
        "InstanceCount": 5,
        "OfferingId": "6fea5434-b379-434c-b07b-a7abexample"
      },
      "ReservationValue": {
        "RemainingUpfrontValue": "424.983828",
        "HourlyPrice": "0.016000",
        "RemainingTotalValue": "845.447828"
      }
    }
  ],
  "IsValidExchange": true,
  "OutputReservedInstancesWillExpireAt": "2020-10-01T13:03:39Z",
  "ReservedInstanceValueRollup": {
    "RemainingUpfrontValue": "0.000000",
    "HourlyPrice": "0.027800",
    "RemainingTotalValue": "730.556200"
  }
}
```

```
},  
  "TargetConfigurationValueRollup": {  
    "RemainingUpfrontValue": "424.983828",  
    "HourlyPrice": "0.016000",  
    "RemainingTotalValue": "845.447828"  
  }  
}
```

- Untuk detail API, lihat [GetReservedInstancesExchangeQuoted](#) di Referensi AWS CLI Perintah.

## get-security-groups-for-vpc

Contoh kode berikut menunjukkan cara menggunakan `get-security-groups-for-vpc`.

### AWS CLI

Untuk melihat grup keamanan yang dapat dikaitkan dengan antarmuka jaringan dalam VPC tertentu.

`get-security-groups-for-vpc` Contoh berikut menunjukkan grup keamanan yang dapat dikaitkan dengan antarmuka jaringan di VPC.

```
aws ec2 get-security-groups-for-vpc \  
  --vpc-id vpc-6c31a611 \  
  --region us-east-1
```

### Output:

```
{  
  "SecurityGroupForVpcs": [  
    {  
      "Description": "launch-wizard-36 created 2022-08-29T15:59:35.338Z",  
      "GroupName": "launch-wizard-36",  
      "OwnerId": "470889052923",  
      "GroupId": "sg-007e0c3027ee885f5",  
      "Tags": [],  
      "PrimaryVpcId": "vpc-6c31a611"  
    },  
    {  
      "Description": "launch-wizard-18 created 2024-01-19T20:22:27.527Z",  
      "GroupName": "launch-wizard-18",  
      "OwnerId": "470889052923",
```

```
    "GroupId": "sg-0147193bef51c9eef",  
    "Tags": [],  
    "PrimaryVpcId": "vpc-6c31a611"  
  }  
}
```

- Untuk detail API, lihat [GetSecurityGroupsForVpc](#) di Referensi AWS CLI Perintah.

## get-serial-console-access-status

Contoh kode berikut menunjukkan cara menggunakan `get-serial-console-access-status`.

### AWS CLI

Untuk melihat status akses akun ke konsol serial

`get-serial-console-access-status` Contoh berikut menentukan apakah akses konsol serial diaktifkan untuk akun Anda.

```
aws ec2 get-serial-console-access-status
```

Output:

```
{  
  "SerialConsoleAccessEnabled": true  
}
```

Untuk informasi selengkapnya, lihat [Konsol Serial EC2](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [GetSerialConsoleAccessStatus](#) di Referensi AWS CLI Perintah.

## get-spot-placement-scores

Contoh kode berikut menunjukkan cara menggunakan `get-spot-placement-scores`.

### AWS CLI

Untuk menghitung skor penempatan Spot untuk persyaratan yang ditentukan

`get-spot-placement-scores` Contoh berikut pertama menghasilkan daftar semua parameter yang mungkin yang dapat ditentukan untuk konfigurasi skor penempatan Spot menggunakan `--generate-cli-skeleton` parameter, dan menyimpan daftar ke file JSON. Kemudian, file

JSON digunakan untuk mengonfigurasi persyaratan yang akan digunakan untuk menghitung skor penempatan Spot.

Untuk menghasilkan semua parameter yang mungkin yang dapat ditentukan untuk konfigurasi skor penempatan Spot, dan menyimpan output langsung ke file JSON.

```
aws ec2 get-spot-placement-scores \  
  --region us-east-1 \  
  --generate-cli-skeleton input > attributes.json
```

Output:

```
{  
  "InstanceTypes": [  
    ""  
  ],  
  "TargetCapacity": 0,  
  "TargetCapacityUnitType": "vcpu",  
  "SingleAvailabilityZone": true,  
  "RegionNames": [  
    ""  
  ],  
  "InstanceRequirementsWithMetadata": {  
    "ArchitectureTypes": [  
      "x86_64_mac"  
    ],  
    "VirtualizationTypes": [  
      "hvm"  
    ],  
    "InstanceRequirements": {  
      "VCpuCount": {  
        "Min": 0,  
        "Max": 0  
      },  
      "MemoryMiB": {  
        "Min": 0,  
        "Max": 0  
      },  
      "CpuManufacturers": [  
        "amd"  
      ],  
      "MemoryGiBPerVCpu": {  
        "Min": 0.0,
```

```
    "Max": 0.0
  },
  "ExcludedInstanceTypes": [
    ""
  ],
  "InstanceGenerations": [
    "previous"
  ],
  "SpotMaxPricePercentageOverLowestPrice": 0,
  "OnDemandMaxPricePercentageOverLowestPrice": 0,
  "BareMetal": "excluded",
  "BurstablePerformance": "excluded",
  "RequireHibernateSupport": true,
  "NetworkInterfaceCount": {
    "Min": 0,
    "Max": 0
  },
  "LocalStorage": "included",
  "LocalStorageTypes": [
    "hdd"
  ],
  "TotalLocalStorageGB": {
    "Min": 0.0,
    "Max": 0.0
  },
  "BaselineEbsBandwidthMbps": {
    "Min": 0,
    "Max": 0
  },
  "AcceleratorTypes": [
    "fpga"
  ],
  "AcceleratorCount": {
    "Min": 0,
    "Max": 0
  },
  "AcceleratorManufacturers": [
    "amd"
  ],
  "AcceleratorNames": [
    "vu9p"
  ],
  "AcceleratorTotalMemoryMiB": {
    "Min": 0,
```

```

        "Max": 0
      }
    }
  },
  "DryRun": true,
  "MaxResults": 0,
  "NextToken": ""
}

```

Konfigurasi file JSON. Anda harus memberikan nilai untuk `TargetCapacity`. Untuk deskripsi setiap parameter dan nilai defaultnya, lihat Menghitung skor penempatan Spot (AWS CLI) < [Hitung skor penempatan Spot untuk persyaratan yang ditentukan dalam `attributes.json`. Tentukan nama dan jalur ke file JSON Anda dengan menggunakan `--cli-input-json` parameter.](https://docs.aws.amazon.com/AWS UserGuide EC2/latest//.html#>. spot-placement-score calculate-sps-cli</a></p>
</div>
<div data-bbox=)

```

aws ec2 get-spot-placement-scores \
  --region us-east-1 \
  --cli-input-json file://attributes.json

```

Output jika `SingleAvailabilityZone` disetel ke `false` atau dihilangkan (jika dihilangkan, defaultnya). `false` Daftar Wilayah yang dinilai dikembalikan.

```

"Recommendation": [
  {
    "Region": "us-east-1",
    "Score": 7
  },
  {
    "Region": "us-west-1",
    "Score": 5
  },
  ...

```

Output `SingleAvailabilityZone` jika diatur ke `true`. Daftar `SingleAvailability` Zona yang dicetak dikembalikan.

```

"Recommendation": [
  {

```



```

    "Region": "us-east-1",
    "AvailabilityZoneId": "use1-az1"
    "Score": 8
  },
  {
    "Region": "us-east-1",
    "AvailabilityZoneId": "usw2-az3"
    "Score": 6
  },
  ...

```

Untuk informasi selengkapnya tentang menghitung skor penempatan Spot, dan misalnya konfigurasi, lihat [Menghitung skor penempatan Spot](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [GetSpotPlacementScores](#) di Referensi AWS CLI Perintah.

## get-subnet-cidr-reservations

Contoh kode berikut menunjukkan cara menggunakan `get-subnet-cidr-reservations`.

### AWS CLI

Untuk mendapatkan informasi tentang reservasi CIDR subnet

`get-subnet-cidr-reservations` Contoh berikut menampilkan informasi tentang reservasi CIDR subnet yang ditentukan.

```
aws ec2 get-subnet-cidr-reservations \
  --subnet-id subnet-03c51e2e6cEXAMPLE
```

Output:

```

{
  "SubnetIpv4CidrReservations": [
    {
      "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
      "SubnetId": "subnet-03c51e2e6cEXAMPLE",
      "Cidr": "10.1.0.16/28",
      "ReservationType": "prefix",
      "OwnerId": "123456789012"
    }
  ],

```

```
"SubnetIpv6CidrReservations": []
}
```

Untuk informasi selengkapnya, lihat [Reservasi CIDR Subnet](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [GetSubnetCidrReservations](#) di Referensi AWS CLI Perintah.

## get-transit-gateway-attachment-propagations

Contoh kode berikut menunjukkan cara menggunakan `get-transit-gateway-attachment-propagations`.

### AWS CLI

Untuk membuat daftar tabel rute tempat lampiran sumber daya yang ditentukan menyebarkan rute

`get-transit-gateway-attachment-propagations` Contoh berikut mencantumkan tabel rute tempat lampiran sumber daya tertentu menyebarkan rute.

```
aws ec2 get-transit-gateway-attachment-propagations \
  --transit-gateway-attachment-id tgw-attach-09fbd47ddfEXAMPLE
```

Output:

```
{
  "TransitGatewayAttachmentPropagations": [
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0882c61b97EXAMPLE",
      "State": "enabled"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [GetTransitGatewayAttachmentPropagations](#) di Referensi AWS CLI Perintah.

## get-transit-gateway-multicast-domain-associations

Contoh kode berikut menunjukkan cara menggunakan `get-transit-gateway-multicast-domain-associations`.

## AWS CLI

Untuk melihat informasi tentang asosiasi domain multicast gateway transit

`get-transit-gateway-multicast-domain-associations` Contoh berikut mengembalikan asosiasi untuk domain multicast tertentu.

```
aws ec2 get-transit-gateway-multicast-domain-associations \  
--transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{  
  "MulticastDomainAssociations": [  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-028c1dd0f8EXAMPLE",  
      "ResourceId": "vpc-01128d2c24EXAMPLE",  
      "ResourceType": "vpc",  
      "Subnet": {  
        "SubnetId": "subnet-000de86e3bEXAMPLE",  
        "State": "associated"  
      }  
    },  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",  
      "ResourceId": "vpc-7EXAMPLE",  
      "ResourceType": "vpc",  
      "Subnet": {  
        "SubnetId": "subnet-4EXAMPLE",  
        "State": "associated"  
      }  
    },  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",  
      "ResourceId": "vpc-7EXAMPLE",  
      "ResourceType": "vpc",  
      "Subnet": {  
        "SubnetId": "subnet-5EXAMPLE",  
        "State": "associated"  
      }  
    },  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
```

```

    "ResourceId": "vpc-7EXAMPLE",
    "ResourceType": "vpc",
    "Subnet": {
        "SubnetId": "subnet-aEXAMPLE",
        "State": "associated"
    }
},
{
    "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
    "ResourceId": "vpc-7EXAMPLE",
    "ResourceType": "vpc",
    "Subnet": {
        "SubnetId": "subnet-fEXAMPLE",
        "State": "associated"
    }
}
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Transit Gateways.

- Untuk detail API, lihat [GetTransitGatewayMulticastDomainAssociations](#) di Referensi AWS CLI Perintah.

## get-transit-gateway-prefix-list-references

Contoh kode berikut menunjukkan cara menggunakan `get-transit-gateway-prefix-list-references`.

### AWS CLI

Untuk mendapatkan referensi daftar awalan dalam tabel rute gateway transit

`get-transit-gateway-prefix-list-references` Contoh berikut mendapatkan referensi daftar awalan untuk tabel rute gateway transit yang ditentukan, dan filter oleh ID daftar awalan tertentu.

```

aws ec2 get-transit-gateway-prefix-list-references \
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
  --filters Name=prefix-list-id,Values=pl-11111122222222333

```

Output:

```
{
  "TransitGatewayPrefixListReferences": [
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
      "PrefixListId": "pl-11111122222222333",
      "PrefixListOwnerId": "123456789012",
      "State": "available",
      "Blackhole": false,
      "TransitGatewayAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
        "ResourceType": "vpc",
        "ResourceId": "vpc-112233445566aabbcc"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Referensi daftar awalan](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [GetTransitGatewayPrefixListReferences](#) di Referensi AWS CLI Perintah.

## get-transit-gateway-route-table-associations

Contoh kode berikut menunjukkan cara menggunakan `get-transit-gateway-route-table-associations`.

### AWS CLI

Untuk mendapatkan informasi tentang asosiasi untuk tabel rute gateway transit yang ditentukan

`get-transit-gateway-route-table-associations` Contoh berikut menampilkan informasi tentang asosiasi untuk tabel rute gateway transit yang ditentukan.

```
aws ec2 get-transit-gateway-route-table-associations \
  --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE
```

### Output:

```
{
  "Associations": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
```

```

        "ResourceId": "vpc-4d7de228",
        "ResourceType": "vpc",
        "State": "associating"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [GetTransitGatewayRouteTableAssociations](#) di Referensi AWS CLI Perintah.

## get-transit-gateway-route-table-propagations

Contoh kode berikut menunjukkan cara menggunakan `get-transit-gateway-route-table-propagations`.

### AWS CLI

Untuk menampilkan informasi tentang propagasi tabel rute untuk tabel rute gateway transit yang ditentukan

`get-transit-gateway-route-table-propagations` Contoh berikut mengembalikan propagasi tabel rute untuk tabel rute yang ditentukan.

```

aws ec2 get-transit-gateway-route-table-propagations \
  --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE

```

### Output:

```

{
  "TransitGatewayRouteTablePropagations": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-01f8100bc7EXAMPLE",
      "ResourceId": "vpc-3EXAMPLE",
      "ResourceType": "vpc",
      "State": "enabled"
    },
    {
      "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
      "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
      "ResourceType": "direct-connect-gateway",

```

```

        "State": "enabled"
    },
    {
        "TransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
        "ResourceId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
        "ResourceType": "direct-connect-gateway",
        "State": "enabled"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [GetTransitGatewayRouteTablePropagations](#) di Referensi AWS CLI Perintah.

## get-verified-access-endpoint-policy

Contoh kode berikut menunjukkan cara menggunakan `get-verified-access-endpoint-policy`.

### AWS CLI

Untuk mendapatkan kebijakan Akses Terverifikasi dari titik akhir

`get-verified-access-endpoint-policy` Contoh berikut mendapatkan kebijakan Akses Terverifikasi dari titik akhir yang ditentukan.

```
aws ec2 get-verified-access-endpoint-policy \
  --verified-access-endpoint-id vae-066fac616d4d546f2
```

### Output:

```
{
  "PolicyEnabled": true,
  "PolicyDocument": "permit(principal,action,resource)\nwhen
{\n  context.identity.groups.contains(\"finance\") &&\n
context.identity.email_verified == true\n};"
}
```

Untuk informasi selengkapnya, lihat [kebijakan Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [GetVerifiedAccessEndpointPolicy](#) di Referensi AWS CLI Perintah.

## get-verified-access-group-policy

Contoh kode berikut menunjukkan cara menggunakan `get-verified-access-group-policy`.

### AWS CLI

Untuk mendapatkan kebijakan Akses Terverifikasi grup

`get-verified-access-group-policy` Contoh berikut mendapatkan kebijakan Akses Terverifikasi dari grup yang ditentukan.

```
aws ec2 get-verified-access-group-policy \
  --verified-access-group-id vagr-0dbe967baf14b7235
```

Output:

```
{
  "PolicyEnabled": true,
  "PolicyDocument": "permit(principal,action,resource)\nwhen
{\n  context.identity.groups.contains(\"finance\") &&\n
context.identity.email_verified == true\n};"
}
```

Untuk informasi selengkapnya, lihat [Grup Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [GetVerifiedAccessGroupPolicy](#) di Referensi AWS CLI Perintah.

## get-vpn-connection-device-sample-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-vpn-connection-device-sample-configuration`.

### AWS CLI

Untuk mengunduh file konfigurasi sampel

`get-vpn-connection-device-sample-configuration` Contoh berikut mengunduh file konfigurasi sampel yang ditentukan. Untuk membuat daftar perangkat gateway dengan file konfigurasi sampel, panggil `get-vpn-connection-device-types` perintah.



```
aws ec2 get-vpn-connection-device-sample-configuration \
  --vpn-connection-id vpn-123456789abc01234 \
  --vpn-connection-device-type-id 5fb390ba
```

Output:

```
{
  "VpnConnectionDeviceSampleConfiguration": "contents-of-the-sample-configuration-
  file"
}
```

Untuk informasi selengkapnya, lihat [Mengunduh file konfigurasi](#) di Panduan Pengguna AWS VPN Site-to-Site.

- Untuk detail API, lihat [GetVpnConnectionDeviceSampleConfiguration](#) di Referensi AWS CLI Perintah.

## get-vpn-connection-device-types

Contoh kode berikut menunjukkan cara menggunakan `get-vpn-connection-device-types`.

AWS CLI

Untuk membuat daftar perangkat gateway dengan contoh file konfigurasi

`get-vpn-connection-device-types` Contoh berikut mencantumkan perangkat gateway dari Palo Alto Networks yang memiliki file konfigurasi sampel.

```
aws ec2 get-vpn-connection-device-types \
  --query "VpnConnectionDeviceTypes[?Vendor=='Palo Alto Networks']"
```

Output:

```
[
  {
    "VpnConnectionDeviceTypeId": "754a6372",
    "Vendor": "Palo Alto Networks",
    "Platform": "PA Series",
    "Software": "PANOS 4.1.2+"
  },
  {
    "VpnConnectionDeviceTypeId": "9612cbed",
```

```
    "Vendor": "Palo Alto Networks",
    "Platform": "PA Series",
    "Software": "PANOS 4.1.2+ (GUI)"
  },
  {
    "VpnConnectionDeviceTypeId": "5fb390ba",
    "Vendor": "Palo Alto Networks",
    "Platform": "PA Series",
    "Software": "PANOS 7.0+"
  }
]
```

Untuk informasi selengkapnya, lihat [Mengunduh file konfigurasi](#) di Panduan pengguna AWS VPN Site-to-Site.

- Untuk detail API, lihat [GetVpnConnectionDeviceTypes](#) di Referensi AWS CLI Perintah.

## **import-client-vpn-client-certificate-revocation-list**

Contoh kode berikut menunjukkan cara menggunakan `import-client-vpn-client-certificate-revocation-list`.

### AWS CLI

Untuk mengimpor daftar pencabutan sertifikat klien

`import-client-vpn-client-certificate-revocation-list` Contoh berikut mengimpor daftar pencabutan sertifikat klien ke titik akhir Client VPN dengan menentukan lokasi file di komputer lokal.

```
aws ec2 import-client-vpn-client-certificate-revocation-list \
  --certificate-revocation-list file:///path/to/crl.pem \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Daftar Pencabutan Sertifikat Klien di Panduan Administrator AWS Client VPN](#).

- Untuk detail API, lihat [ImportClientVpnClientCertificateRevocationList](#) di Referensi AWS CLI Perintah.

## import-image

Contoh kode berikut menunjukkan cara menggunakan `import-image`.

### AWS CLI

Untuk mengimpor file gambar VM sebagai AMI

`import-image` Contoh berikut mengimpor OVA tertentu.

```
aws ec2 import-image \  
  --disk-containers Format=ova,UserBucket="{S3Bucket=my-import-bucket,S3Key=vms/my-  
server-vm.ova}"
```

Output:

```
{  
  "ImportTaskId": "import-ami-1234567890abcdef0",  
  "Progress": "2",  
  "SnapshotDetails": [  
    {  
      "DiskImageSize": 0.0,  
      "Format": "ova",  
      "UserBucket": {  
        "S3Bucket": "my-import-bucket",  
        "S3Key": "vms/my-server-vm.ova"  
      }  
    }  
  ],  
  "Status": "active",  
  "StatusMessage": "pending"  
}
```

- Untuk detail API, lihat [ImportImage](#) di Referensi AWS CLI Perintah.

## import-key-pair

Contoh kode berikut menunjukkan cara menggunakan `import-key-pair`.

## AWS CLI

Untuk mengimpor kunci publik

Pertama, buat key pair dengan alat pilihan Anda. Misalnya, gunakan perintah `ssh-keygen` ini:

Perintah:

```
ssh-keygen -t rsa -C "my-key" -f ~/.ssh/my-key
```

Output:

```
Generating public/private rsa key pair.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/ec2-user/.ssh/my-key.  
Your public key has been saved in /home/ec2-user/.ssh/my-key.pub.  
...
```

Perintah contoh ini mengimpor kunci publik yang ditentukan.

Perintah:

```
aws ec2 import-key-pair --key-name "my-key" --public-key-material fileb://~/.ssh/my-key.pub
```

Output:

```
{  
  "KeyName": "my-key",  
  "KeyFingerprint": "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca"  
}
```

- Untuk detail API, lihat [ImportKeyPair](#) di Referensi AWS CLI Perintah.

## **import-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `import-snapshot`.

## AWS CLI

Untuk mengimpor snapshot

`import-snapshot` Contoh berikut mengimpor disk yang ditentukan sebagai snapshot.

```
aws ec2 import-snapshot \  
  --description "My server VMDK" \  
  --disk-container Format=VMDK,UserBucket={S3Bucket=my-import-bucket,S3Key=vms/my-  
server-vm.vmdk}
```

Output:

```
{  
  "Description": "My server VMDK",  
  "ImportTaskId": "import-snap-1234567890abcdef0",  
  "SnapshotTaskDetail": {  
    "Description": "My server VMDK",  
    "DiskImageSize": "0.0",  
    "Format": "VMDK",  
    "Progress": "3",  
    "Status": "active",  
    "StatusMessage": "pending"  
    "UserBucket": {  
      "S3Bucket": "my-import-bucket",  
      "S3Key": "vms/my-server-vm.vmdk"  
    }  
  }  
}
```

- Untuk detail API, lihat [ImportSnapshot](#) di Referensi AWS CLI Perintah.

## **list-images-in-recycle-bin**

Contoh kode berikut menunjukkan cara menggunakan `list-images-in-recycle-bin`.

AWS CLI

Untuk membuat daftar gambar di Recycle Bin

`list-images-in-recycle-bin` Contoh berikut mencantumkan semua gambar yang saat ini disimpan di Recycle Bin.

```
aws ec2 list-images-in-recycle-bin
```

Output:

```
{
  "Images": [
    {
      "RecycleBinEnterTime": "2022-03-14T15:35:08.000Z",
      "Description": "Monthly AMI One",
      "RecycleBinExitTime": "2022-03-15T15:35:08.000Z",
      "Name": "AMI_01",
      "ImageId": "ami-0111222333444abcd"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulihkan AMI dari Recycle Bin](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [ListImagesInRecycleBin](#) di Referensi AWS CLI Perintah.

## list-snapshots-in-recycle-bin

Contoh kode berikut menunjukkan cara menggunakan `list-snapshots-in-recycle-bin`.

### AWS CLI

Untuk melihat snapshot di Recycle Bin

`list-snapshots-in-recycle-bin` Contoh berikut mencantumkan informasi tentang snapshot di Recycle Bin, termasuk ID snapshot, deskripsi snapshot, ID volume dari mana snapshot dibuat, tanggal dan waktu ketika snapshot dihapus dan masuk ke Recycle Bin, dan tanggal dan waktu ketika periode retensi berakhir.

```
aws ec2 list-snapshots-in-recycle-bin \
  --snapshot-id snap-01234567890abcdef
```

Output:

```
{
  "SnapshotRecycleBinInfo": [
    {
      "Description": "Monthly data backup snapshot",
      "RecycleBinEnterTime": "2022-12-01T13:00:00.000Z",
      "RecycleBinExitTime": "2022-12-15T13:00:00.000Z",
```

```

        "VolumeId": "vol-abcdef09876543210",
        "SnapshotId": "snap-01234567890abcdef"
    }
]
}

```

Untuk informasi selengkapnya tentang Recycle Bin untuk Amazon EBS, lihat [Memulihkan snapshot dari Recycle Bin di Panduan Pengguna Amazon EC2](#).

- Untuk detail API, lihat [ListSnapshotsInRecycleBin](#) di Referensi AWS CLI Perintah.

## modify-address-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-address-attribute`.

### AWS CLI

Untuk memodifikasi atribut nama domain yang terkait dengan alamat IP elastis

`modify-address-attribute` Contoh berikut memodifikasi atribut nama domain dari alamat IP elastis.

Linux:

```

aws ec2 modify-address-attribute \
  --allocation-id eipalloc-abcdef01234567890 \
  --domain-name example.com

```

Windows:

```

aws ec2 modify-address-attribute ^
  --allocation-id eipalloc-abcdef01234567890 ^
  --domain-name example.com

```

Output:

```

{
  "Addresses": [
    {
      "PublicIp": "192.0.2.0",
      "AllocationId": "eipalloc-abcdef01234567890",
      "PtrRecord": "example.net."
    }
  ]
}

```

```
    "PtrRecordUpdate": {
      "Value": "example.com.",
      "Status": "PENDING"
    }
  ]
}
```

Untuk memantau perubahan yang tertunda dan untuk melihat atribut yang dimodifikasi dari alamat IP elastis, lihat [describe-addresses-attribute](#) di Referensi Perintah AWS CLI.

- Untuk detail API, lihat [ModifyAddressAttribute](#) di Referensi AWS CLI Perintah.

## modify-availability-zone-group

Contoh kode berikut menunjukkan cara menggunakan `modify-availability-zone-group`.

### AWS CLI

Untuk mengaktifkan grup zona

`modify-availability-zone-group` Contoh berikut memungkinkan kelompok zona yang ditentukan.

```
aws ec2 modify-availability-zone-group \
  --group-name us-west-2-lax-1 \
  --opt-in-status opted-in
```

Output:

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Wilayah dan Zona](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyAvailabilityZoneGroup](#) di Referensi AWS CLI Perintah.

## modify-capacity-reservation-fleet

Contoh kode berikut menunjukkan cara menggunakan `modify-capacity-reservation-fleet`.



## AWS CLI

### Contoh 1: Untuk memodifikasi total kapasitas target Armada Reservasi Kapasitas

`modify-capacity-reservation-fleet` Contoh berikut memodifikasi total kapasitas target Armada Reservasi Kapasitas yang ditentukan. Saat Anda memodifikasi total kapasitas target Armada Reservasi Kapasitas, Armada secara otomatis membuat Reservasi Kapasitas baru, atau memodifikasi atau membatalkan Reservasi Kapasitas yang ada di Armada untuk memenuhi total kapasitas target yang baru. Anda tidak dapat mencoba modifikasi tambahan pada Armada saat berada dalam status `modifying`.

```
aws ec2 modify-capacity-reservation-fleet \  
  --capacity-reservation-fleet-id crf-01234567890abcdef \  
  --total-target-capacity 160
```

#### Output:

```
{  
  "Return": true  
}
```

### Contoh 2: Untuk mengubah tanggal akhir Armada Reservasi Kapasitas

`modify-capacity-reservation-fleet` Contoh berikut memodifikasi tanggal akhir Armada Reservasi Kapasitas yang ditentukan. Ketika Anda memodifikasi tanggal akhir Armada, tanggal akhir untuk semua Reservasi Kapasitas individu akan diperbarui sesuai dengan modifikasi itu. Anda tidak dapat mencoba modifikasi tambahan pada Armada saat berada dalam status `modifying`.

```
aws ec2 modify-capacity-reservation-fleet \  
  --capacity-reservation-fleet-id crf-01234567890abcdef \  
  --end-date 2022-07-04T23:59:59.000Z
```

#### Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya tentang Armada Reservasi [Kapasitas](#), lihat [Armada Reservasi Kapasitas](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [ModifyCapacityReservationFleet](#) di Referensi AWS CLI Perintah.

## modify-capacity-reservation

Contoh kode berikut menunjukkan cara menggunakan `modify-capacity-reservation`.

### AWS CLI

Contoh 1: Untuk mengubah jumlah instans yang dicadangkan oleh reservasi kapasitas yang ada

`modify-capacity-reservation` Contoh berikut mengubah jumlah contoh yang kapasitas cadangan kapasitas reservasi.

```
aws ec2 modify-capacity-reservation \  
  --capacity-reservation-id cr-1234abcd56EXAMPLE \  
  --instance-count 5
```

Output:

```
{  
  "Return": true  
}
```

Contoh 2: Untuk mengubah tanggal dan waktu akhir reservasi kapasitas yang ada

`modify-capacity-reservation` Contoh berikut memodifikasi reservasi kapasitas yang ada untuk berakhir pada tanggal dan waktu yang ditentukan.

```
aws ec2 modify-capacity-reservation \  
  --capacity-reservation-id cr-1234abcd56EXAMPLE \  
  --end-date-type limited \  
  --end-date 2019-08-31T23:59:59Z
```

Untuk informasi selengkapnya, lihat [Memodifikasi Reservasi Kapasitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyCapacityReservation](#) di Referensi AWS CLI Perintah.

## modify-client-vpn-endpoint

Contoh kode berikut menunjukkan cara menggunakan `modify-client-vpn-endpoint`.

### AWS CLI

Untuk memodifikasi titik akhir Client VPN

`modify-client-vpn-endpoint` Contoh berikut memungkinkan pencatatan koneksi klien untuk titik akhir Client VPN yang ditentukan.

```
aws ec2 modify-client-vpn-endpoint \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --connection-log-options Enabled=true,CloudwatchLogGroup=ClientVPNLogs
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir Client VPN](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [ModifyClientVpnEndpoint](#) di Referensi AWS CLI Perintah.

## modify-default-credit-specification

Contoh kode berikut menunjukkan cara menggunakan `modify-default-credit-specification`.

### AWS CLI

Untuk memodifikasi opsi kredit default

`modify-default-credit-specification` Contoh berikut memodifikasi opsi kredit default untuk instance T2.

```
aws ec2 modify-default-credit-specification \  
  --instance-family t2 \  
  --cpu-credits unlimited
```

Output:

```
{
  "InstanceFamilyCreditSpecification": {
    "InstanceFamily": "t2",
    "CpuCredits": "unlimited"
  }
}
```

- Untuk detail API, lihat [ModifyDefaultCreditSpecification](#) di Referensi AWS CLI Perintah.

## **modify-ebs-default-kms-key-id**

Contoh kode berikut menunjukkan cara menggunakan `modify-ebs-default-kms-key-id`.

AWS CLI

Untuk mengatur CMK default Anda untuk enkripsi EBS

`modify-ebs-default-kms-key-id` Contoh berikut menetapkan CMK yang ditentukan sebagai CMK default untuk enkripsi EBS untuk AWS akun Anda di Wilayah saat ini.

```
aws ec2 modify-ebs-default-kms-key-id \
  --kms-key-id alias/my-cmk
```

Output:

```
{
  "KmsKeyId": "arn:aws:kms:us-
west-2:123456789012:key/0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE"
}
```

- Untuk detail API, lihat [ModifyEbsDefaultKmsKeyId](#) di Referensi AWS CLI Perintah.

## **modify-fleet**

Contoh kode berikut menunjukkan cara menggunakan `modify-fleet`.

AWS CLI

Untuk menskalakan Armada EC2

`modify-fleet` Contoh berikut memodifikasi kapasitas target Armada EC2 yang ditentukan. Jika nilai yang ditentukan lebih besar dari kapasitas saat ini, Armada EC2 meluncurkan instans tambahan. Jika nilai yang ditentukan kurang dari kapasitas saat ini, Armada EC2 membatalkan permintaan terbuka dan jika kebijakan penghentian tersebut `terminate`, armada EC2 menghentikan setiap instans yang melebihi kapasitas target baru.

```
aws ec2 modify-fleet \  
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \  
  --target-capacity-specification TotalTargetCapacity=5
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Armada EC2](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyFleet](#) di Referensi AWS CLI Perintah.

## **modify-fpga-image-attribute**

Contoh kode berikut menunjukkan cara menggunakan `modify-fpga-image-attribute`.

AWS CLI

Untuk memodifikasi atribut gambar Amazon FPGA

Contoh ini menambahkan izin muat untuk ID 123456789012 akun untuk AFI yang ditentukan.

Perintah:

```
aws ec2 modify-fpga-image-attribute --attribute loadPermission --fpga-image-id  
afi-0d123e123bfc85abc --load-permission Add=[{UserId=123456789012}]
```

Output:

```
{  
  "FpgaImageAttribute": {  
    "FpgaImageId": "afi-0d123e123bfc85abc",
```

```
    "LoadPermissions": [  
      {  
        "UserId": "123456789012"  
      }  
    ]  
  }  
}
```

- Untuk detail API, lihat [ModifyFpgaImageAttribute](#) di Referensi AWS CLI Perintah.

## modify-hosts

Contoh kode berikut menunjukkan cara menggunakan `modify-hosts`.

### AWS CLI

Contoh 1: Untuk mengaktifkan penempatan otomatis untuk Host Khusus

`modify-hosts` Contoh berikut memungkinkan penempatan otomatis untuk Host Khusus sehingga menerima peluncuran instans yang tidak ditargetkan yang cocok dengan konfigurasi tipe instance-nya.

```
aws ec2 modify-hosts \  
  --host-id h-06c2f189b4EXAMPLE \  
  --auto-placement on
```

Output:

```
{  
  "Successful": [  
    "h-06c2f189b4EXAMPLE"  
  ],  
  "Unsuccessful": []  
}
```

Contoh 2: Untuk mengaktifkan pemulihan host untuk Host Khusus

`modify-hosts` Contoh berikut memungkinkan pemulihan host untuk Host Khusus yang ditentukan.

```
aws ec2 modify-hosts \  
  --host-id h-06c2f189b4EXAMPLE \  
  --auto-placement on
```

```
--host-id h-06c2f189b4EXAMPLE \  
--host-recovery on
```

Output:

```
{  
  "Successful": [  
    "h-06c2f189b4EXAMPLE"  
  ],  
  "Unsuccessful": []  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Penempatan Otomatis Host Khusus](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyHosts](#) di Referensi AWS CLI Perintah.

## modify-id-format

Contoh kode berikut menunjukkan cara menggunakan `modify-id-format`.

### AWS CLI

Untuk mengaktifkan format ID yang lebih panjang untuk sumber daya

`modify-id-format` Contoh berikut memungkinkan format ID yang lebih panjang untuk jenis `instance` sumber daya.

```
aws ec2 modify-id-format \  
  --resource instance \  
  --use-long-ids
```

Untuk menonaktifkan format ID yang lebih panjang untuk sumber daya

`modify-id-format` Contoh berikut menonaktifkan format ID yang lebih panjang untuk jenis `instance` sumber daya.

```
aws ec2 modify-id-format \  
  --resource instance \  
  --no-use-long-ids
```

`modify-id-format` Contoh berikut memungkinkan format ID yang lebih panjang untuk semua jenis sumber daya yang didukung yang berada dalam periode keikutsertaannya.

```
aws ec2 modify-id-format \  
  --resource all-current \  
  --use-long-ids
```

- Untuk detail API, lihat [ModifyIdFormat](#) di Referensi AWS CLI Perintah.

## `modify-identity-id-format`

Contoh kode berikut menunjukkan cara menggunakan `modify-identity-id-format`.

### AWS CLI

Untuk mengaktifkan peran IAM untuk menggunakan ID yang lebih panjang untuk sumber daya

`modify-identity-id-format` Contoh berikut memungkinkan peran IAM `EC2Role` di AWS akun Anda untuk menggunakan format ID panjang untuk jenis instance sumber daya.

```
aws ec2 modify-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:role/EC2Role \  
  --resource instance \  
  --use-long-ids
```

Untuk mengaktifkan pengguna IAM menggunakan ID yang lebih panjang untuk sumber daya

`modify-identity-id-format` Contoh berikut memungkinkan pengguna IAM `AdminUser` di AWS akun Anda untuk menggunakan format ID yang lebih panjang untuk jenis volume sumber daya.

```
aws ec2 modify-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:user/AdminUser \  
  --resource volume \  
  --use-long-ids
```

`modify-identity-id-format` Contoh berikut memungkinkan pengguna IAM `AdminUser` di AWS akun Anda untuk menggunakan format ID yang lebih panjang untuk semua jenis sumber daya yang didukung yang berada dalam periode keikutsertaannya.



```
aws ec2 modify-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:user/AdminUser \  
  --resource all-current \  
  --use-long-ids
```

- Untuk detail API, lihat [ModifyIdentityIdFormat](#) di Referensi AWS CLI Perintah.

## modify-image-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-image-attribute`.

### AWS CLI

Contoh 1: Untuk membuat AMI publik

`modify-image-attribute` Contoh berikut membuat AMI yang ditentukan publik.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Add=[{Group=all}]"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk membuat AMI pribadi

`modify-image-attribute` Contoh berikut membuat AMI yang ditentukan menjadi pribadi.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Remove=[{Group=all}]"
```

Perintah ini tidak menghasilkan output.

Contoh 3: Untuk memberikan izin peluncuran ke AWS akun

`modify-image-attribute` Contoh berikut memberikan izin peluncuran ke akun yang ditentukan AWS .

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Add=[{UserId=123456789012}]"
```

Perintah ini tidak menghasilkan output.

Contoh 4: Untuk menghapus izin peluncuran dari AWS akun

`modify-instance-attribute` Contoh berikut menghapus izin peluncuran dari AWS akun yang ditentukan.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Remove=[{UserId=123456789012}]"
```

- Untuk detail API, lihat [ModifyImageAttribute](#) di Referensi AWS CLI Perintah.

## **modify-instance-attribute**

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-attribute`.

### AWS CLI

Contoh 1: Untuk memodifikasi jenis instance

`modify-instance-attribute` Contoh berikut memodifikasi jenis instance dari instance yang ditentukan. Instans harus berada dalam status `stopped`.

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --instance-type "{\"Value\": \"m1.small\"}"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk mengaktifkan jaringan yang disempurnakan pada sebuah instance

`modify-instance-attribute` Contoh berikut memungkinkan jaringan ditingkatkan untuk contoh yang ditentukan. Instans harus berada dalam status `stopped`.

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --sriov-net-support simple
```

Perintah ini tidak menghasilkan output.

Contoh 3: Untuk memodifikasi `sourceDestCheck` atribut

`modify-instance-attribute` Contoh berikut menetapkan `sourceDestCheck` atribut dari contoh yang ditentukan untuk `true`. Instans harus dalam VPC.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --source-dest-check "{\"Value\": true}"
```

Perintah ini tidak menghasilkan output.

Contoh 4: Untuk memodifikasi `deleteOnTermination` atribut volume root

`modify-instance-attribute` Contoh berikut menyetel `deleteOnTermination` atribut untuk volume root instance yang didukung Amazon EBS yang ditentukan. `false` Secara default, atribut ini adalah `true` untuk volume root.

Perintah:

```
aws ec2 modify-instance-attribute \
  --instance-id i-1234567890abcdef0 \
  --block-device-mappings "[{\"DeviceName\": \"/dev/sda1\", \"Ebs\": {\"DeleteOnTermination\": false}}]"
```

Perintah ini tidak menghasilkan output.

Contoh 5: Untuk memodifikasi data pengguna yang dilampirkan ke sebuah instance

`modify-instance-attribute` Contoh berikut menambahkan isi file `UserData.txt` sebagai `UserData` untuk contoh yang ditentukan.

Isi file asli `UserData.txt`:

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

Isi file harus dikodekan base64. Perintah pertama mengonversi file teks ke base64 dan menyimpannya sebagai file baru.

Versi perintah Linux/macOS:

```
base64 UserData.txt > UserData.base64.txt
```

Perintah ini tidak menghasilkan output.

Versi Windows dari perintah:

```
certutil -encode UserData.txt tmp.b64 && findstr /v /c:- tmp.b64 >
UserData.base64.txt
```

Output:

```
Input Length = 67
Output Length = 152
CertUtil: -encode command completed successfully.
```

Sekarang Anda dapat mereferensikan file itu dalam perintah CLI yang berikut:

```
aws ec2 modify-instance-attribute \
  --instance-id=i-09b5a14dbca622e76 \
  --attribute userData --value file://UserData.base64.txt
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Data Pengguna dan AWS CLI](#) di Panduan Pengguna EC2.

- Untuk detail API, lihat [ModifyInstanceAttribute](#) di Referensi AWS CLI Perintah.

## **modify-instance-capacity-reservation-attributes**

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-capacity-reservation-attributes`.

### AWS CLI

Contoh 1: Untuk mengubah setelan penargetan reservasi kapasitas instans

`modify-instance-capacity-reservation-attributes` Contoh berikut memodifikasi instance berhenti untuk menargetkan reservasi kapasitas tertentu.

```
aws ec2 modify-instance-capacity-reservation-attributes \
  --instance-id i-EXAMPLE8765abcd4e \
  --capacity-reservation-specification
  'CapacityReservationTarget={CapacityReservationId= cr-1234abcd56EXAMPLE }'
```

**Output:**

```
{
  "Return": true
}
```

Contoh 2: Untuk mengubah setelan penargetan reservasi kapasitas instans

`modify-instance-capacity-reservation-attributes` Contoh berikut memodifikasi instance berhenti yang menargetkan reservasi kapasitas yang ditentukan untuk diluncurkan di reservasi kapasitas apa pun yang memiliki atribut yang cocok (tipe instans, platform, Availability Zone) dan yang memiliki kriteria pencocokan instance terbuka.

```
aws ec2 modify-instance-capacity-reservation-attributes \
  --instance-id i-EXAMPLE8765abcd4e \
  --capacity-reservation-specification 'CapacityReservationPreference=open'
```

**Output:**

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Setelan Reservasi Kapasitas Instans](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyInstanceCapacityReservationAttributes](#) di Referensi AWS CLI Perintah.

**modify-instance-credit-specification**

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-credit-specification`.

**AWS CLI**

Untuk memodifikasi opsi kredit untuk penggunaan CPU dari sebuah instance

Contoh ini memodifikasi opsi kredit untuk penggunaan CPU dari instance yang ditentukan di wilayah tertentu menjadi “tidak terbatas”. Opsi kredit yang valid adalah “standar” dan “tidak terbatas”.

**Perintah:**

```
aws ec2 modify-instance-credit-specification --instance-credit-specification
"InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

**Output:**

```
{
  "SuccessfulInstanceCreditSpecifications": [
    {
      "InstanceId": "i-1234567890abcdef0"
    }
  ],
  "UnsuccessfulInstanceCreditSpecifications": []
}
```

- Untuk detail API, lihat [ModifyInstanceCreditSpecification](#) di Referensi AWS CLI Perintah.

**modify-instance-event-start-time**

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-event-start-time`.

**AWS CLI**

Untuk memodifikasi waktu mulai acara untuk sebuah instance

`modify-instance-event-start-time` Perintah berikut menunjukkan cara memodifikasi waktu mulai acara untuk contoh yang ditentukan. Tentukan ID acara dengan menggunakan `--instance-event-id` parameter. Tentukan tanggal dan waktu baru dengan menggunakan `--not-before` parameter.

```
aws ec2 modify-instance-event-start-time --instance-id i-1234567890abcdef0
--instance-event-id instance-event-0abcdef1234567890 --not-before
2019-03-25T10:00:00.000
```

**Output:**

```
"Event": {
  "InstanceEventId": "instance-event-0abcdef1234567890",
  "Code": "system-reboot",
```

```

    "Description": "scheduled reboot",
    "NotAfter": "2019-03-25T12:00:00.000Z",
    "NotBefore": "2019-03-25T10:00:00.000Z",
    "NotBeforeDeadline": "2019-04-22T21:00:00.000Z"
  }

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Instans yang Dijadwalkan untuk Reboot](#) di Panduan Pengguna Amazon Elastic Compute Cloud

- Untuk detail API, lihat [ModifyInstanceEventStartTime](#) di Referensi AWS CLI Perintah.

## modify-instance-event-window

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-event-window`.

### AWS CLI

Contoh 1: Untuk mengubah rentang waktu jendela acara

`modify-instance-event-window` Contoh berikut memodifikasi rentang waktu jendela acara. Tentukan parameter `time-range` untuk memodifikasi rentang waktu. Anda juga tidak dapat menentukan parameter `cron-expression`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890
  --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8

```

Output:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "TimeRanges": [
      {
        "StartWeekDay": "monday",
        "StartHour": 2,
        "EndWeekDay": "wednesday",
        "EndHour": 8
      }
    ],
    "Name": "myEventWindowName",
  }
}

```

```

    "AssociationTarget": {
      "InstanceIds": [
        "i-0abcdef1234567890",
        "i-0be35f9acb8ba01f0"
      ],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 2: Untuk memodifikasi serangkaian rentang waktu untuk jendela acara

`modify-instance-event-window` Contoh berikut memodifikasi rentang waktu jendela acara. Tentukan parameter `time-range` untuk memodifikasi rentang waktu. Anda juga tidak dapat menentukan parameter `cron-expression`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --time-range '[{"StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
"wednesday", "EndHour": 8},
  {"StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
"EndHour": 8}]'

```

Output:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "TimeRanges": [
      {

```



```

        "StartWeekDay": "monday",
        "StartHour": 2,
        "EndWeekDay": "wednesday",
        "EndHour": 8
    },
    {
        "StartWeekDay": "thursday",
        "StartHour": 2,
        "EndWeekDay": "friday",
        "EndHour": 8
    }
],
"Name": "myEventWindowName",
"AssociationTarget": {
    "InstanceIds": [
        "i-0abcdef1234567890",
        "i-0be35f9acb8ba01f0"
    ],
    "Tags": [],
    "DedicatedHostIds": []
},
"State": "creating",
"Tags": [
    {
        "Key": "K1",
        "Value": "V1"
    }
]
}
}

```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

Contoh 3: Untuk memodifikasi ekspresi cron dari jendela acara

`modify-instance-event-window` Contoh berikut memodifikasi ekspresi cron dari jendela acara. Tentukan parameter `cron-expression` untuk memodifikasi ekspresi cron. Anda juga tidak dapat menentukan parameter `time-range`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \

```

```
--cron-expression "* 21-23 * * 2,3"
```

Output:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [
        "i-0abcdef1234567890",
        "i-0be35f9acb8ba01f0"
      ],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}
```

Untuk batasan jendela acara, lihat [Pertimbangan](#) di bagian Acara Terjadwal pada Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [ModifyInstanceEventWindow](#) di Referensi AWS CLI Perintah.

## modify-instance-maintenance-options

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-maintenance-options`.

### AWS CLI

Contoh 1: Untuk menonaktifkan perilaku pemulihan sebuah instance

`modify-instance-maintenance-options` Contoh berikut menonaktifkan pemulihan otomatis yang disederhanakan untuk instance yang berjalan atau berhenti.

```
aws ec2 modify-instance-maintenance-options \  
  --instance-id i-0abcdef1234567890 \  
  --auto-recovery disabled
```

Output:

```
{  
  "InstanceId": "i-0abcdef1234567890",  
  "AutoRecovery": "disabled"  
}
```

Untuk informasi selengkapnya, lihat [Memulihkan instans Anda](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Contoh 2: Untuk mengatur perilaku pemulihan instance ke default

`modify-instance-maintenance-options` Contoh berikut menetapkan perilaku pemulihan otomatis ke default yang memungkinkan pemulihan otomatis disederhanakan untuk jenis instans yang didukung.

```
aws ec2 modify-instance-maintenance-options \  
  --instance-id i-0abcdef1234567890 \  
  --auto-recovery default
```

Output:

```
{  
  "InstanceId": "i-0abcdef1234567890",  
  "AutoRecovery": "default"  
}
```

Untuk informasi selengkapnya, lihat [Memulihkan instans Anda](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.

- Untuk detail API, lihat [ModifyInstanceMaintenanceOptions](#) di Referensi AWS CLI Perintah.

## **modify-instance-metadata-options**

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-metadata-options`.

## AWS CLI

### Contoh 1: Untuk mengaktifkan IMDSv2

`modify-instance-metadata-options` Contoh berikut mengkonfigurasi penggunaan IMDSv2 pada contoh yang ditentukan.

```
aws ec2 modify-instance-metadata-options \  
  --instance-id i-1234567898abcdef0 \  
  --http-tokens required \  
  --http-endpoint enabled
```

### Output:

```
{  
  "InstanceId": "i-1234567898abcdef0",  
  "InstanceMetadataOptions": {  
    "State": "pending",  
    "HttpTokens": "required",  
    "HttpPutResponseHopLimit": 1,  
    "HttpEndpoint": "enabled"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Metadana instans dan data pengguna](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

### Contoh 2: Untuk menonaktifkan metadata instance

`modify-instance-metadata-options` Contoh berikut menonaktifkan penggunaan semua versi metadata instance pada instance tertentu.

```
aws ec2 modify-instance-metadata-options \  
  --instance-id i-1234567898abcdef0 \  
  --http-endpoint disabled
```

### Output:

```
{  
  "InstanceId": "i-1234567898abcdef0",
```

```
"InstanceMetadataOptions": {
  "State": "pending",
  "HttpTokens": "required",
  "HttpPutResponseHopLimit": 1,
  "HttpEndpoint": "disabled"
}
```

Untuk informasi selengkapnya, lihat [Metadana instans dan data pengguna](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 3: Untuk mengaktifkan metadata instance IPv6 endpoint untuk instans Anda

`modify-instance-metadata-options` Contoh berikut menunjukkan cara mengaktifkan titik akhir IPv6 untuk layanan metadata instance.

```
aws ec2 modify-instance-metadata-options \
  --instance-id i-1234567898abcdef0 \
  --http-protocol-ipv6 enabled \
  --http-endpoint enabled
```

Output:

```
{
  "InstanceId": "i-1234567898abcdef0",
  "InstanceMetadataOptions": {
    "State": "pending",
    "HttpTokens": "required",
    "HttpPutResponseHopLimit": 1,
    "HttpEndpoint": "enabled",
    "HttpProtocolIpv6": "enabled"
  }
}
```

Secara default, titik akhir IPv6 dinonaktifkan. Hal ini juga akan terjadi bahkan jika Anda telah meluncurkan instans ke subnet khusus IPv6. Titik akhir IPv6 untuk IMDS hanya dapat diakses pada instans yang dibangun di atas Sistem Nitro. Untuk informasi selengkapnya, lihat [Metadana instans dan data pengguna](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyInstanceMetadataOptions](#) di Referensi AWS CLI Perintah.

## modify-instance-placement

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-placement`.

### AWS CLI

Contoh 1: Untuk menghapus afinitas instance dengan Host Khusus

`modify-instance-placement` Contoh berikut menghapus afinitas instans dengan Host Khusus dan memungkinkannya diluncurkan pada Host Khusus yang tersedia di akun Anda yang mendukung jenis instance-nya.

```
aws ec2 modify-instance-placement \  
  --instance-id i-0e6ddf6187EXAMPLE \  
  --affinity default
```

Output:

```
{  
  "Return": true  
}
```

Contoh 2: Untuk membangun afinitas antara instance dan Host Khusus yang ditentukan

`modify-instance-placement` Contoh berikut menetapkan hubungan peluncuran antara instance dan Host Khusus. Instance hanya dapat berjalan pada Host Khusus yang ditentukan.

```
aws ec2 modify-instance-placement \  
  --instance-id i-0e6ddf6187EXAMPLE \  
  --affinity host \  
  --host-id i-0e6ddf6187EXAMPLE
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Penyewaan Instance dan Afinitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

### Contoh 3: Untuk memindahkan instance ke grup penempatan

`modify-instance-placement` Contoh berikut memindahkan instance ke grup penempatan, menghentikan instance, memodifikasi penempatan instance, dan kemudian restart instance.

```
aws ec2 stop-instances \  
  --instance-ids i-0123a456700123456  
  
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name MySpreadGroup  
  
aws ec2 start-instances \  
  --instance-ids i-0123a456700123456
```

Untuk informasi selengkapnya, lihat [Mengubah Grup Penempatan untuk Instance](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

### Contoh 4: Untuk menghapus instance dari grup penempatan

`modify-instance-placement` Contoh berikut menghapus instance dari grup penempatan dengan menghentikan instance, memodifikasi penempatan instance, dan kemudian memulai ulang instance. Contoh berikut menentukan string kosong ("" ) untuk nama grup penempatan untuk menunjukkan bahwa instance ini tidak akan berada di grup penempatan.

Hentikan instance:

```
aws ec2 stop-instances \  
  --instance-ids i-0123a456700123456
```

Ubah penempatan (Windows Command Prompt, Linux, dan macOS):

```
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name ""
```

Ubah penempatan (Windows PowerShell):

```
aws ec2 modify-instance-placement `  
  --instance-id i-0123a456700123456 `
```

```
--group-name ""
```

Mulai ulang instance:

```
aws ec2 start-instances \  
  --instance-ids i-0123a456700123456
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Penyewaan Instance dan Afinitas](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [ModifyInstancePlacement](#) di Referensi AWS CLI Perintah.

## modify-ipam-pool

Contoh kode berikut menunjukkan cara menggunakan `modify-ipam-pool`.

### AWS CLI

Untuk memodifikasi kolam IPAM

`modify-ipam-pool` Contoh berikut memodifikasi kolam IPAM.

(Linux):

```
aws ec2 modify-ipam-pool \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --add-allocation-resource-tags "Key=Owner,Value=Build Team" \  
  --clear-allocation-default-netmask-length \  
  --allocation-min-netmask-length 14
```

(Windows):

```
aws ec2 modify-ipam-pool ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --add-allocation-resource-tags "Key=Owner,Value=Build Team" ^  
  --clear-allocation-default-netmask-length ^
```



```
--allocation-min-netmask-length 14
```

Output:

```
{
  "IpamPool": {
    "OwnerId": "123456789012",
    "IpamPoolId": "ipam-pool-0533048da7d823723",
    "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0533048da7d823723",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
    "IpamScopeType": "private",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "Locale": "None",
    "PoolDepth": 1,
    "State": "modify-complete",
    "AutoImport": true,
    "AddressFamily": "ipv4",
    "AllocationMinNetmaskLength": 14,
    "AllocationMaxNetmaskLength": 26,
    "AllocationResourceTags": [
      {
        "Key": "Environment",
        "Value": "Preprod"
      },
      {
        "Key": "Owner",
        "Value": "Build Team"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Mengedit kumpulan](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [ModifyIpamPool](#) di Referensi AWS CLI Perintah.

## **modify-ipam-resource-cidr**

Contoh kode berikut menunjukkan cara menggunakan `modify-ipam-resource-cidr`.

## AWS CLI

Untuk memodifikasi CIDR yang dialokasikan ke sumber daya

`modify-ipam-resource-cidr` Contoh berikut memodifikasi sumber daya CIDR.

(Linux):

```
aws ec2 modify-ipam-resource-cidr \  
  --current-ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \  
  --destination-ipam-scope-id ipam-scope-0da34c61fd189a141 \  
  --resource-id vpc-010e1791024eb0af9 \  
  --resource-cidr 10.0.1.0/24 \  
  --resource-region us-east-1 \  
  --monitored
```

(Windows):

```
aws ec2 modify-ipam-resource-cidr ^  
  --current-ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^  
  --destination-ipam-scope-id ipam-scope-0da34c61fd189a141 ^  
  --resource-id vpc-010e1791024eb0af9 ^  
  --resource-cidr 10.0.1.0/24 ^  
  --resource-region us-east-1 ^  
  --monitored
```

Output:

```
{  
  "IpamResourceCidr": {  
    "IpamId": "ipam-08440e7a3acde3908",  
    "IpamScopeId": "ipam-scope-0da34c61fd189a141",  
    "IpamPoolId": "ipam-pool-0533048da7d823723",  
    "ResourceRegion": "us-east-1",  
    "ResourceOwnerId": "123456789012",  
    "ResourceId": "vpc-010e1791024eb0af9",  
    "ResourceCidr": "10.0.1.0/24",  
    "ResourceType": "vpc",  
    "ResourceTags": [  
      {  
        "Key": "Environment",  
        "Value": "Preprod"  
      },  
    ],  
  },  
}
```

```
        {
            "Key": "Owner",
            "Value": "Build Team"
        }
    ],
    "IpUsage": 0.0,
    "ComplianceStatus": "noncompliant",
    "ManagementState": "managed",
    "OverlapStatus": "overlapping",
    "VpcId": "vpc-010e1791024eb0af9"
}
}
```

Untuk informasi selengkapnya tentang memindahkan sumber daya, lihat [Memindahkan CIDR sumber daya antar cakupan di Panduan Pengguna Amazon VPC IPAM](#).

Untuk informasi selengkapnya tentang mengubah status pemantauan, lihat [Mengubah status pemantauan CIDR sumber daya di Panduan Pengguna Amazon VPC IPAM](#).

- Untuk detail API, lihat [ModifyIpamResourceCidr](#) di Referensi AWS CLI Perintah.

## **modify-ipam-resource-discovery**

Contoh kode berikut menunjukkan cara menggunakan `modify-ipam-resource-discovery`.

### AWS CLI

Untuk memodifikasi wilayah operasi penemuan sumber daya

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM yang ingin memodifikasi wilayah operasi penemuan sumber daya.

Untuk melengkapi permintaan ini:

Anda tidak dapat mengubah penemuan sumber daya default dan Anda harus menjadi pemilik penemuan sumber daya. Anda memerlukan ID penemuan sumber daya, yang bisa Anda dapatkan. [describe-ipam-resource-discoveries](#)

`modify-ipam-resource-discovery` Contoh berikut memodifikasi penemuan sumber daya non-default di akun Anda AWS .

```
aws ec2 modify-ipam-resource-discovery \
    --ipam-resource-discovery-id ipam-res-disco-0f4ef577a9f37a162 \
```

```
--add-operating-regions RegionName='us-west-1' \  
--remove-operating-regions RegionName='us-east-2' \  
--region us-east-1
```

## Output:

```
{  
  "IpamResourceDiscovery": {  
    "OwnerId": "149977607591",  
    "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",  
    "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-  
discovery/ipam-res-disco-0365d2977fc1672fe",  
    "IpamResourceDiscoveryRegion": "us-east-1",  
    "Description": "Example",  
    "OperatingRegions": [  
      {  
        "RegionName": "us-east-1"  
      },  
      {  
        "RegionName": "us-west-1"  
      }  
    ],  
    "IsDefault": false,  
    "State": "modify-in-progress"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan penemuan sumber daya di Panduan Pengguna Amazon VPC IPAM](#).

- Untuk detail API, lihat [ModifyIpamResourceDiscovery](#) di Referensi AWS CLI Perintah.

## modify-ipam-scope

Contoh kode berikut menunjukkan cara menggunakan `modify-ipam-scope`.

### AWS CLI

Untuk memodifikasi deskripsi ruang lingkup

Dalam skenario ini, Anda adalah admin yang didelegasikan IPAM yang ingin memodifikasi deskripsi cakupan IPAM.

Untuk menyelesaikan permintaan ini, Anda memerlukan ID cakupan, yang bisa Anda dapatkan [describe-ipam-scopes](#).

`modify-ipam-scope` Contoh berikut memperbarui deskripsi ruang lingkup.

```
aws ec2 modify-ipam-scope \  
  --ipam-scope-id ipam-scope-0d3539a30b57dcdd1 \  
  --description example \  
  --region us-east-1
```

Output:

```
{  
  "IpamScope": {  
    "OwnerId": "320805250157",  
    "IpamScopeId": "ipam-scope-0d3539a30b57dcdd1",  
    "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-  
scope-0d3539a30b57dcdd1",  
    "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",  
    "IpamRegion": "us-east-1",  
    "IpamScopeType": "public",  
    "IsDefault": true,  
    "Description": "example",  
    "PoolCount": 1,  
    "State": "modify-in-progress"  
  }  
}
```

Untuk informasi selengkapnya tentang cakupan, lihat [Cara kerja IPAM](#) di Panduan Pengguna Amazon VPC IPAM.

- Untuk detail API, lihat [ModifyIpamScope](#) di Referensi AWS CLI Perintah.

## **modify-ipam**

Contoh kode berikut menunjukkan cara menggunakan `modify-ipam`.

### AWS CLI

Untuk memodifikasi IPAM

`modify-ipam` Contoh berikut memodifikasi IPAM dengan menambahkan Wilayah Operasi.

(Linux):

```
aws ec2 modify-ipam \  
  --ipam-id ipam-08440e7a3acde3908 \  
  --add-operating-regions RegionName=us-west-2
```

(Windows):

```
aws ec2 modify-ipam ^  
  --ipam-id ipam-08440e7a3acde3908 ^  
  --add-operating-regions RegionName=us-west-2
```

Output:

```
{  
  "Ipam": {  
    "OwnerId": "123456789012",  
    "IpamId": "ipam-08440e7a3acde3908",  
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",  
    "IpamRegion": "us-east-1",  
    "PublicDefaultScopeId": "ipam-scope-0b9eed026396dbc16",  
    "PrivateDefaultScopeId": "ipam-scope-02fc38cd4c48e7d38",  
    "ScopeCount": 3,  
    "OperatingRegions": [  
      {  
        "RegionName": "us-east-1"  
      },  
      {  
        "RegionName": "us-east-2"  
      },  
      {  
        "RegionName": "us-west-1"  
      },  
      {  
        "RegionName": "us-west-2"  
      }  
    ],  
    "State": "modify-in-progress"  
  }  
}
```

- Untuk detail API, lihat [ModifyIpam](#) di Referensi AWS CLI Perintah.

## modify-launch-template

Contoh kode berikut menunjukkan cara menggunakan `modify-launch-template`.

### AWS CLI

Untuk mengubah versi template peluncuran default

Contoh ini menentukan versi 2 dari template peluncuran yang ditentukan sebagai versi default.

Perintah:

```
aws ec2 modify-launch-template --launch-template-id lt-0abcd290751193123 --default-version 2
```

Output:

```
{
  "LaunchTemplate": {
    "LatestVersionNumber": 2,
    "LaunchTemplateId": "lt-0abcd290751193123",
    "LaunchTemplateName": "WebServers",
    "DefaultVersionNumber": 2,
    "CreatedBy": "arn:aws:iam::123456789012:root",
    "CreateTime": "2017-12-01T13:35:46.000Z"
  }
}
```

- Untuk detail API, lihat [ModifyLaunchTemplate](#) di Referensi AWS CLI Perintah.

## modify-managed-prefix-list

Contoh kode berikut menunjukkan cara menggunakan `modify-managed-prefix-list`.

### AWS CLI

Untuk memodifikasi daftar awalan

`modify-managed-prefix-list` Contoh berikut menambahkan entri ke daftar awalan yang ditentukan.

```
aws ec2 modify-managed-prefix-list \
```

```
--prefix-list-id pl-0123456abcabcabc1 \  
--add-entries Cidr=10.1.0.0/16,Description=vpc-c \  
--current-version 1
```

### Output:

```
{  
  "PrefixList": {  
    "PrefixListId": "pl-0123456abcabcabc1",  
    "AddressFamily": "IPv4",  
    "State": "modify-in-progress",  
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/  
pl-0123456abcabcabc1",  
    "PrefixListName": "vpc-cidrs",  
    "MaxEntries": 10,  
    "Version": 1,  
    "OwnerId": "123456789012"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Daftar awalan terkelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [ModifyManagedPrefixList](#) di Referensi AWS CLI Perintah.

## modify-network-interface-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-network-interface-attribute`.

### AWS CLI

Untuk memodifikasi atribut lampiran antarmuka jaringan

Perintah contoh ini memodifikasi `attachment` atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --  
attachment AttachmentId=eni-attach-43348162,DeleteOnTermination=false
```

Untuk memodifikasi atribut deskripsi antarmuka jaringan

Perintah contoh ini memodifikasi `description` atribut antarmuka jaringan yang ditentukan.



Perintah:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --description "My description"
```

Untuk memodifikasi atribut GroupSet dari antarmuka jaringan

Perintah contoh ini memodifikasi groupSet atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --groups sg-903004f8 sg-1a2b3c4d
```

Untuk memodifikasi sourceDestCheck atribut antarmuka jaringan

Perintah contoh ini memodifikasi sourceDestCheck atribut antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --no-source-dest-check
```

- Untuk detail API, lihat [ModifyNetworkInterfaceAttribute](#) di Referensi AWS CLI Perintah.

## modify-private-dns-name-options

Contoh kode berikut menunjukkan cara menggunakan `modify-private-dns-name-options`.

AWS CLI

Untuk memodifikasi opsi untuk nama host misalnya

`modify-private-dns-name-options` Contoh berikut menonaktifkan opsi untuk menanggapi kueri DNS misalnya nama host dengan catatan DNS A.

```
aws ec2 modify-private-dns-name-options \
  --instance-id i-1234567890abcdef0 \
  --no-enable-resource-name-dns-a-record
```

Output:

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [jenis nama host instans Amazon EC2 di Panduan Pengguna Amazon EC2](#).

- Untuk detail API, lihat [ModifyPrivateDnsNameOptions](#) di Referensi AWS CLI Perintah.

## **modify-reserved-instances**

Contoh kode berikut menunjukkan cara menggunakan `modify-reserved-instances`.

### AWS CLI

Untuk memodifikasi Instans Cadangan

Perintah contoh ini memindahkan Instans Cadangan ke Availability Zone lain di wilayah yang sama.

Perintah:

```
aws ec2 modify-reserved-instances --reserved-instances-ids b847fa93-e282-4f55-
b59a-1342f5bd7c02 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-
Classic,InstanceCount=10
```

Output:

```
{
  "ReservedInstancesModificationId": "rimod-d3ed4335-b1d3-4de6-ab31-0f13aaf46687"
}
```

Untuk memodifikasi platform jaringan Instans Cadangan

Perintah contoh ini mengonversi EC2-Classic Reserved Instances ke EC2-VPC.

Perintah:

```
aws ec2 modify-reserved-instances --reserved-instances-ids f127bd27-edb7-44c9-
a0eb-0d7e09259af0 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-
VPC,InstanceCount=5
```

**Output:**

```
{
  "ReservedInstancesModificationId": "rimod-82fa9020-668f-4fb6-945d-61537009d291"
}
```

Untuk informasi selengkapnya, lihat Memodifikasi Instans Cadangan Anda di Panduan Pengguna Amazon EC2.

Untuk mengubah ukuran instans dari Instans Cadangan

Perintah contoh ini memodifikasi Instans Cadangan yang memiliki 10 instance Linux/Unix m1.small di us-west-1c sehingga 8 instance m1.small menjadi 2 instance m1.large, dan 2 m1.small sisanya menjadi 1 m1.medium instance di Availability Zone yang sama. Perintah:

```
aws ec2 modify-reserved-instances --reserved-instances-ids
1ba8e2e3-3556-4264-949e-63ee671405a9 --target-configurations AvailabilityZone=us-
west-1c,Platform=EC2-Classic,InstanceCount=2,InstanceType=m1.large
AvailabilityZone=us-west-1c,Platform=EC2-
Classic,InstanceCount=1,InstanceType=m1.medium
```

**Output:**

```
{
  "ReservedInstancesModificationId": "rimod-acc5f240-080d-4717-b3e3-1c6b11fa00b6"
}
```

Untuk informasi selengkapnya, lihat Memodifikasi Ukuran Instans Reservasi Anda di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [ModifyReservedInstances](#) di Referensi AWS CLI Perintah.

**modify-security-group-rules**

Contoh kode berikut menunjukkan cara menggunakan `modify-security-group-rules`.

**AWS CLI**

Untuk memodifikasi aturan grup keamanan untuk memperbarui deskripsi aturan, protokol IP, dan rentang alamat CidrIpv4

`modify-security-group-rules` Contoh berikut memperbarui deskripsi, protokol IP, dan rentang CIDR IPV4 dari aturan grup keamanan tertentu. Gunakan `security-group-rules` parameter untuk memasukkan pembaruan untuk aturan grup keamanan yang ditentukan.

- 1menentukan semua protokol.

```
aws ec2 modify-security-group-rules \  
  --group-id sg-1234567890abcdef0 \  
  --security-group-rules SecurityGroupId=sgr-  
  abcdef01234567890,SecurityGroupRule='{Description=test,IpProtocol=-1,CidrIpv4=0.0.0.0/0}'
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya tentang aturan grup [keamanan](#), lihat [Aturan grup keamanan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [ModifySecurityGroupRules](#) di Referensi AWS CLI Perintah.

## **modify-snapshot-attribute**

Contoh kode berikut menunjukkan cara menggunakan `modify-snapshot-attribute`.

### AWS CLI

Contoh 1: Untuk memodifikasi atribut snapshot

`modify-snapshot-attribute` Contoh berikut memperbarui `createVolumePermission` atribut untuk snapshot yang ditentukan, menghapus izin volume untuk pengguna tertentu.

```
aws ec2 modify-snapshot-attribute \  
  --snapshot-id snap-1234567890abcdef0 \  
  --attribute createVolumePermission \  
  --operation-type remove \  
  --user-ids 123456789012
```

Contoh 2: Untuk membuat snapshot publik

`modify-snapshot-attribute` Contoh berikut membuat snapshot yang ditentukan publik.

```
aws ec2 modify-snapshot-attribute \  
  --snapshot-id snap-1234567890abcdef0 \  
  --attribute createVolumePermission \  
  --operation-type add \  
  --group-names all
```

- Untuk detail API, lihat [ModifySnapshotAttribute](#) di Referensi AWS CLI Perintah.

## modify-snapshot-tier

Contoh kode berikut menunjukkan cara menggunakan `modify-snapshot-tier`.

### AWS CLI

Contoh 1: Untuk mengarsipkan snapshot

`modify-snapshot-tier` Contoh berikut mengarsipkan snapshot yang ditentukan.

```
aws ec2 modify-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --storage-tier archive
```

Output:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "TieringStartTime": "2021-09-15T16:44:37.574Z"  
}
```

Parameter respons `TieringStartTime` menunjukkan tanggal dan waktu proses arsip dimulai, dalam format waktu UTC (YYY-MM-DDTHH:MM:SSZ).

Untuk informasi selengkapnya tentang pengarsipan snapshot, lihat Mengarsipkan [snapshot Amazon EBS di Panduan Pengguna Amazon EC2](#).

- Untuk detail API, lihat [ModifySnapshotTier](#) di Referensi AWS CLI Perintah.

## modify-spot-fleet-request

Contoh kode berikut menunjukkan cara menggunakan `modify-spot-fleet-request`.

## AWS CLI

Untuk mengubah permintaan armada Spot

Perintah contoh ini memperbarui kapasitas target permintaan armada Spot yang ditentukan.

Perintah:

```
aws ec2 modify-spot-fleet-request --target-capacity 20 --spot-fleet-request-id
sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
  "Return": true
}
```

Perintah contoh ini mengurangi kapasitas target permintaan armada Spot yang ditentukan tanpa menghentikan Instans Spot apa pun sebagai hasilnya.

Perintah:

```
aws ec2 modify-spot-fleet-request --target-capacity 10 --excess-capacity-
termination-policy NoTermination --spot-fleet-request-ids sfr-73fbd2ce-
aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [ModifySpotFleetRequest](#) di Referensi AWS CLI Perintah.

## **modify-subnet-attribute**

Contoh kode berikut menunjukkan cara menggunakan `modify-subnet-attribute`.

## AWS CLI

Untuk mengubah perilaku pengalamatan IPv4 publik subnet

Contoh ini memodifikasi subnet-1a2b3c4d untuk menentukan bahwa semua instance yang diluncurkan ke subnet ini diberi alamat IPv4 publik. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --map-public-ip-on-launch
```

Untuk mengubah perilaku pengalamatan IPv6 subnet

Contoh ini memodifikasi subnet-1a2b3c4d untuk menentukan bahwa semua instance yang diluncurkan ke subnet ini diberi alamat IPv6 dari kisaran subnet.

Perintah:

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --assign-ipv6-address-on-creation
```

Untuk informasi selengkapnya, lihat Alamat IP di VPC Anda di Panduan Pengguna Virtual Private Cloud.AWS

- Untuk detail API, lihat [ModifySubnetAttribute](#) di Referensi AWS CLI Perintah.

## **modify-traffic-mirror-filter-network-services**

Contoh kode berikut menunjukkan cara menggunakan `modify-traffic-mirror-filter-network-services`.

AWS CLI

Untuk menambahkan layanan jaringan ke filter Traffic Mirror

`modify-traffic-mirror-filter-network-services` Contoh berikut menambahkan layanan jaringan DNS Amazon ke filter yang ditentukan.

```
aws ec2 modify-traffic-mirror-filter-network-services \
  --traffic-mirror-filter-id tmf-04812ff784EXAMPLE \
  --add-network-service amazon-dns
```

**Output:**

```
{
  "TrafficMirrorFilter": {
    "Tags": [
      {
        "Key": "Name",
        "Value": "Production"
      }
    ],
    "EgressFilterRules": [],
    "NetworkServices": [
      "amazon-dns"
    ],
    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
    "IngressFilterRules": [
      {
        "SourceCidrBlock": "0.0.0.0/0",
        "RuleNumber": 1,
        "DestinationCidrBlock": "0.0.0.0/0",
        "Description": "TCP Rule",
        "Protocol": 6,
        "TrafficDirection": "ingress",
        "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
        "RuleAction": "accept",
        "TrafficMirrorFilterRuleId": "tmf-04812ff784EXAMPLE"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Layanan Jaringan Filter Cermin Lalu Lintas](#) di Panduan Pencerminan AWS Lalu Lintas.

- Untuk detail API, lihat [ModifyTrafficMirrorFilterNetworkServices](#) di Referensi AWS CLI Perintah.

**modify-traffic-mirror-filter-rule**

Contoh kode berikut menunjukkan cara menggunakan `modify-traffic-mirror-filter-rule`.

**AWS CLI**

Untuk memodifikasi aturan filter cermin lalu lintas



`modify-traffic-mirror-filter-rule` Contoh berikut memodifikasi deskripsi aturan filter cermin lalu lintas yang ditentukan.

```
aws ec2 modify-traffic-mirror-filter-rule \  
  --traffic-mirror-filter-rule-id tmfr-0ca76e0e08EXAMPLE \  
  --description "TCP Rule"
```

Output:

```
{  
  "TrafficMirrorFilterRule": {  
    "TrafficMirrorFilterRuleId": "tmfr-0ca76e0e08EXAMPLE",  
    "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",  
    "TrafficDirection": "ingress",  
    "RuleNumber": 100,  
    "RuleAction": "accept",  
    "Protocol": 6,  
    "DestinationCidrBlock": "10.0.0.0/24",  
    "SourceCidrBlock": "10.0.0.0/24",  
    "Description": "TCP Rule"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Aturan Filter Cermin Lalu Lintas Anda](#) di Panduan Pencermian AWS Lalu Lintas.

- Untuk detail API, lihat [ModifyTrafficMirrorFilterRule](#) di Referensi AWS CLI Perintah.

## `modify-traffic-mirror-session`

Contoh kode berikut menunjukkan cara menggunakan `modify-traffic-mirror-session`.

### AWS CLI

Untuk memodifikasi sesi cermin lalu lintas

`modify-traffic-mirror-session` Contoh berikut mengubah deskripsi sesi cermin lalu lintas dan jumlah paket yang akan dicerminkan.

```
aws ec2 modify-traffic-mirror-session \  
  --description "Change packet length" \  
  --rule-id tmfr-0ca76e0e08EXAMPLE
```

```
--traffic-mirror-session-id tms-08a33b1214EXAMPLE \
--remove-fields "packet-length"
```

Output:

```
{
  "TrafficMirrorSession": {
    "TrafficMirrorSessionId": "tms-08a33b1214EXAMPLE",
    "TrafficMirrorTargetId": "tmt-07f75d8feeEXAMPLE",
    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
    "NetworkInterfaceId": "eni-070203f901EXAMPLE",
    "OwnerId": "111122223333",
    "SessionNumber": 1,
    "VirtualNetworkId": 7159709,
    "Description": "Change packet length",
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi sesi cermin lalu lintas Anda](#) di Panduan Pencerminalan Lalu Lintas.

- Untuk detail API, lihat [ModifyTrafficMirrorSession](#) di Referensi AWS CLI Perintah.

## modify-transit-gateway-prefix-list-reference

Contoh kode berikut menunjukkan cara menggunakan `modify-transit-gateway-prefix-list-reference`.

AWS CLI

Untuk memodifikasi referensi ke daftar awalan

`modify-transit-gateway-prefix-list-reference` Contoh berikut memodifikasi referensi daftar awalan dalam tabel rute yang ditentukan dengan mengubah lampiran yang lalu lintas dirutekan.

```
aws ec2 modify-transit-gateway-prefix-list-reference \
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
  --prefix-list-id pl-11111122222222333 \
  --transit-gateway-attachment-id tgw-attach-aabbccddaabbccaab
```

**Output:**

```
{
  "TransitGatewayPrefixListReference": {
    "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
    "PrefixListId": "pl-1111112222222333",
    "PrefixListOwnerId": "123456789012",
    "State": "modifying",
    "Blackhole": false,
    "TransitGatewayAttachment": {
      "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
      "ResourceType": "vpc",
      "ResourceId": "vpc-112233445566aabbcc"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Referensi daftar awalan](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [ModifyTransitGatewayPrefixListReferenced](#) di Referensi AWS CLI Perintah.

**modify-transit-gateway-vpc-attachment**

Contoh kode berikut menunjukkan cara menggunakan `modify-transit-gateway-vpc-attachment`.

**AWS CLI**

Untuk memodifikasi lampiran VPC gateway transit

`modify-transit-gateway-vpc-attachment` Contoh berikut menambahkan subnet ke lampiran VPC gateway transit yang ditentukan.

```
aws ec2 modify-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-09fbd47ddfEXAMPLE \
  --add-subnet-ids subnet-0e51f45802EXAMPLE
```

**Output:**

```
{
  "TransitGatewayVpcAttachment": {
```

```

    "TransitGatewayAttachmentId": "tgw-attach-09fbd47ddfEXAMPLE",
    "TransitGatewayId": "tgw-0560315ccfEXAMPLE",
    "VpcId": "vpc-5eccc927",
    "VpcOwnerId": "111122223333",
    "State": "modifying",
    "SubnetIds": [
      "subnet-0e51f45802EXAMPLE",
      "subnet-1EXAMPLE"
    ],
    "CreationTime": "2019-08-08T16:47:38.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Lampiran gateway transit ke VPC](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [ModifyTransitGatewayVpcAttachment](#) di Referensi AWS CLI Perintah.

## modify-transit-gateway

Contoh kode berikut menunjukkan cara menggunakan `modify-transit-gateway`.

### AWS CLI

Untuk memodifikasi gateway transit

`modify-transit-gateway` Contoh berikut memodifikasi gateway transit yang ditentukan dengan mengaktifkan dukungan ECMP untuk lampiran VPN.

```

aws ec2 modify-transit-gateway \
  --transit-gateway-id tgw-111111222222aaaaa \
  --options VpnEcmpSupport=enable

```

Output:

```

{
  "TransitGateway": {
    "TransitGatewayId": "tgw-111111222222aaaaa",

```

```

    "TransitGatewayArn": "64512",
    "State": "modifying",
    "OwnerId": "123456789012",
    "CreationTime": "2020-04-30T08:41:37.000Z",
    "Options": {
      "AmazonSideAsn": 64512,
      "AutoAcceptSharedAttachments": "disable",
      "DefaultRouteTableAssociation": "enable",
      "AssociationDefaultRouteTableId": "tgw-rtb-0123456789abcd123",
      "DefaultRouteTablePropagation": "enable",
      "PropagationDefaultRouteTableId": "tgw-rtb-0123456789abcd123",
      "VpnEcmpSupport": "enable",
      "DnsSupport": "enable"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Gateway transit di Panduan Gerbang Transit](#).

- Untuk detail API, lihat [ModifyTransitGateway](#) di Referensi AWS CLI Perintah.

## modify-verified-access-endpoint-policy

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-endpoint-policy`.

### AWS CLI

Untuk mengonfigurasi kebijakan Akses Terverifikasi untuk titik akhir

`modify-verified-access-endpoint-policy` Contoh berikut menambahkan kebijakan Akses Terverifikasi yang ditentukan ke titik akhir Akses Terverifikasi yang ditentukan.

```

aws ec2 modify-verified-access-endpoint-policy \
  --verified-access-endpoint-id vae-066fac616d4d546f2 \
  --policy-enabled \
  --policy-document file://policy.txt

```

Isi dari `policy.txt`:

```

permit(principal, action, resource)
when {

```

```
context.identity.groups.contains("finance") &&
context.identity.email.verified == true
};
```

Output:

```
{
  "PolicyEnabled": true,
  "PolicyDocument": "permit(principal,action,resource)\nwhen
{\n  context.identity.groups.contains(\"finance\") &&\n
context.identity.email_verified == true\n};"
}
```

Untuk informasi selengkapnya, lihat [kebijakan Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessEndpointPolicy](#) di Referensi AWS CLI Perintah.

## modify-verified-access-endpoint

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-endpoint`.

### AWS CLI

Untuk mengubah konfigurasi titik akhir Akses Terverifikasi

`modify-verified-access-endpoint` Contoh berikut menambahkan deskripsi yang ditentukan ke titik akhir Akses Terverifikasi yang ditentukan.

```
aws ec2 modify-verified-access-endpoint \
  --verified-access-endpoint-id vae-066fac616d4d546f2 \
  --description "Testing Verified Access"
```

Output:

```
{
  "VerifiedAccessEndpoint": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
    "ApplicationDomain": "example.com",
```

```

    "EndpointType": "network-interface",
    "AttachmentType": "vpc",
    "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
    "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
    "SecurityGroupIds": [
        "sg-004915970c4c8f13a"
    ],
    "NetworkInterfaceOptions": {
        "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
        "Protocol": "https",
        "Port": 443
    },
    "Status": {
        "Code": "updating"
    },
    "Description": "Testing Verified Access",
    "CreationTime": "2023-08-25T20:54:43",
    "LastUpdatedTime": "2023-08-25T22:46:32"
}
}

```

Untuk informasi selengkapnya, lihat [Titik akhir Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessEndpoint](#) di Referensi AWS CLI Perintah.

## modify-verified-access-group-policy

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-group-policy`.

### AWS CLI

Untuk mengonfigurasi kebijakan Akses Terverifikasi untuk grup

`modify-verified-access-group-policy` Contoh berikut menambahkan kebijakan Akses Terverifikasi yang ditentukan ke grup Akses Terverifikasi yang ditentukan.

```
aws ec2 modify-verified-access-group-policy \
```

```
--verified-access-group-id vagr-0dbe967baf14b7235 \  
--policy-enabled \  
--policy-document file://policy.txt
```

Isi dari policy.txt:

```
permit(principal,action,resource)  
when {  
    context.identity.groups.contains("finance") &&  
    context.identity.email.verified == true  
};
```

Output:

```
{  
  "PolicyEnabled": true,  
  "PolicyDocument": "permit(principal,action,resource)\nwhen  
{\n  context.identity.groups.contains(\"finance\") &&\n  context.identity.email_verified == true\n};"  
}
```

Untuk informasi selengkapnya, lihat [Grup Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessGroupPolicy](#) di Referensi AWS CLI Perintah.

## modify-verified-access-group

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-group`.

### AWS CLI

Untuk mengubah konfigurasi grup Akses Terverifikasi

`modify-verified-access-group` Contoh berikut menambahkan deskripsi yang ditentukan ke grup Akses Terverifikasi yang ditentukan.

```
aws ec2 modify-verified-access-group \  
--verified-access-group-id vagr-0dbe967baf14b7235 \  
--description "Testing Verified Access"
```



**Output:**

```
{
  "VerifiedAccessGroup": {
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "Owner": "123456789012",
    "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
    "CreationTime": "2023-08-25T19:55:19",
    "LastUpdatedTime": "2023-08-25T22:17:25"
  }
}
```

Untuk informasi selengkapnya, lihat [Grup Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessGroup](#) di Referensi AWS CLI Perintah.

**modify-verified-access-instance-logging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-instance-logging-configuration`.

**AWS CLI**

Untuk mengaktifkan pencatatan untuk instance Akses Terverifikasi

`modify-verified-access-instance-logging-configuration` Contoh berikut memungkinkan pencatatan akses untuk instance Akses Terverifikasi yang ditentukan. Log akan dikirimkan ke grup CloudWatch log Log yang ditentukan.

```
aws ec2 modify-verified-access-instance-logging-configuration \
  --verified-access-instance-id vai-0ce000c0b7643abea \
  --access-logs CloudWatchLogs={Enabled=true,LogGroup=my-log-group}
```

**Output:**

```
{
  "LoggingConfiguration": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
```

```

    "AccessLogs": {
      "S3": {
        "Enabled": false
      },
      "CloudWatchLogs": {
        "Enabled": true,
        "DeliveryStatus": {
          "Code": "success"
        },
        "LogGroup": "my-log-group"
      },
      "KinesisDataFirehose": {
        "Enabled": false
      },
      "LogVersion": "ocsf-1.0.0-rc.2",
      "IncludeTrustContext": false
    }
  }
}

```

Untuk informasi selengkapnya, lihat [log Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessInstanceLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## modify-verified-access-instance

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-instance`.

### AWS CLI

Untuk mengubah konfigurasi instance Akses Terverifikasi

`modify-verified-access-instance` Contoh berikut menambahkan deskripsi yang ditentukan ke instance Akses Terverifikasi yang ditentukan.

```

aws ec2 modify-verified-access-instance \
  --verified-access-instance-id vai-0ce000c0b7643abea \
  --description "Testing Verified Access"

```

Output:

```
{
  "VerifiedAccessInstance": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "VerifiedAccessTrustProviders": [
      {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center"
      }
    ],
    "CreationTime": "2023-08-25T18:27:56",
    "LastUpdatedTime": "2023-08-25T22:41:04"
  }
}
```

Untuk informasi selengkapnya, lihat [Instans Akses Terverifikasi](#) di Panduan Pengguna Akses AWS Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessInstance](#) di Referensi AWS CLI Perintah.

## modify-verified-access-trust-provider

Contoh kode berikut menunjukkan cara menggunakan `modify-verified-access-trust-provider`.

### AWS CLI

Untuk mengubah konfigurasi penyedia kepercayaan Akses Terverifikasi

`modify-verified-access-trust-provider` Contoh berikut menambahkan deskripsi yang ditentukan ke penyedia kepercayaan Akses Terverifikasi yang ditentukan.

```
aws ec2 modify-verified-access-trust-provider \
  --verified-access-trust-provider-id vatp-0bb32de759a3e19e7 \
  --description "Testing Verified Access"
```

Output:

```
{
  "VerifiedAccessTrustProvider": {
```

```
"VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
"Description": "Testing Verified Access",
"TrustProviderType": "user",
"UserTrustProviderType": "iam-identity-center",
"PolicyReferenceName": "idc",
"CreationTime": "2023-08-25T19:00:38",
"LastUpdatedTime": "2023-08-25T19:18:21"
}
}
```

Untuk informasi selengkapnya, lihat [Penyedia kepercayaan untuk Akses AWS Terverifikasi](#) di Panduan Pengguna Akses Terverifikasi.

- Untuk detail API, lihat [ModifyVerifiedAccessTrustProvider](#) di Referensi AWS CLI Perintah.

## modify-volume-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-volume-attribute`.

### AWS CLI

Untuk memodifikasi atribut volume

Contoh ini menetapkan `autoEnableIo` atribut volume dengan ID `vol-1234567890abcdef0` ke `true`. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 modify-volume-attribute --volume-id vol-1234567890abcdef0 --auto-enable-io
```

- Untuk detail API, lihat [ModifyVolumeAttribute](#) di Referensi AWS CLI Perintah.

## modify-volume

Contoh kode berikut menunjukkan cara menggunakan `modify-volume`.

### AWS CLI

Contoh 1: Untuk memodifikasi volume dengan mengubah ukurannya

`modify-volume` Contoh berikut mengubah ukuran volume yang ditentukan menjadi 150GB.

**Perintah:**

```
aws ec2 modify-volume --size 150 --volume-id vol-1234567890abcdef0
```

**Output:**

```
{
  "VolumeModification": {
    "TargetSize": 150,
    "TargetVolumeType": "io1",
    "ModificationState": "modifying",
    "VolumeId": " vol-1234567890abcdef0",
    "TargetIops": 100,
    "StartTime": "2019-05-17T11:27:19.000Z",
    "Progress": 0,
    "OriginalVolumeType": "io1",
    "OriginalIops": 100,
    "OriginalSize": 100
  }
}
```

Contoh 2: Untuk memodifikasi volume dengan mengubah jenis, ukuran, dan nilai IOPS

`modify-volume` Contoh berikut mengubah jenis volume menjadi Provisioned IOPS SSD, menetapkan target IOPS rate menjadi 10000, dan menetapkan ukuran volume menjadi 350GB.

```
aws ec2 modify-volume \
  --volume-type io1 \
  --iops 10000 \
  --size 350 \
  --volume-id vol-1234567890abcdef0
```

**Output:**

```
{
  "VolumeModification": {
    "TargetSize": 350,
    "TargetVolumeType": "io1",
    "ModificationState": "modifying",
    "VolumeId": "vol-0721c1a9d08c93bf6",
    "TargetIops": 10000,
  }
}
```

```
    "StartTime": "2019-05-17T11:38:57.000Z",
    "Progress": 0,
    "OriginalVolumeType": "gp2",
    "OriginalIops": 150,
    "OriginalSize": 50
  }
}
```

- Untuk detail API, lihat [ModifyVolume](#) di Referensi AWS CLI Perintah.

## modify-vpc-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-attribute`.

### AWS CLI

Untuk memodifikasi `enableDnsSupport` atribut

Contoh ini memodifikasi `enableDnsSupport` atribut. Atribut ini menunjukkan apakah resolusi DNS diaktifkan untuk VPC. Jika atribut `init` `true`, server DNS Amazon menyelesaikan nama host DNS untuk instance Anda ke alamat IP yang sesuai; jika tidak, tidak. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 modify-vpc-attribute --vpc-id vpc-a01106c2 --enable-dns-support "{\"Value\":false}"
```

Untuk memodifikasi `enableDnsHostnames` atribut

Contoh ini memodifikasi `enableDnsHostnames` atribut. Atribut ini menunjukkan apakah instance yang diluncurkan di VPC mendapatkan nama host DNS. Jika atribut `init` `true`, instance di VPC mendapatkan nama host DNS; jika tidak, mereka tidak. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 modify-vpc-attribute --vpc-id vpc-a01106c2 --enable-dns-hostnames "{\"Value\":false}"
```

- Untuk detail API, lihat [ModifyVpcAttribute](#) di Referensi AWS CLI Perintah.

## modify-vpc-endpoint-connection-notification

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-endpoint-connection-notification`.

### AWS CLI

Untuk memodifikasi pemberitahuan koneksi titik akhir

Contoh ini mengubah topik SNS untuk pemberitahuan koneksi titik akhir yang ditentukan.

Perintah:

```
aws ec2 modify-vpc-endpoint-connection-notification --connection-notification-id vpce-nfn-008776de7e03f5abc --connection-events Accept Reject --connection-notification-arn arn:aws:sns:us-east-2:123456789012:mytopic
```

Output:

```
{
  "ReturnValue": true
}
```

- Untuk detail API, lihat [ModifyVpcEndpointConnectionNotification](#) di Referensi AWS CLI Perintah.

## modify-vpc-endpoint-service-configuration

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-endpoint-service-configuration`.

### AWS CLI

Untuk memodifikasi konfigurasi layanan endpoint

Contoh ini mengubah persyaratan penerimaan untuk layanan endpoint yang ditentukan.

Perintah:

```
aws ec2 modify-vpc-endpoint-service-configuration --service-id vpce-svc-09222513e6e77dc86 --no-acceptance-required
```

**Output:**

```
{
  "ReturnValue": true
}
```

- Untuk detail API, lihat [ModifyVpcEndpointServiceConfiguration](#) di Referensi AWS CLI Perintah.

**modify-vpc-endpoint-service-payer-responsibility**

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-endpoint-service-payer-responsibility`.

**AWS CLI**

Untuk memodifikasi tanggung jawab pembayar

`modify-vpc-endpoint-service-payer-responsibility` Contoh berikut memodifikasi tanggung jawab pembayar dari layanan endpoint yang ditentukan.

```
aws ec2 modify-vpc-endpoint-service-payer-responsibility \
  --service-id vpce-svc-071afff70666e61e0 \
  --payer-responsibility ServiceOwner
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [ModifyVpcEndpointServicePayerResponsibility](#) di Referensi AWS CLI Perintah.

**modify-vpc-endpoint-service-permissions**

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-endpoint-service-permissions`.

**AWS CLI**

Untuk memodifikasi izin layanan titik akhir

Contoh ini menambahkan izin untuk AWS akun untuk terhubung ke layanan endpoint yang ditentukan.



**Perintah:**

```
aws ec2 modify-vpc-endpoint-service-permissions --service-id vpce-  
svc-03d5ebb7d9579a2b3 --add-allowed-principals '["arn:aws:iam::123456789012:root"]'
```

**Output:**

```
{  
  "ReturnValue": true  
}
```

Contoh ini menambahkan izin untuk pengguna IAM tertentu (admin) untuk terhubung ke layanan titik akhir yang ditentukan.

**Perintah:**

```
aws ec2 modify-vpc-endpoint-service-permissions --service-id vpce-  
svc-03d5ebb7d9579a2b3 --add-allowed-principals '["arn:aws:iam::123456789012:user/  
admin"]'
```

- Untuk detail API, lihat [ModifyVpcEndpointServicePermissions](#) di Referensi AWS CLI Perintah.

**modify-vpc-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-endpoint`.

**AWS CLI**

Untuk memodifikasi titik akhir gateway

Contoh ini memodifikasi titik akhir gateway `vpce-1a2b3c4d` dengan mengaitkan tabel rute `rtb-aaa222bb` dengan titik akhir, dan mengatur ulang dokumen kebijakan.

**Perintah:**

```
aws ec2 modify-vpc-endpoint --vpc-endpoint-id vpce-1a2b3c4d --add-route-table-ids  
rtb-aaa222bb --reset-policy
```

**Output:**

```
{
  "Return": true
}
```

Untuk memodifikasi titik akhir antarmuka

Contoh ini memodifikasi titik akhir antarmuka `vpce-0fe5b17a0707d6fa5` dengan menambahkan subnet `subnet-d6fcaa8d` ke titik akhir.

Perintah:

```
aws ec2 modify-vpc-endpoint --vpc-endpoint-id vpce-0fe5b17a0707d6fa5 --add-subnet-id
subnet-d6fcaa8d
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [ModifyVpcEndpoint](#) di Referensi AWS CLI Perintah.

## **modify-vpc-peering-connection-options**

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-peering-connection-options`.

AWS CLI

Untuk mengaktifkan komunikasi melalui koneksi peering VPC dari koneksi lokal Anda ClassicLink

Dalam contoh ini, untuk koneksi peering `pcx-aaaabbbb`, pemilik VPC pemohon memodifikasi opsi koneksi peering VPC untuk mengaktifkan koneksi lokal untuk berkomunikasi dengan VPC rekan. ClassicLink

Perintah:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-
connection-id pcx-aaaabbbb --requester-peering-connection-options
AllowEgressFromLocalClassicLinkToRemoteVpc=true
```

**Output:**

```
{
  "RequesterPeeringConnectionOptions": {
    "AllowEgressFromLocalClassicLinkToRemoteVpc": true
  }
}
```

Untuk mengaktifkan komunikasi melalui koneksi peering VPC dari VPC lokal Anda ke koneksi jarak jauh ClassicLink

Dalam contoh ini, pemilik VPC penerima memodifikasi opsi koneksi peering VPC untuk memungkinkan VPC lokal berkomunikasi dengan koneksi di VPC rekan. ClassicLink

**Perintah:**

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-connection-id pcx-aaaabbbb --accepter-peering-connection-options AllowEgressFromLocalVpcToRemoteClassicLink=true
```

**Output:**

```
{
  "AcceptorPeeringConnectionOptions": {
    "AllowEgressFromLocalVpcToRemoteClassicLink": true
  }
}
```

Untuk mengaktifkan dukungan resolusi DNS untuk koneksi peering VPC

Dalam contoh ini, pemilik VPC pemohon memodifikasi opsi koneksi peering VPC untuk mengaktifkan VPC lokal pcx-aaaabbbb untuk menyelesaikan nama host DNS publik ke alamat IP pribadi saat ditanyakan dari instance di VPC rekan.

**Perintah:**

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-connection-id pcx-aaaabbbb --requester-peering-connection-options AllowDnsResolutionFromRemoteVpc=true
```

**Output:**

```
{
  "RequesterPeeringConnectionOptions": {
    "AllowDnsResolutionFromRemoteVpc": true
  }
}
```

- Untuk detail API, lihat [ModifyVpcPeeringConnectionOptions](#) di Referensi AWS CLI Perintah.

## modify-vpc-tenancy

Contoh kode berikut menunjukkan cara menggunakan `modify-vpc-tenancy`.

### AWS CLI

Untuk memodifikasi sewa VPC

Contoh ini memodifikasi penyewaan VPC ke. `vpc-1a2b3c4d` default

Perintah:

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [ModifyVpcTenancy](#) di Referensi AWS CLI Perintah.

## modify-vpn-connection-options

Contoh kode berikut menunjukkan cara menggunakan `modify-vpn-connection-options`.

### AWS CLI

Untuk memodifikasi opsi koneksi VPN Anda

`modify-vpn-connection-options` Contoh berikut memodifikasi IPv4 CIDR lokal di sisi gateway pelanggan dari koneksi VPN yang ditentukan.

```
aws ec2 modify-vpn-connection-options \  
--vpn-connection-id vpn-1122334455aabbccd \  
--local-ipv4-network-cidr 10.0.0.0/16
```

**Output:**

```
{  
  "VpnConnections": [  
    {  
      "CustomerGatewayConfiguration": "...configuration information...",  
      "CustomerGatewayId": "cgw-01234567abcde1234",  
      "Category": "VPN",  
      "State": "modifying",  
      "Type": "ipsec.1",  
      "VpnConnectionId": "vpn-1122334455aabbccd",  
      "TransitGatewayId": "tgw-00112233445566aab",  
      "Options": {  
        "EnableAcceleration": false,  
        "StaticRoutesOnly": true,  
        "LocalIpv4NetworkCidr": "10.0.0.0/16",  
        "RemoteIpv4NetworkCidr": "0.0.0.0/0",  
        "TunnelInsideIpVersion": "ipv4"  
      },  
      "Routes": [],  
      "Tags": [  
        {  
          "Key": "Name",  
          "Value": "CanadaVPN"  
        }  
      ],  
      "VgwTelemetry": [  
        {  
          "AcceptedRouteCount": 0,  
          "LastStatusChange": "2020-07-29T10:35:11.000Z",  
          "OutsideIpAddress": "203.0.113.3",  
          "Status": "DOWN",  
          "StatusMessage": ""  
        },  
        {  
          "AcceptedRouteCount": 0,  
          "LastStatusChange": "2020-09-02T09:09:33.000Z",  
          "OutsideIpAddress": "203.0.113.5",  
          "Status": "UP",  
        }  
      ]  
    }  
  ]  
}
```

```

    "StatusMessage": ""
  }
]
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi opsi koneksi VPN Site-to-Site di Panduan Pengguna VPN Site-to-Site](#).AWS

- Untuk detail API, lihat [ModifyVpnConnectionOptions](#) di Referensi AWS CLI Perintah.

## modify-vpn-connection

Contoh kode berikut menunjukkan cara menggunakan `modify-vpn-connection`.

### AWS CLI

Untuk memodifikasi koneksi VPN

`modify-vpn-connection` Contoh berikut mengubah gateway target untuk koneksi VPN `vpn-12345678901234567` ke gateway pribadi `virtualvgw-11223344556677889`:

```

aws ec2 modify-vpn-connection \
  --vpn-connection-id vpn-12345678901234567 \
  --vpn-gateway-id vgw-11223344556677889

```

Output:

```

{
  "VpnConnection": {
    "CustomerGatewayConfiguration": "...configuration information...",
    "CustomerGatewayId": "cgw-aabbccdde1122334",
    "Category": "VPN",
    "State": "modifying",
    "Type": "ipsec.1",
    "VpnConnectionId": "vpn-12345678901234567",
    "VpnGatewayId": "vgw-11223344556677889",
    "Options": {
      "StaticRoutesOnly": false
    },
    "VgwTelemetry": [

```

```

    {
      "AcceptedRouteCount": 0,
      "LastStatusChange": "2019-07-17T07:34:00.000Z",
      "OutsideIpAddress": "18.210.3.222",
      "Status": "DOWN",
      "StatusMessage": "IPSEC IS DOWN"
    },
    {
      "AcceptedRouteCount": 0,
      "LastStatusChange": "2019-07-20T21:20:16.000Z",
      "OutsideIpAddress": "34.193.129.33",
      "Status": "DOWN",
      "StatusMessage": "IPSEC IS DOWN"
    }
  ]
}

```

- Untuk detail API, lihat [ModifyVpnConnection](#) di Referensi AWS CLI Perintah.

## modify-vpn-tunnel-certificate

Contoh kode berikut menunjukkan cara menggunakan `modify-vpn-tunnel-certificate`.

### AWS CLI

Untuk memutar sertifikat terowongan VPN

`modify-vpn-tunnel-certificate` Contoh berikut memutar sertifikat untuk terowongan yang ditentukan untuk koneksi VPN

```

aws ec2 modify-vpn-tunnel-certificate \
  --vpn-tunnel-outside-ip-address 203.0.113.17 \
  --vpn-connection-id vpn-12345678901234567

```

Output:

```

{
  "VpnConnection": {
    "CustomerGatewayConfiguration": "...configuration information...",
    "CustomerGatewayId": "cgw-aabbccdee1122334",

```

```

    "Category": "VPN",
    "State": "modifying",
    "Type": "ipsec.1",
    "VpnConnectionId": "vpn-12345678901234567",
    "VpnGatewayId": "vgw-11223344556677889",
    "Options": {
      "StaticRoutesOnly": false
    },
    "VgwTelemetry": [
      {
        "AcceptedRouteCount": 0,
        "LastStatusChange": "2019-09-11T17:27:14.000Z",
        "OutsideIpAddress": "203.0.113.17",
        "Status": "DOWN",
        "StatusMessage": "IPSEC IS DOWN",
        "CertificateArn": "arn:aws:acm:us-east-1:123456789101:certificate/c544d8ce-20b8-4fff-98b0-example"
      },
      {
        "AcceptedRouteCount": 0,
        "LastStatusChange": "2019-09-11T17:26:47.000Z",
        "OutsideIpAddress": "203.0.114.18",
        "Status": "DOWN",
        "StatusMessage": "IPSEC IS DOWN",
        "CertificateArn": "arn:aws:acm:us-east-1:123456789101:certificate/5ab64566-761b-4ad3-b259-example"
      }
    ]
  }
}

```

- Untuk detail API, lihat [ModifyVpnTunnelCertificate](#) di Referensi AWS CLI Perintah.

## modify-vpn-tunnel-options

Contoh kode berikut menunjukkan cara menggunakan `modify-vpn-tunnel-options`.

### AWS CLI

Untuk memodifikasi opsi terowongan untuk koneksi VPN

`modify-vpn-tunnel-options` Contoh berikut memperbarui grup Diffie-Hellman yang diizinkan untuk terowongan dan koneksi VPN yang ditentukan.



```
aws ec2 modify-vpn-tunnel-options \  
  --vpn-connection-id vpn-12345678901234567 \  
  --vpn-tunnel-outside-ip-address 203.0.113.17 \  
  --tunnel-options Phase1DHGroupNumbers=[{Value=14},{Value=15},{Value=16},  
{Value=17},{Value=18}],Phase2DHGroupNumbers=[{Value=14},{Value=15},{Value=16},  
{Value=17},{Value=18}]
```

### Output:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "...configuration information...",  
    "CustomerGatewayId": "cgw-aabbccdde1122334",  
    "Category": "VPN",  
    "State": "available",  
    "Type": "ipsec.1",  
    "VpnConnectionId": "vpn-12345678901234567",  
    "VpnGatewayId": "vgw-11223344556677889",  
    "Options": {  
      "StaticRoutesOnly": false,  
      "TunnelOptions": [  
        {  
          "OutsideIpAddress": "203.0.113.17",  
          "Phase1DHGroupNumbers": [  
            {  
              "Value": 14  
            },  
            {  
              "Value": 15  
            },  
            {  
              "Value": 16  
            },  
            {  
              "Value": 17  
            },  
            {  
              "Value": 18  
            }  
          ],  
          "Phase2DHGroupNumbers": [  
            {  
              "Value": 14  
            }  
          ]  
        }  
      ]  
    }  
  }  
}
```

```
    },
    {
      "Value": 15
    },
    {
      "Value": 16
    },
    {
      "Value": 17
    },
    {
      "Value": 18
    }
  ]
},
{
  "OutsideIpAddress": "203.0.114.19"
}
]
},
"VgwTelemetry": [
  {
    "AcceptedRouteCount": 0,
    "LastStatusChange": "2019-09-10T21:56:54.000Z",
    "OutsideIpAddress": "203.0.113.17",
    "Status": "DOWN",
    "StatusMessage": "IPSEC IS DOWN"
  },
  {
    "AcceptedRouteCount": 0,
    "LastStatusChange": "2019-09-10T21:56:43.000Z",
    "OutsideIpAddress": "203.0.114.19",
    "Status": "DOWN",
    "StatusMessage": "IPSEC IS DOWN"
  }
]
}
}
```

- Untuk detail API, lihat [ModifyVpnTunnelOptions](#) di Referensi AWS CLI Perintah.

## monitor-instances

Contoh kode berikut menunjukkan cara menggunakan `monitor-instances`.

### AWS CLI

Untuk mengaktifkan pemantauan yang mendetail untuk instans

Perintah contoh ini mengaktifkan pemantauan yang mendetail untuk instans tertentu.

Perintah:

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
  "InstanceMonitorings": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "Monitoring": {
        "State": "pending"
      }
    }
  ]
}
```

- Untuk detail API, lihat [MonitorInstances](#) di Referensi AWS CLI Perintah.

## move-address-to-vpc

Contoh kode berikut menunjukkan cara menggunakan `move-address-to-vpc`.

### AWS CLI

Untuk memindahkan alamat ke EC2-VPC

Contoh ini memindahkan alamat IP Elastis 54.123.4.56 ke platform EC2-VPC.

Perintah:

```
aws ec2 move-address-to-vpc --public-ip 54.123.4.56
```

**Output:**

```
{
  "Status": "MoveInProgress"
}
```

- Untuk detail API, lihat [MoveAddressToVpc](#) Referensi AWS CLI Perintah.

**move-byoip-cidr-to-ipam**

Contoh kode berikut menunjukkan cara menggunakan `move-byoip-cidr-to-ipam`.

**AWS CLI**

Untuk mentransfer BYOIP CIDR ke IPAM

`move-byoip-cidr-to-ipam` Contoh berikut mentransfer BYOIP CIDR ke IPAM.

(Linux):

```
aws ec2 move-byoip-cidr-to-ipam \
  --region us-west-2 \
  --ipam-pool-id ipam-pool-0a03d430ca3f5c035 \
  --ipam-pool-owner 111111111111 \
  --cidr 130.137.249.0/24
```

(Windows):

```
aws ec2 move-byoip-cidr-to-ipam ^
  --region us-west-2 ^
  --ipam-pool-id ipam-pool-0a03d430ca3f5c035 ^
  --ipam-pool-owner 111111111111 ^
  --cidr 130.137.249.0/24
```

**Output:**

```
{
  "ByoipCidr": {
    "Cidr": "130.137.249.0/24",
    "State": "pending-transfer"
  }
}
```

Untuk informasi selengkapnya, lihat [Tutorial: Mentransfer CIDR BYOIP IPv4 yang ada ke IPAM di Panduan Pengguna](#) Amazon VPC IPAM.

- Untuk detail API, lihat [MoveByoipCidrToIpam](#) di Referensi AWS CLI Perintah.

## network-insights-access-scope

Contoh kode berikut menunjukkan cara menggunakan `network-insights-access-scope`.

### AWS CLI

Untuk membuat cakupan akses Network Insights

`create-network-insights-access-scope` Contoh berikut membuat cakupan akses wawasan jaringan di AWS akun Anda.

```
aws ec2 create-network-insights-access-scope \  
  --cli-input-json file://access-scope-file.json
```

Isi dari `access-scope-file.json`:

```
{  
  {  
    "MatchPaths": [  
      {  
        "Source": {  
          "ResourceStatement": {  
            "Resources": [  
              "vpc-abcd12e3"  
            ]  
          }  
        }  
      ],  
      "ExcludePaths": [  
        {  
          "Source": {  
            "ResourceStatement": {  
              "ResourceTypes": [  
                "AWS::EC2::InternetGateway"  
              ]  
            }  
          }  
        }  
      ]  
    }  
  }  
}
```

```

    }
  ]
}

```

### Output:

```

{
  "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789111"
}{
  "NetworkInsightsAccessScope": {
    "NetworkInsightsAccessScopeId": "nis-123456789222",
    "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-1:123456789222:network-insights-access-scope/nis-123456789222",
    "CreateDate": "2022-01-25T19:20:28.796000+00:00",
    "UpdatedDate": "2022-01-25T19:20:28.797000+00:00"
  },
  "NetworkInsightsAccessScopeContent": {
    "NetworkInsightsAccessScopeId": "nis-04c0c0fbca737c404",
    "MatchPaths": [
      {
        "Source": {
          "ResourceStatement": {
            "Resources": [
              "vpc-abcd12e3"
            ]
          }
        }
      }
    ],
    "ExcludePaths": [
      {
        "Source": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::InternetGateway"
            ]
          }
        }
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [NetworkInsightsAccessScoped](#) di Referensi AWS CLI Perintah.

## **provision-byoip-cidr**

Contoh kode berikut menunjukkan cara menggunakan `provision-byoip-cidr`.

### AWS CLI

Untuk menyediakan rentang alamat

`provision-byoip-cidr` Contoh berikut menyediakan rentang alamat IP publik untuk digunakan dengan AWS.

```
aws ec2 provision-byoip-cidr \  
  --cidr 203.0.113.25/24 \  
  --cidr-authorization-context Message="$text_message",Signature="$signed_message"
```

Output:

```
{  
  "ByoipCidr": {  
    "Cidr": "203.0.113.25/24",  
    "State": "pending-provision"  
  }  
}
```

Untuk informasi selengkapnya tentang membuat string pesan untuk konteks otorisasi, lihat [Membawa Alamat IP Anda Sendiri](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [ProvisionByoipCidr](#) di Referensi AWS CLI Perintah.

## **provision-ipam-pool-cidr**

Contoh kode berikut menunjukkan cara menggunakan `provision-ipam-pool-cidr`.

### AWS CLI

Untuk menyediakan CIDR ke kolam IPAM

`provision-ipam-pool-cidr` Contoh berikut menyediakan CIDR ke kolam IPAM.

(Linux):

```
aws ec2 provision-ipam-pool-cidr \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --cidr 10.0.0.0/24
```

(Windows):

```
aws ec2 provision-ipam-pool-cidr ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --cidr 10.0.0.0/24
```

Output:

```
{  
  "IpamPoolCidr": {  
    "Cidr": "10.0.0.0/24",  
    "State": "pending-provision"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menyediakan CIDR ke kumpulan di Panduan Pengguna Amazon VPC IPAM](#).

- Untuk detail API, lihat [ProvisionIpamPoolCidr](#) di Referensi AWS CLI Perintah.

## **purchase-host-reservation**

Contoh kode berikut menunjukkan cara menggunakan `purchase-host-reservation`.

AWS CLI

Untuk membeli Reservasi Tuan Rumah Khusus

Contoh ini membeli penawaran Reservasi Host Khusus yang ditentukan untuk Host Khusus yang ditentukan di akun Anda.

Perintah:



```
aws ec2 purchase-host-reservation --offering-id hro-03f707bf363b6b324 --host-id-set h-013abcd2a00cbd123
```

Output:

```
{
  "TotalHourlyPrice": "1.499",
  "Purchase": [
    {
      "HourlyPrice": "1.499",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "HostIdSet": [
        "h-013abcd2a00cbd123"
      ],
      "HostReservationId": "hr-0d418a3a4ffc669ae",
      "UpfrontPrice": "0.000",
      "Duration": 31536000
    }
  ],
  "TotalUpfrontPrice": "0.000"
}
```

- Untuk detail API, lihat [PurchaseHostReservation](#) di Referensi AWS CLI Perintah.

## **purchase-reserved-instances-offering**

Contoh kode berikut menunjukkan cara menggunakan `purchase-reserved-instances-offering`.

### AWS CLI

Untuk membeli penawaran Instans Cadangan

Perintah contoh ini menggambarkan pembelian penawaran Instans Cadangan, yang menentukan ID penawaran dan jumlah instans.

Perintah:

```
aws ec2 purchase-reserved-instances-offering --reserved-instances-offering-id ec06327e-dd07-46ee-9398-75b5fexample --instance-count 3
```

**Output:**

```
{
  "ReservedInstancesId": "af9f760e-6f91-4559-85f7-4980eexample"
}
```

- Untuk detail API, lihat [PurchaseReservedInstancesOffering](#) di Referensi AWS CLI Perintah.

**purchase-scheduled-instances**

Contoh kode berikut menunjukkan cara menggunakan `purchase-scheduled-instances`.

**AWS CLI**

Untuk membeli Instance Terjadwal

Contoh ini membeli Instance Terjadwal.

**Perintah:**

```
aws ec2 purchase-scheduled-instances --purchase-requests file://purchase-
request.json
```

**Permintaan pembelian.json:**

```
[
  {
    "PurchaseToken": "eyJ2IjoiMSIsInMiOjEsImMiOi...",
    "InstanceCount": 1
  }
]
```

**Output:**

```
{
  "ScheduledInstanceSet": [
    {
      "AvailabilityZone": "us-west-2b",
      "ScheduledInstanceId": "sci-1234-1234-1234-1234-123456789012",
      "HourlyPrice": "0.095",
      "CreateDate": "2016-01-25T21:43:38.612Z",
    }
  ]
}
```

```

    "Recurrence": {
      "OccurrenceDaySet": [
        1
      ],
      "Interval": 1,
      "Frequency": "Weekly",
      "OccurrenceRelativeToEnd": false,
      "OccurrenceUnit": ""
    },
    "Platform": "Linux/UNIX",
    "TermEndDate": "2017-01-31T09:00:00Z",
    "InstanceCount": 1,
    "SlotDurationInHours": 32,
    "TermStartDate": "2016-01-31T09:00:00Z",
    "NetworkPlatform": "EC2-VPC",
    "TotalScheduledInstanceHours": 1696,
    "NextSlotStartTime": "2016-01-31T09:00:00Z",
    "InstanceType": "c4.large"
  }
]
}

```

- Untuk detail API, lihat [PurchaseScheduledInstances](#) di Referensi AWS CLI Perintah.

## reboot-instances

Contoh kode berikut menunjukkan cara menggunakan `reboot-instances`.

### AWS CLI

Untuk melakukan boot ulang instans Amazon EC2

Contoh ini melakukan boot ulang instans tertentu. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 reboot-instances --instance-ids i-1234567890abcdef5
```

Untuk informasi selengkapnya, lihat [Melakukan Boot Ulang Instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [RebootInstances](#) di Referensi AWS CLI Perintah.

## register-image

Contoh kode berikut menunjukkan cara menggunakan `register-image`.

### AWS CLI

Contoh 1: Untuk mendaftarkan AMI menggunakan file manifes

`register-image` Contoh berikut mendaftarkan AMI menggunakan file manifes yang ditentukan di Amazon S3.

```
aws ec2 register-image \  
  --name my-image \  
  --image-location my-s3-bucket/myimage/image.manifest.xml
```

Output:

```
{  
  "ImageId": "ami-1234567890EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Amazon Machine Image \(AMI\)](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk mendaftarkan AMI menggunakan snapshot perangkat root

`register-image` Contoh berikut mendaftarkan AMI menggunakan snapshot yang ditentukan dari volume root EBS sebagai perangkat. `/dev/xvda` Pemetaan perangkat blok juga menyertakan volume 100 GiB EBS kosong sebagai perangkat. `/dev/xvdf`

```
aws ec2 register-image \  
  --name my-image \  
  --root-device-name /dev/xvda \  
  --block-device-mappings DeviceName=/dev/  
xvda,Ebs={SnapshotId=snap-0db2cf683925d191f} DeviceName=/dev/  
xvdf,Ebs={VolumeSize=100}
```

Output:

```
{  
  "ImageId": "ami-1a2b3c4d5eEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Amazon Machine Image \(AMI\)](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [RegisterImage](#) di Referensi AWS CLI Perintah.

## **register-instance-event-notification-attributes**

Contoh kode berikut menunjukkan cara menggunakan `register-instance-event-notification-attributes`.

### AWS CLI

Contoh 1: Untuk menyertakan semua tag dalam pemberitahuan acara

`register-instance-event-notification-attributes` Contoh berikut mencakup semua tag dalam pemberitahuan acara.

```
aws ec2 register-instance-event-notification-attributes \  
  --instance-tag-attribute IncludeAllTagsOfInstance=true
```

Output:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [],  
    "IncludeAllTagsOfInstance": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Acara terjadwal untuk instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

Contoh 2: Untuk menyertakan tag tertentu dalam pemberitahuan acara

`register-instance-event-notification-attributes` Contoh berikut mencakup tag yang ditentukan dalam pemberitahuan acara. Anda tidak dapat menentukan tag jika `IncludeAllTagsOfInstance` adalah `true`.

```
aws ec2 register-instance-event-notification-attributes \  
  --instance-tag-attribute InstanceTagKeys="tag-key1","tag-key2"
```

**Output:**

```
{
  "InstanceTagAttribute": {
    "InstanceTagKeys": [
      "tag-key1",
      "tag-key2"
    ],
    "IncludeAllTagsOfInstance": false
  }
}
```

Untuk informasi selengkapnya, lihat [Acara terjadwal untuk instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans Linux.

- Untuk detail API, lihat [RegisterInstanceEventNotificationAttributes](#) di Referensi AWS CLI Perintah.

**register-transit-gateway-multicast-group-sources**

Contoh kode berikut menunjukkan cara menggunakan `register-transit-gateway-multicast-group-sources`.

**AWS CLI**

Untuk mendaftarkan sumber dengan grup multicast gateway transit.

`register-transit-gateway-multicast-group-sources` Contoh berikut mendaftarkan sumber grup antarmuka jaringan tertentu dengan grup multicast.

```
aws ec2 register-transit-gateway-multicast-group-sources \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-07f290fc3c090cbae
```

**Output:**

```
{
  "RegisteredMulticastGroupSources": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "RegisteredNetworkInterfaceIds": [
      "eni-07f290fc3c090cbae"
    ]
  }
}
```

```

    ],
    "GroupIpAddress": "224.0.1.0"
  }
}

```

Untuk informasi selengkapnya, lihat [Mendaftarkan Sumber dengan Grup Multicast di Panduan Pengguna AWS Transit Gateways](#).

- Untuk detail API, lihat [RegisterTransitGatewayMulticastGroupSources](#) di Referensi AWS CLI Perintah.

## register-transit-gateway-multicast-group-members

Contoh kode berikut menunjukkan cara menggunakan `register-transit-gateway-multicast-group-members`.

### AWS CLI

Untuk melihat informasi tentang asosiasi domain multicast gateway transit

`register-transit-gateway-multicast-group-members` Contoh berikut mengembalikan asosiasi untuk domain multicast tertentu.

```

aws ec2 register-transit-gateway-multicast-group-members \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-0e246d32695012e81

```

Output:

```

{
  "RegisteredMulticastGroupMembers": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "RegisteredNetworkInterfaceIds": [
      "eni-0e246d32695012e81"
    ],
    "GroupIpAddress": "224.0.1.0"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Pengguna Transit Gateways.

- Untuk detail API, lihat [RegisterTransitGatewayMulticastGroupMembers](#) di Referensi AWS CLI Perintah.

## register-transit-gateway-multicast-group-sources

Contoh kode berikut menunjukkan cara menggunakan `register-transit-gateway-multicast-group-sources`.

### AWS CLI

Untuk mendaftarkan sumber dengan grup multicast gateway transit.

`register-transit-gateway-multicast-group-sources` Contoh berikut mendaftarkan sumber grup antarmuka jaringan tertentu dengan grup multicast.

```
aws ec2 register-transit-gateway-multicast-group-sources \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-07f290fc3c090cbae
```

Output:

```
{
  "RegisteredMulticastGroupSources": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "RegisteredNetworkInterfaceIds": [
      "eni-07f290fc3c090cbae"
    ],
    "GroupIpAddress": "224.0.1.0"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola domain multicast](#) di Panduan Transit Gateways.

- Untuk detail API, lihat [RegisterTransitGatewayMulticastGroupSources](#) di Referensi AWS CLI Perintah.

## reject-transit-gateway-peering-attachment

Contoh kode berikut menunjukkan cara menggunakan `reject-transit-gateway-peering-attachment`.



## AWS CLI

Untuk menolak lampiran peering gateway transit

`reject-transit-gateway-peering-attachment` Contoh berikut menolak permintaan lampiran peering gateway transit yang ditentukan. `--region` Parameter menentukan Wilayah tempat gateway transit penerima berada.

```
aws ec2 reject-transit-gateway-peering-attachment \  
  --transit-gateway-attachment-id tgw-attach-4455667788aabbccd \  
  --region us-east-2
```

Output:

```
{  
  "TransitGatewayPeeringAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",  
    "RequesterTgwInfo": {  
      "TransitGatewayId": "tgw-123abc05e04123abc",  
      "OwnerId": "123456789012",  
      "Region": "us-west-2"  
    },  
    "AcceptorTgwInfo": {  
      "TransitGatewayId": "tgw-11223344aabbcc112",  
      "OwnerId": "123456789012",  
      "Region": "us-east-2"  
    },  
    "State": "rejecting",  
    "CreationTime": "2019-12-09T11:50:31.000Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Lampiran Pengintip Transit Gateway](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [RejectTransitGatewayPeeringAttachment](#) di Referensi AWS CLI Perintah.

## **reject-transit-gateway-vpc-attachment**

Contoh kode berikut menunjukkan cara menggunakan `reject-transit-gateway-vpc-attachment`.

## AWS CLI

Untuk menolak lampiran VPC gateway transit

`reject-transit-gateway-vpc-attachment` Contoh berikut menolak lampiran VPC gateway transit yang ditentukan.

```
aws ec2 reject-transit-gateway-vpc-attachment \  
  --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Output:

```
{  
  "TransitGatewayVpcAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",  
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",  
    "VpcId": "vpc-07e8ffd50fEXAMPLE",  
    "VpcOwnerId": "111122223333",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-0752213d59EXAMPLE"  
    ],  
    "CreationTime": "2019-07-10T17:33:46.000Z",  
    "Options": {  
      "DnsSupport": "enable",  
      "Ipv6Support": "disable"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Lampiran gateway transit ke VPC](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [RejectTransitGatewayVpcAttachment](#) di Referensi AWS CLI Perintah.

### **reject-transit-gateway-vpc-attachments**

Contoh kode berikut menunjukkan cara menggunakan `reject-transit-gateway-vpc-attachments`.

## AWS CLI

Untuk menolak lampiran VPC gateway transit

`reject-transit-gateway-vpc-attachment` Contoh berikut menolak lampiran VPC gateway transit yang ditentukan.

```
aws ec2 reject-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Output:

```
{
  "TransitGatewayVpcAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "VpcId": "vpc-07e8ffd50fEXAMPLE",
    "VpcOwnerId": "111122223333",
    "State": "pending",
    "SubnetIds": [
      "subnet-0752213d59EXAMPLE"
    ],
    "CreationTime": "2019-07-10T17:33:46.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Lampiran gateway transit ke VPC](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [RejectTransitGatewayVpcAttachments](#) di Referensi AWS CLI Perintah.

## `reject-vpc-endpoint-connections`

Contoh kode berikut menunjukkan cara menggunakan `reject-vpc-endpoint-connections`.

## AWS CLI

Untuk menolak permintaan koneksi titik akhir antarmuka

Contoh ini menolak permintaan koneksi endpoint yang ditentukan untuk layanan endpoint yang ditentukan.

Perintah:

```
aws ec2 reject-vpc-endpoint-connections --service-id vpce-svc-03d5ebb7d9579a2b3 --vpc-endpoint-ids vpce-0c1308d7312217abc
```

Output:

```
{
  "Unsuccessful": []
}
```

- Untuk detail API, lihat [RejectVpcEndpointConnections](#) di Referensi AWS CLI Perintah.

## **reject-vpc-peering-connection**

Contoh kode berikut menunjukkan cara menggunakan `reject-vpc-peering-connection`.

AWS CLI

Untuk menolak koneksi peering VPC

Contoh ini menolak permintaan koneksi peering VPC yang ditentukan.

Perintah:

```
aws ec2 reject-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [RejectVpcPeeringConnection](#) di Referensi AWS CLI Perintah.

## **release-address**

Contoh kode berikut menunjukkan cara menggunakan `release-address`.

## AWS CLI

Guna merilis alamat IP Elastis untuk EC2-Classic

Contoh ini merilis alamat IP Elastis untuk digunakan dengan instans di EC2-Classic. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 release-address --public-ip 198.51.100.0
```

Guna merilis alamat IP Elastis untuk EC2-VPC

Contoh ini merilis alamat IP Elastis untuk digunakan dengan instans di VPC. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 release-address --allocation-id eipalloc-64d5890a
```

- Untuk detail API, lihat [ReleaseAddress](#) di Referensi AWS CLI Perintah.

## release-hosts

Contoh kode berikut menunjukkan cara menggunakan `release-hosts`.

### AWS CLI

Untuk melepaskan host Khusus dari akun Anda

Untuk melepaskan host Khusus dari akun Anda. Instance yang ada di host harus dihentikan atau dihentikan sebelum host dapat dirilis.

Perintah:

```
aws ec2 release-hosts --host-id=h-0029d6e3cacf1b3da
```

Output:

```
{
  "Successful": [
```

```
    "h-0029d6e3cacf1b3da"  
    ],  
    "Unsuccessful": []  
}
```

- Untuk detail API, lihat [ReleaseHosts](#) di Referensi AWS CLI Perintah.

## release-ipam-pool-allocation

Contoh kode berikut menunjukkan cara menggunakan `release-ipam-pool-allocation`.

### AWS CLI

Untuk merilis alokasi kolam IPAM

Dalam contoh ini, Anda adalah admin yang didelegasikan IPAM yang mencoba menghapus kumpulan IPAM tetapi menerima kesalahan bahwa Anda tidak dapat menghapus kumpulan saat kumpulan memiliki alokasi. Anda menggunakan perintah ini untuk melepaskan alokasi kumpulan.

Perhatikan hal berikut:

Anda hanya dapat menggunakan perintah ini untuk alokasi kustom. Untuk menghapus alokasi sumber daya tanpa menghapus sumber daya, atur status yang dipantau ke `false` menggunakan [modify-ipam-resource-cidr](#). Untuk menyelesaikan permintaan ini, Anda memerlukan ID kolam IPAM, yang bisa Anda dapatkan. [describe-ipam-pools](#) Anda juga memerlukan ID alokasi, yang bisa Anda dapatkan dengan [get-ipam-pool-allocations](#). Jika Anda tidak ingin menghapus alokasi satu per satu, Anda dapat menggunakan `--cascade` option ketika Anda menghapus kolam IPAM untuk secara otomatis melepaskan alokasi apa pun di kolam sebelum menghapusnya. Ada sejumlah prasyarat sebelum menjalankan perintah ini. Untuk informasi selengkapnya, lihat [Melepaskan alokasi](#) di Panduan Pengguna Amazon VPC IPAM `--region`. Di mana Anda menjalankan perintah ini harus merupakan lokal dari kumpulan IPAM di mana alokasi berada.

`release-ipam-pool-allocation` Contoh berikut merilis alokasi kolam IPAM.

```
aws ec2 release-ipam-pool-allocation \  
  --ipam-pool-id ipam-pool-07bdd12d7c94e4693 \  
  --cidr 10.0.0.0/23 \  
  --ipam-pool-allocation-id ipam-pool-alloc-0e66a1f730da54791b99465b79e7d1e89 \  
  --region us-west-1
```

**Output:**

```
{
  "Success": true
}
```

Setelah Anda merilis alokasi, Anda mungkin ingin menjalankannya [delete-ipam-pool](#).

- Untuk detail API, lihat [ReleaseIpamPoolAllocation](#) di Referensi AWS CLI Perintah.

**replace-iam-instance-profile-association**

Contoh kode berikut menunjukkan cara menggunakan `replace-iam-instance-profile-association`.

**AWS CLI**

Guna mengganti profil instans IAM untuk instans

Contoh ini menggantikan profil instans IAM yang direpresentasikan oleh kaitan `iip-assoc-060bae234aac2e7fa` dengan profil instans IAM bernama `AdminRole`.

```
aws ec2 replace-iam-instance-profile-association \
  --iam-instance-profile Name=AdminRole \
  --association-id iip-assoc-060bae234aac2e7fa
```

**Output:**

```
{
  "IamInstanceProfileAssociation": {
    "InstanceId": "i-087711ddaf98f9489",
    "State": "associating",
    "AssociationId": "iip-assoc-0b215292fab192820",
    "IamInstanceProfile": {
      "Id": "AIPAJLNLDX3AMYZNWYYAY",
      "Arn": "arn:aws:iam::123456789012:instance-profile/AdminRole"
    }
  }
}
```

- Untuk detail API, lihat [ReplaceIamInstanceProfileAssociation](#) di Referensi AWS CLI Perintah.

## replace-network-acl-association

Contoh kode berikut menunjukkan cara menggunakan `replace-network-acl-association`.

### AWS CLI

Untuk mengganti ACL jaringan yang terkait dengan subnet

Contoh ini mengaitkan ACL jaringan tertentu dengan subnet untuk asosiasi ACL jaringan tertentu.

Perintah:

```
aws ec2 replace-network-acl-association --association-id aclassoc-e5b95c8c --
network-acl-id acl-5fb85d36
```

Output:

```
{
  "NewAssociationId": "aclassoc-3999875b"
}
```

- Untuk detail API, lihat [ReplaceNetworkAclAssociation](#) di Referensi AWS CLI Perintah.

## replace-network-acl-entry

Contoh kode berikut menunjukkan cara menggunakan `replace-network-acl-entry`.

### AWS CLI

Untuk mengganti entri ACL jaringan

Contoh ini menggantikan entri untuk ACL jaringan tertentu. Aturan baru 100 memungkinkan masuknya lalu lintas dari 203.0.113.12/24 pada port UDP 53 (DNS) ke subnet terkait.

Perintah:

```
aws ec2 replace-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-
number 100 --protocol udp --port-range From=53,To=53 --cidr-block 203.0.113.12/24 --
rule-action allow
```

- Untuk detail API, lihat [ReplaceNetworkAclEntry](#) di Referensi AWS CLI Perintah.



## replace-route-table-association

Contoh kode berikut menunjukkan cara menggunakan `replace-route-table-association`.

### AWS CLI

Untuk mengganti tabel rute yang terkait dengan subnet

Contoh ini mengaitkan tabel rute yang ditentukan dengan subnet untuk asosiasi tabel rute yang ditentukan.

Perintah:

```
aws ec2 replace-route-table-association --association-id rtbassoc-781d0d1a --route-table-id rtb-22574640
```

Output:

```
{
  "NewAssociationId": "rtbassoc-3a1f0f58"
}
```

- Untuk detail API, lihat [ReplaceRouteTableAssociation](#) di Referensi AWS CLI Perintah.

## replace-route

Contoh kode berikut menunjukkan cara menggunakan `replace-route`.

### AWS CLI

Untuk mengganti rute

Contoh ini menggantikan rute yang ditentukan dalam tabel rute yang ditentukan. Rute baru cocok dengan CIDR yang ditentukan dan mengirimkan lalu lintas ke gateway pribadi virtual yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 replace-route --route-table-id rtb-22574640 --destination-cidr-block 10.0.0.0/16 --gateway-id vgw-9a4cacf3
```

- Untuk detail API, lihat [ReplaceRoute](#) di Referensi AWS CLI Perintah.

## replace-transit-gateway-route

Contoh kode berikut menunjukkan cara menggunakan `replace-transit-gateway-route`.

### AWS CLI

Untuk mengganti rute yang ditentukan dalam tabel rute gateway transit yang ditentukan

`replace-transit-gateway-route` Contoh berikut menggantikan rute dalam tabel rute gateway transit yang ditentukan.

```
aws ec2 replace-transit-gateway-route \
  --destination-cidr-block 10.0.2.0/24 \
  --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE \
  --transit-gateway-route-table-id tgw-rtb-0a823edbbeEXAMPLE
```

### Output:

```
{
  "Route": {
    "DestinationCidrBlock": "10.0.2.0/24",
    "TransitGatewayAttachments": [
      {
        "ResourceId": "vpc-4EXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
        "ResourceType": "vpc"
      }
    ],
    "Type": "static",
    "State": "active"
  }
}
```

Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [ReplaceTransitGatewayRoute](#) di Referensi AWS CLI Perintah.

## report-instance-status

Contoh kode berikut menunjukkan cara menggunakan `report-instance-status`.

## AWS CLI

Untuk melaporkan umpan balik status untuk sebuah instans

Perintah contoh ini melaporkan umpan balik status untuk instance yang ditentukan.

Perintah:

```
aws ec2 report-instance-status --instances i-1234567890abcdef0 --status impaired --reason-codes unresponsive
```

- Untuk detail API, lihat [ReportInstanceStatus](#) di Referensi AWS CLI Perintah.

## request-spot-fleet

Contoh kode berikut menunjukkan cara menggunakan `request-spot-fleet`.

### AWS CLI

Untuk meminta armada Spot di subnet dengan harga terendah

Perintah contoh ini membuat permintaan armada Spot dengan dua spesifikasi peluncuran yang hanya berbeda dengan subnet. Armada Spot meluncurkan instans di subnet yang ditentukan dengan harga terendah. Jika instance diluncurkan dalam VPC default, mereka menerima alamat IP publik secara default. Jika instans diluncurkan dalam VPC nondefault, mereka tidak menerima alamat IP publik secara default.

Perhatikan bahwa Anda tidak dapat menentukan subnet yang berbeda dari Availability Zone yang sama dalam permintaan armada Spot.

Perintah:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
```

```
    "ImageId": "ami-1a2b3c4d",
    "KeyName": "my-key-pair",
    "SecurityGroups": [
      {
        "GroupId": "sg-1a2b3c4d"
      }
    ],
    "InstanceType": "m3.medium",
    "SubnetId": "subnet-1a2b3c4d, subnet-3c4d5e6f",
    "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
    }
  }
]
```

#### Output:

```
{
  "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

Untuk meminta armada Spot di Availability Zone dengan harga terendah

Perintah contoh ini membuat permintaan armada Spot dengan dua spesifikasi peluncuran yang hanya berbeda dengan Availability Zone. Armada Spot meluncurkan instans di Availability Zone yang ditentukan dengan harga terendah. Jika akun Anda hanya mendukung EC2-VPC, Amazon EC2 meluncurkan instans Spot di subnet default Availability Zone. Jika akun Anda mendukung EC2-Classic, Amazon EC2 meluncurkan instans di EC2-Classic di Availability Zone.

#### Perintah:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

#### config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
```

```
    "ImageId": "ami-1a2b3c4d",
    "KeyName": "my-key-pair",
    "SecurityGroups": [
      {
        "GroupId": "sg-1a2b3c4d"
      }
    ],
    "InstanceType": "m3.medium",
    "Placement": {
      "AvailabilityZone": "us-west-2a, us-west-2b"
    },
    "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
    }
  }
]
```

Untuk meluncurkan instans Spot di subnet dan menentukannya alamat IP publik

Perintah contoh ini menetapkan alamat publik ke instance yang diluncurkan di VPC nondefault. Perhatikan bahwa ketika Anda menentukan antarmuka jaringan, Anda harus menyertakan ID subnet dan ID grup keamanan menggunakan antarmuka jaringan.

Perintah:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
      "ImageId": "ami-1a2b3c4d",
      "KeyName": "my-key-pair",
      "InstanceType": "m3.medium",
      "NetworkInterfaces": [
        {
          "DeviceIndex": 0,
          "SubnetId": "subnet-1a2b3c4d",
```

```
        "Groups": [ "sg-1a2b3c4d" ],
        "AssociatePublicIpAddress": true
      }
    ],
    "IamInstanceProfile": {
      "Arn": "arn:aws:iam::880185128111:instance-profile/my-iam-role"
    }
  }
]
}
```

Untuk meminta armada Spot menggunakan strategi alokasi yang beragam

Perintah contoh ini membuat permintaan armada Spot yang meluncurkan 30 instance menggunakan strategi alokasi yang beragam. Spesifikasi peluncuran berbeda menurut jenis instans. Armada Spot mendistribusikan instans di seluruh spesifikasi peluncuran sehingga ada 10 instance dari setiap jenis.

Perintah:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

config.json:

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
      "ImageId": "ami-1a2b3c4d",
      "InstanceType": "c4.2xlarge",
      "SubnetId": "subnet-1a2b3c4d"
    },
    {
      "ImageId": "ami-1a2b3c4d",
      "InstanceType": "m3.2xlarge",
      "SubnetId": "subnet-1a2b3c4d"
    },
    {
      "ImageId": "ami-1a2b3c4d",
```

```
        "InstanceType": "r3.2xlarge",
        "SubnetId": "subnet-1a2b3c4d"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Permintaan Armada Spot](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [RequestSpotFleet](#) di Referensi AWS CLI Perintah.

## request-spot-instances

Contoh kode berikut menunjukkan cara menggunakan `request-spot-instances`.

### AWS CLI

Untuk meminta Instans Spot

Perintah contoh ini membuat permintaan Instans Spot satu kali untuk lima instance di Availability Zone yang ditentukan. Jika akun Anda hanya mendukung EC2-VPC, Amazon EC2 meluncurkan instans di subnet default dari Availability Zone yang ditentukan. Jika akun Anda mendukung EC2-Classic, Amazon EC2 meluncurkan instans di EC2-Classic di Availability Zone yang ditentukan.

Perintah:

```
aws ec2 request-spot-instances --spot-price "0.03" --instance-count 5 --type "one-time" --launch-specification file://specification.json
```

Spesifikasi.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d" ],
  "InstanceType": "m3.medium",
  "Placement": {
    "AvailabilityZone": "us-west-2a"
  },
  "IamInstanceProfile": {
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

```
}
```

## Output:

```
{
  "SpotInstanceRequests": [
    {
      "Status": {
        "UpdateTime": "2014-03-25T20:54:21.000Z",
        "Code": "pending-evaluation",
        "Message": "Your Spot request has been submitted for review, and is
pending evaluation."
      },
      "ProductDescription": "Linux/UNIX",
      "SpotInstanceRequestId": "sir-df6f405d",
      "State": "open",
      "LaunchSpecification": {
        "Placement": {
          "AvailabilityZone": "us-west-2a"
        },
        "ImageId": "ami-1a2b3c4d",
        "KeyName": "my-key-pair",
        "SecurityGroups": [
          {
            "GroupName": "my-security-group",
            "GroupId": "sg-1a2b3c4d"
          }
        ],
        "Monitoring": {
          "Enabled": false
        },
        "IamInstanceProfile": {
          "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
        },
        "InstanceType": "m3.medium"
      },
      "Type": "one-time",
      "CreateTime": "2014-03-25T20:54:20.000Z",
      "SpotPrice": "0.050000"
    },
    ...
  ]
}
```



Perintah contoh ini membuat permintaan Instans Spot satu kali untuk lima instance di subnet yang ditentukan. Amazon EC2 meluncurkan instans di subnet yang ditentukan. Jika VPC adalah VPC nondefault, instance tidak menerima alamat IP publik secara default.

Perintah:

```
aws ec2 request-spot-instances --spot-price "0.050" --instance-count 5 --type "one-time" --launch-specification file://specification.json
```

Spesifikasi.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "SecurityGroupIds": [ "sg-1a2b3c4d" ],
  "InstanceType": "m3.medium",
  "SubnetId": "subnet-1a2b3c4d",
  "IamInstanceProfile": {
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Output:

```
{
  "SpotInstanceRequests": [
    {
      "Status": {
        "UpdateTime": "2014-03-25T22:21:58.000Z",
        "Code": "pending-evaluation",
        "Message": "Your Spot request has been submitted for review, and is pending evaluation."
      },
      "ProductDescription": "Linux/UNIX",
      "SpotInstanceRequestId": "sir-df6f405d",
      "State": "open",
      "LaunchSpecification": {
        "Placement": {
          "AvailabilityZone": "us-west-2a"
        }
      },
      "ImageId": "ami-1a2b3c4d",
      "SecurityGroups": [
        {

```

```

        "GroupName": "my-security-group",
        "GroupID": "sg-1a2b3c4d"
    }
]
"SubnetId": "subnet-1a2b3c4d",
"Monitoring": {
    "Enabled": false
},
"IamInstanceProfile": {
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
},
"InstanceType": "m3.medium",
},
"Type": "one-time",
"CreateTime": "2014-03-25T22:21:58.000Z",
"SpotPrice": "0.050000"
},
...
]
}

```

Contoh ini menetapkan alamat IP publik ke Instans Spot yang Anda luncurkan di VPC nondefault. Perhatikan bahwa ketika Anda menentukan antarmuka jaringan, Anda harus menyertakan ID subnet dan ID grup keamanan menggunakan antarmuka jaringan.

Perintah:

```
aws ec2 request-spot-instances --spot-price "0.050" --instance-count 1 --type "one-time" --launch-specification file://specification.json
```

Spesifikasi.json:

```

{
  "ImageId": "ami-1a2b3c4d",
  "KeyName": "my-key-pair",
  "InstanceType": "m3.medium",
  "NetworkInterfaces": [
    {
      "DeviceIndex": 0,
      "SubnetId": "subnet-1a2b3c4d",
      "Groups": [ "sg-1a2b3c4d" ],
      "AssociatePublicIpAddress": true
    }
  ]
}

```

```
],  
  "IamInstanceProfile": {  
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
  }  
}
```

- Untuk detail API, lihat [RequestSpotInstances](#) di Referensi AWS CLI Perintah.

## reset-address-attribute

Contoh kode berikut menunjukkan cara menggunakan `reset-address-attribute`.

### AWS CLI

Untuk mengatur ulang atribut nama domain yang terkait dengan alamat IP elastis

`reset-address-attribute` Contoh berikut mengatur ulang atribut nama domain dari alamat IP elastis.

Linux:

```
aws ec2 reset-address-attribute \  
  --allocation-id eipalloc-abcdef01234567890 \  
  --attribute domain-name
```

Windows:

```
aws ec2 reset-address-attribute ^  
  --allocation-id eipalloc-abcdef01234567890 ^  
  --attribute domain-name
```

Output:

```
{  
  "Addresses": [  
    {  
      "PublicIp": "192.0.2.0",  
      "AllocationId": "eipalloc-abcdef01234567890",  
      "PtrRecord": "example.com."  
      "PtrRecordUpdate": {  
        "Value": "example.net.",  
        "Status": "PENDING"  
      }  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Untuk memantau perubahan yang tertunda, lihat [describe-addresses-attribute](#) di Referensi Perintah AWS CLI.

- Untuk detail API, lihat [ResetAddressAttribute](#) di Referensi AWS CLI Perintah.

## **reset-ebs-default-kms-key-id**

Contoh kode berikut menunjukkan cara menggunakan `reset-ebs-default-kms-key-id`.

### AWS CLI

Untuk mengatur ulang CMK default Anda untuk enkripsi EBS

`reset-ebs-default-kms-key-id` Contoh berikut me-reset CMK default untuk enkripsi EBS untuk AWS akun Anda di Wilayah saat ini.

```
aws ec2 reset-ebs-default-kms-key-id
```

Output:

```
{  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/8c5b2c63-b9bc-45a3-a87a-5513eEXAMPLE"  
}
```

- Untuk detail API, lihat [ResetEbsDefaultKmsKeyId](#) di Referensi AWS CLI Perintah.

## **reset-fpga-image-attribute**

Contoh kode berikut menunjukkan cara menggunakan `reset-fpga-image-attribute`.

### AWS CLI

Untuk mengatur ulang atribut gambar Amazon FPGA

Contoh ini mengatur ulang izin pemuatan untuk AFI yang ditentukan.

Perintah:

```
aws ec2 reset-fpga-image-attribute --fpga-image-id afi-0d123e123bfc85abc --attribute
loadPermission
```

Output:

```
{
  "Return": true
}
```

- Untuk detail API, lihat [ResetFpgaImageAttribute](#) di Referensi AWS CLI Perintah.

## **reset-image-attribute**

Contoh kode berikut menunjukkan cara menggunakan `reset-image-attribute`.

### AWS CLI

Untuk mengatur ulang atribut `launchPermission`

Contoh ini me-reset `launchPermission` atribut untuk AMI yang ditentukan ke nilai defaultnya. Secara default, AMI bersifat pribadi. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 reset-image-attribute --image-id ami-5731123e --attribute launchPermission
```

- Untuk detail API, lihat [ResetImageAttribute](#) di Referensi AWS CLI Perintah.

## **reset-instance-attribute**

Contoh kode berikut menunjukkan cara menggunakan `reset-instance-attribute`.

### AWS CLI

Untuk mengatur ulang `sourceDestCheck` atribut

Contoh ini me-reset `sourceDestCheck` atribut dari instance tertentu. Instans harus dalam VPC. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --attribute
sourceDestCheck
```

Untuk mengatur ulang atribut kernel

Contoh ini me-reset kernel atribut dari instance tertentu. Instans harus berada dalam status stopped. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --attribute
kernel
```

Untuk mengatur ulang atribut ramdisk

Contoh ini me-reset ramdisk atribut dari instance tertentu. Instans harus berada dalam status stopped. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --attribute
ramdisk
```

- Untuk detail API, lihat [ResetInstanceAttribute](#) di Referensi AWS CLI Perintah.

## **reset-network-interface-attribute**

Contoh kode berikut menunjukkan cara menggunakan `reset-network-interface-attribute`.

### AWS CLI

Untuk mengatur ulang atribut antarmuka jaringan

`reset-network-interface-attribute` Contoh berikut me-reset nilai atribut pemeriksaan sumber/tujuan ke. true

```
aws ec2 reset-network-interface-attribute \
  --network-interface-id eni-686ea200 \
  --source-dest-check
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [ResetNetworkInterfaceAttribute](#) di Referensi AWS CLI Perintah.

## **reset-snapshot-attribute**

Contoh kode berikut menunjukkan cara menggunakan `reset-snapshot-attribute`.

### AWS CLI

Untuk mengatur ulang atribut snapshot

Contoh ini mengatur ulang izin volume buat untuk snapshot. `snap-1234567890abcdef0` Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 reset-snapshot-attribute --snapshot-id snap-1234567890abcdef0 --attribute createVolumePermission
```

- Untuk detail API, lihat [ResetSnapshotAttribute](#) di Referensi AWS CLI Perintah.

## **restore-address-to-classic**

Contoh kode berikut menunjukkan cara menggunakan `restore-address-to-classic`.

### AWS CLI

Untuk mengembalikan alamat ke EC2-Classic

Contoh ini mengembalikan alamat IP Elastis 198.51.100.0 ke platform EC2-Classic.

Perintah:

```
aws ec2 restore-address-to-classic --public-ip 198.51.100.0
```

Output:

```
{
  "Status": "MoveInProgress",
  "PublicIp": "198.51.100.0"
}
```

- Untuk detail API, lihat [RestoreAddressToClassic](#) di Referensi AWS CLI Perintah.

## restore-image-from-recycle-bin

Contoh kode berikut menunjukkan cara menggunakan `restore-image-from-recycle-bin`.

### AWS CLI

Untuk mengembalikan gambar dari Recycle Bin

`restore-image-from-recycle-bin` Contoh berikut mengembalikan AMI `ami-0111222333444abcd` dari Recycle Bin.

```
aws ec2 restore-image-from-recycle-bin \
  --image-id ami-0111222333444abcd
```

Output:

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Memulihkan AMI dari Recycle Bin](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [RestoreImageFromRecycleBin](#) di Referensi AWS CLI Perintah.

## restore-managed-prefix-list-version

Contoh kode berikut menunjukkan cara menggunakan `restore-managed-prefix-list-version`.

### AWS CLI

US-barat-2\*\*Untuk mengembalikan versi daftar awalan\*\*

Berikut ini `restore-managed-prefix-list-version` mengembalikan entri dari versi 1 dari daftar awalan yang ditentukan.

```
aws ec2 restore-managed-prefix-list-version \
  --prefix-list-id pl-0123456abcabc1 \
  --current-version 2 \
  --previous-version 1
```



**Output:**

```
{
  "PrefixList": {
    "PrefixListId": "pl-0123456abcabcabc1",
    "AddressFamily": "IPv4",
    "State": "restore-in-progress",
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
    "PrefixListName": "vpc-cidrs",
    "MaxEntries": 10,
    "Version": 2,
    "OwnerId": "123456789012"
  }
}
```

Untuk informasi selengkapnya, lihat [Daftar awalan dikelola](#) di Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [RestoreManagedPrefixListVersion](#) di Referensi AWS CLI Perintah.

**restore-snapshot-from-recycle-bin**

Contoh kode berikut menunjukkan cara menggunakan `restore-snapshot-from-recycle-bin`.

**AWS CLI**

Untuk mengembalikan snapshot dari Recycle Bin

`restore-snapshot-from-recycle-bin` Contoh berikut mengembalikan snapshot dari Recycle Bin. Saat Anda memulihkan snapshot dari Keranjang Sampah, snapshot segera tersedia untuk digunakan, dan akan dihapus dari Keranjang Sampah. Anda dapat menggunakan AMI yang dipulihkan dengan cara yang sama seperti Anda menggunakan AMI lainnya di akun Anda.

```
aws ec2 restore-snapshot-from-recycle-bin \
  --snapshot-id snap-01234567890abcdef
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang Recycle Bin untuk Amazon EBS, lihat [Memulihkan snapshot dari Recycle Bin di Panduan Pengguna](#) Amazon EC2.

- Untuk detail API, lihat [RestoreSnapshotFromRecycleBin](#) di Referensi AWS CLI Perintah.

## restore-snapshot-tier

Contoh kode berikut menunjukkan cara menggunakan `restore-snapshot-tier`.

### AWS CLI

Contoh 1: Untuk memulihkan snapshot yang diarsipkan secara permanen

`restore-snapshot-tier` Contoh berikut secara permanen mengembalikan snapshot yang ditentukan. Tentukan `--snapshot-id` dan sertakan `permanent-restore` opsi.

```
aws ec2 restore-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --permanent-restore
```

Output:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "IsPermanentRestore": true  
}
```

Untuk informasi selengkapnya tentang pengarsipan snapshot, lihat [Mengarsipkan snapshot Amazon EBS < https://docs.aws.amazon.com/AWS\\_EC2/terbaru/ /snapshot-archive.html](https://docs.aws.amazon.com/AWS_EC2/terbaru/ /snapshot-archive.html) [UserGuide](#) > di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk memulihkan snapshot yang diarsipkan sementara

`restore-snapshot-tier` Contoh berikut untuk sementara mengembalikan snapshot yang ditentukan. Abaikan `--permanent-restore` opsi. Tentukan `--snapshot-id` dan `temporary-restore-days`, untuk, tentukan jumlah hari untuk mengembalikan snapshot. `temporary-restore-days` harus ditentukan dalam beberapa hari. Rentang yang diizinkan adalah 1 untuk 180. Jika Anda tidak menentukan nilai, secara otomatis nilainya adalah 1 hari.

```
aws ec2 restore-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --temporary-restore-days 5
```

Output:

```
{
```

```
"SnapshotId": "snap-01234567890abcdef",  
"RestoreDuration": 5,  
"IsPermanentRestore": false  
}
```

Untuk informasi selengkapnya tentang pengarsipan snapshot, lihat Mengarsipkan snapshot Amazon EBS < [https://docs.aws.amazon.com/AWS EC2/terbaru/snapshot-archive.html](https://docs.aws.amazon.com/AWS_EC2/terbaru/snapshot-archive.html) UserGuide > di Panduan Pengguna Amazon EC2.

Contoh 3: Untuk memodifikasi periode pemulihan

`restore-snapshot-tier` Contoh berikut mengubah periode pemulihan untuk snapshot yang ditentukan menjadi 10 hari.

```
aws ec2 restore-snapshot-tier \  
--snapshot-id snap-01234567890abcdef  
--temporary-restore-days 10
```

Output:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "RestoreDuration": 10,  
  "IsPermanentRestore": false  
}
```

Untuk informasi selengkapnya tentang pengarsipan snapshot, lihat Mengarsipkan snapshot Amazon EBS < [https://docs.aws.amazon.com/AWS EC2/terbaru/snapshot-archive.html](https://docs.aws.amazon.com/AWS_EC2/terbaru/snapshot-archive.html) UserGuide > di Panduan Pengguna Amazon EC2.

Contoh 4: Untuk memodifikasi jenis pemulihan

`restore-snapshot-tier` Contoh berikut mengubah jenis pemulihan untuk snapshot yang ditentukan dari sementara ke permanen.

```
aws ec2 restore-snapshot-tier \  
--snapshot-id snap-01234567890abcdef  
--permanent-restore
```

Output:

```
{
  "SnapshotId": "snap-01234567890abcdef",
  "IsPermanentRestore": true
}
```

Untuk informasi selengkapnya tentang pengarsipan snapshot, lihat [Mengarsipkan snapshot Amazon EBS < https://docs.aws.amazon.com/AWS EC2/terbaru/ /snapshot-archive.html UserGuide >](https://docs.aws.amazon.com/AWS_EC2/terbaru/snapshot-archive.html) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [RestoreSnapshotTier](#) di Referensi AWS CLI Perintah.

## revoke-client-vpn-ingress

Contoh kode berikut menunjukkan cara menggunakan `revoke-client-vpn-ingress`.

### AWS CLI

Untuk mencabut aturan otorisasi untuk titik akhir Client VPN

`revoke-client-vpn-ingress` Contoh berikut mencabut aturan untuk akses internet (`0.0.0.0/0`) untuk semua grup.

```
aws ec2 revoke-client-vpn-ingress \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --target-network-cidr 0.0.0.0/0 --revoke-all-groups
```

### Output:

```
{
  "Status": {
    "Code": "revoking"
  }
}
```

Untuk informasi selengkapnya, lihat [Aturan Otorisasi](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [RevokeClientVpnIngress](#) di Referensi AWS CLI Perintah.

## revoke-security-group-egress

Contoh kode berikut menunjukkan cara menggunakan `revoke-security-group-egress`.

## AWS CLI

Contoh 1: Untuk menghapus aturan yang memungkinkan lalu lintas keluar ke rentang alamat tertentu

`revoke-security-group-egress` Contoh perintah berikut menghapus aturan yang memberikan akses ke rentang alamat yang ditentukan pada port TCP 80.

```
aws ec2 revoke-security-group-egress \  
  --group-id sg-026c12253ce15eff7 \  
  --ip-permissions \  
  [{IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=10.0.0.0/16}]}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Grup keamanan](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menghapus aturan yang memungkinkan lalu lintas keluar ke grup keamanan tertentu

`revoke-security-group-egress` Contoh perintah berikut menghapus aturan yang memberikan akses ke grup keamanan tertentu pada port TCP 80.

```
aws ec2 revoke-security-group-egress \  
  --group-id sg-026c12253ce15eff7 \  
  --ip-permissions '[{"IpProtocol": "tcp", "FromPort": 443, "ToPort":   
  443, "UserIdGroupPairs": [{"GroupId": "sg-06df23a01ff2df86d"}]}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Grup keamanan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [RevokeSecurityGroupEgress](#) di Referensi AWS CLI Perintah.

## **revoke-security-group-ingress**

Contoh kode berikut menunjukkan cara menggunakan `revoke-security-group-ingress`.

## AWS CLI

Contoh 1: Untuk menghapus aturan dari grup keamanan

`revoke-security-group-ingress` Contoh berikut menghapus akses port TCP 22 untuk rentang `203.0.113.0/24` alamat dari grup keamanan yang ditentukan untuk VPC default.

```
aws ec2 revoke-security-group-ingress \  
  --group-name mySecurityGroup \  
  --protocol tcp \  
  --port 22 \  
  --cidr 203.0.113.0/24
```

Perintah ini tidak menghasilkan output jika berhasil.

Untuk informasi selengkapnya, lihat [Grup keamanan](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk menghapus aturan menggunakan set izin IP

`revoke-security-group-ingress` Contoh berikut menggunakan `ip-permissions` parameter untuk menghapus aturan masuk yang memungkinkan pesan ICMP Destination Unreachable: Fragmentation Needed and Don't Fragment was Set (Tipe 3, Kode 4).

```
aws ec2 revoke-security-group-ingress \  
  --group-id sg-026c12253ce15eff7 \  
  --ip-permissions \  
  IpProtocol=icmp,FromPort=3,ToPort=4,IpRanges=[{CidrIp=0.0.0.0/0}]
```

Perintah ini tidak menghasilkan output jika berhasil.

Untuk informasi selengkapnya, lihat [Grup keamanan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [RevokeSecurityGroupIngress](#) di Referensi AWS CLI Perintah.

## run-instances

Contoh kode berikut menunjukkan cara menggunakan `run-instances`.

### AWS CLI

Contoh 1: Untuk meluncurkan instans ke subnet default

Contoh `run-instances` berikut meluncurkan instans tunggal tipe `t2.micro` ke subnet default untuk Wilayah saat ini dan mengaitkannya dengan subnet default untuk VPC default pada

Wilayah tersebut. Pasangan Kunci bersifat opsional jika Anda tidak berencana untuk terhubung ke instans menggunakan SSH (Linux) atau RDP (Windows).

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --key-name MyKeyPair
```

Output:

```
{  
  "Instances": [  
    {  
      "AmiLaunchIndex": 0,  
      "ImageId": "ami-0abcdef1234567890",  
      "InstanceId": "i-1231231230abcdef0",  
      "InstanceType": "t2.micro",  
      "KeyName": "MyKeyPair",  
      "LaunchTime": "2018-05-10T08:05:20.000Z",  
      "Monitoring": {  
        "State": "disabled"  
      },  
      "Placement": {  
        "AvailabilityZone": "us-east-2a",  
        "GroupName": "",  
        "Tenancy": "default"  
      },  
      "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",  
      "PrivateIpAddress": "10.0.0.157",  
      "ProductCodes": [],  
      "PublicDnsName": "",  
      "State": {  
        "Code": 0,  
        "Name": "pending"  
      },  
      "StateTransitionReason": "",  
      "SubnetId": "subnet-04a636d18e83cfacb",  
      "VpcId": "vpc-1234567890abcdef0",  
      "Architecture": "x86_64",  
      "BlockDeviceMappings": [],  
      "ClientToken": "",  
      "EbsOptimized": false,  
      "Hypervisor": "xen",
```

```
"NetworkInterfaces": [
  {
    "Attachment": {
      "AttachTime": "2018-05-10T08:05:20.000Z",
      "AttachmentId": "eni-attach-0e325c07e928a0405",
      "DeleteOnTermination": true,
      "DeviceIndex": 0,
      "Status": "attaching"
    },
    "Description": "",
    "Groups": [
      {
        "GroupName": "MySecurityGroup",
        "GroupId": "sg-0598c7d356eba48d7"
      }
    ],
    "Ipv6Addresses": [],
    "MacAddress": "0a:ab:58:e0:67:e2",
    "NetworkInterfaceId": "eni-0c0a29997760baee7",
    "OwnerId": "123456789012",
    "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",
    "PrivateIpAddress": "10.0.0.157",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
        "PrivateIpAddress": "10.0.0.157"
      }
    ],
    "SourceDestCheck": true,
    "Status": "in-use",
    "SubnetId": "subnet-04a636d18e83cfacb",
    "VpcId": "vpc-1234567890abcdef0",
    "InterfaceType": "interface"
  }
],
"RootDeviceName": "/dev/xvda",
"RootDeviceType": "ebs",
"SecurityGroups": [
  {
    "GroupName": "MySecurityGroup",
    "GroupId": "sg-0598c7d356eba48d7"
  }
]
```



```

    ],
    "SourceDestCheck": true,
    "StateReason": {
      "Code": "pending",
      "Message": "pending"
    },
    "Tags": [],
    "VirtualizationType": "hvm",
    "CpuOptions": {
      "CoreCount": 1,
      "ThreadsPerCore": 1
    },
    "CapacityReservationSpecification": {
      "CapacityReservationPreference": "open"
    },
    "MetadataOptions": {
      "State": "pending",
      "HttpTokens": "optional",
      "HttpPutResponseHopLimit": 1,
      "HttpEndpoint": "enabled"
    }
  }
],
"OwnerId": "123456789012",
"ReservationId": "r-02a3f596d91211712"
}

```

Contoh 2: Untuk meluncurkan instans ke subnet non-default dan menambahkan alamat IP publik

Contoh `run-instances` berikut meminta alamat IP publik untuk instans yang Anda luncurkan ke subnet non-default. Instans tersebut dikaitkan dengan grup keamanan tertentu.

```

aws ec2 run-instances \
  --image-id ami-0abcdef1234567890 \
  --instance-type t2.micro \
  --subnet-id subnet-08fc749671b2d077c \
  --security-group-ids sg-0b0384b66d7d692f9 \
  --associate-public-ip-address \
  --key-name MyKeyPair

```

Untuk contoh output `run-instances`, lihat Contoh 1.

Contoh 3: Untuk meluncurkan instans dengan volume tambahan

Contoh `run-instances` berikut menggunakan pemetaan perangkat blok, yang ditentukan dalam `mapping.json`, untuk melampirkan volume tambahan saat peluncuran. Pemetaan perangkat blok dapat menentukan volume EBS, volume penyimpanan instans, atau volume EBS dan volume penyimpanan instans.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --subnet-id subnet-08fc749671b2d077c \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --key-name MyKeyPair \  
  --block-device-mappings file://mapping.json
```

Isi dari `mapping.json`. Contoh ini menambahkan `/dev/sdh` sebagai volume EBS kosong dengan ukuran 100 GiB.

```
[  
  {  
    "DeviceName": "/dev/sdh",  
    "Ebs": {  
      "VolumeSize": 100  
    }  
  }  
]
```

Isi dari `mapping.json`. Contoh ini menambahkan `ephemeral1` sebagai volume penyimpanan instans.

```
[  
  {  
    "DeviceName": "/dev/sdc",  
    "VirtualName": "ephemeral1"  
  }  
]
```

Untuk contoh output `run-instances`, lihat Contoh 1.

Untuk informasi selengkapnya tentang pemetaan perangkat blok, lihat [Pemetaan perangkat blok](#) di Panduan Pengguna Amazon EC2.

Contoh 4: Untuk meluncurkan instans dan menambahkan tanda pada pembuatan

Contoh `run-instances` berikut menambahkan tanda dengan kunci `webserver` dan nilai `production` pada instans. Perintah tersebut juga menerapkan tanda dengan kunci `cost-center` dan nilai `cc123` pada volume EBS yang dibuat (dalam hal ini, volume `root`).

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 1 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --tag-specifications  
'ResourceType=instance,Tags=[{Key=webserver,Value=production}]'  
'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

Untuk contoh output `run-instances`, lihat Contoh 1.

Contoh 5: Untuk meluncurkan instans dengan data pengguna

Contoh `run-instances` berikut meneruskan data pengguna dalam file bernama `my_script.txt` yang berisi skrip konfigurasi untuk instans Anda. Skrip berjalan saat peluncuran.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 1 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --user-data file://my_script.txt
```

Untuk contoh output `run-instances`, lihat Contoh 1.

Untuk informasi selengkapnya tentang data pengguna instans, lihat [Bekerja dengan data pengguna instans](#) di Panduan Pengguna Amazon EC2.

Contoh 6: Untuk meluncurkan instans performa yang dapat melonjak

Contoh `run-instances` berikut meluncurkan instans `t2.micro` dengan opsi kredit `unlimited`. Ketika meluncurkan instans T2, jika Anda tidak menentukan `--credit-specification`,

default-nya adalah opsi kredit standard. Ketika meluncurkan instans T3, default-nya adalah opsi kredit unlimited.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 1 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --credit-specification CpuCredits=unlimited
```

Untuk contoh output `run-instances`, lihat Contoh 1.

Untuk informasi selengkapnya tentang instans performa yang dapat melonjak, lihat [Instans performa yang dapat melonjak](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [RunInstances](#) di Referensi AWS CLI Perintah.

## run-scheduled-instances

Contoh kode berikut menunjukkan cara menggunakan `run-scheduled-instances`.

### AWS CLI

Untuk meluncurkan Instance Terjadwal

Contoh ini meluncurkan Instance Terjadwal yang ditentukan dalam VPC.

Perintah:

```
aws ec2 run-scheduled-instances --scheduled-instance-id  
sci-1234-1234-1234-1234-123456789012 --instance-count 1 --launch-specification  
file://launch-specification.json
```

Launch-specification.json:

```
{  
  "ImageId": "ami-12345678",  
  "KeyName": "my-key-pair",  
  "InstanceType": "c4.large",
```

```
"NetworkInterfaces": [  
  {  
    "DeviceIndex": 0,  
    "SubnetId": "subnet-12345678",  
    "AssociatePublicIpAddress": true,  
    "Groups": ["sg-12345678"]  
  }  
],  
"IamInstanceProfile": {  
  "Name": "my-iam-role"  
}  
}
```

### Output:

```
{  
  "InstanceIdSet": [  
    "i-1234567890abcdef0"  
  ]  
}
```

Contoh ini meluncurkan Instance Terjadwal yang ditentukan di EC2-Classic.

### Perintah:

```
aws ec2 run-scheduled-instances --scheduled-instance-id  
sci-1234-1234-1234-1234-123456789012 --instance-count 1 --launch-specification  
file://launch-specification.json
```

### Launch-specification.json:

```
{  
  "ImageId": "ami-12345678",  
  "KeyName": "my-key-pair",  
  "SecurityGroupIds": ["sg-12345678"],  
  "InstanceType": "c4.large",  
  "Placement": {  
    "AvailabilityZone": "us-west-2b"  
  }  
  "IamInstanceProfile": {  
    "Name": "my-iam-role"  
  }  
}
```

```
}
```

Output:

```
{
  "InstanceIdSet": [
    "i-1234567890abcdef0"
  ]
}
```

- Untuk detail API, lihat [RunScheduledInstances](#) di Referensi AWS CLI Perintah.

## search-local-gateway-routes

Contoh kode berikut menunjukkan cara menggunakan `search-local-gateway-routes`.

### AWS CLI

Untuk mencari rute dalam tabel rute gateway lokal

`search-local-gateway-routes` Contoh berikut mencari rute statis dalam tabel rute gateway lokal yang ditentukan.

```
aws ec2 search-local-gateway-routes \
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE \
  --filters "Name=type,Values=static"
```

Output:

```
{
  "Route": {
    "DestinationCidrBlock": "0.0.0.0/0",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
    "Type": "static",
    "State": "deleted",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7EXAMPLE"
  }
}
```

- Untuk detail API, lihat [SearchLocalGatewayRoutes](#) di Referensi AWS CLI Perintah.

## search-transit-gateway-multicast-groups

Contoh kode berikut menunjukkan cara menggunakan `search-transit-gateway-multicast-groups`.

### AWS CLI

Untuk mencari satu atau beberapa grup multicast gateway transit dan mengembalikan informasi keanggotaan grup

`search-transit-gateway-multicast-groups` Contoh berikut mengembalikan keanggotaan grup grup multicast tertentu.

```
aws ec2 search-transit-gateway-multicast-groups \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-000fb24d04EXAMPLE
```

Output:

```
{
  "MulticastGroups": [
    {
      "GroupIpAddress": "224.0.1.0",
      "TransitGatewayAttachmentId": "tgw-attach-0372e72386EXAMPLE",
      "SubnetId": "subnet-0187aff814EXAMPLE",
      "ResourceId": "vpc-0065acced4EXAMPLE",
      "ResourceType": "vpc",
      "NetworkInterfaceId": "eni-03847706f6EXAMPLE",
      "GroupMember": false,
      "GroupSource": true,
      "SourceType": "static"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola grup multicast](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [SearchTransitGatewayMulticastGroups](#) di Referensi AWS CLI Perintah.

## search-transit-gateway-routes

Contoh kode berikut menunjukkan cara menggunakan `search-transit-gateway-routes`.

## AWS CLI

Untuk mencari rute dalam tabel rute gateway transit yang ditentukan

`search-transit-gateway-routes` Contoh berikut mengembalikan semua rute yang bertipe `static` dalam tabel rute yang ditentukan.

```
aws ec2 search-transit-gateway-routes \  
  --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \  
  --filters "Name=type,Values=static"
```

Output:

```
{  
  "Routes": [  
    {  
      "DestinationCidrBlock": "10.0.2.0/24",  
      "TransitGatewayAttachments": [  
        {  
          "ResourceId": "vpc-4EXAMPLE",  
          "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",  
          "ResourceType": "vpc"  
        }  
      ],  
      "Type": "static",  
      "State": "active"  
    },  
    {  
      "DestinationCidrBlock": "10.1.0.0/24",  
      "TransitGatewayAttachments": [  
        {  
          "ResourceId": "vpc-4EXAMPLE",  
          "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",  
          "ResourceType": "vpc"  
        }  
      ],  
      "Type": "static",  
      "State": "active"  
    }  
  ],  
  "AdditionalRoutesAvailable": false  
}
```



Untuk informasi selengkapnya, lihat [Tabel rute gateway transit](#) di Panduan Gerbang Transit.

- Untuk detail API, lihat [SearchTransitGatewayRoutes](#) di Referensi AWS CLI Perintah.

## send-diagnostic-interrupt

Contoh kode berikut menunjukkan cara menggunakan `send-diagnostic-interrupt`.

### AWS CLI

Untuk mengirim interupsi diagnostik

`send-diagnostic-interrupt` Contoh berikut mengirimkan interupsi diagnostik ke instance yang ditentukan.

```
aws ec2 send-diagnostic-interrupt \  
  --instance-id i-1234567890abcdef0
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SendDiagnosticInterrupt](#) di Referensi AWS CLI Perintah.

## start-instances

Contoh kode berikut menunjukkan cara menggunakan `start-instances`.

### AWS CLI

Untuk memulai instans Amazon EC2

Contoh ini memulai instans yang didukung Amazon EBS tertentu.

Perintah:

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{  
  "StartingInstances": [  
    {
```

```

    "InstanceId": "i-1234567890abcdef0",
    "CurrentState": {
      "Code": 0,
      "Name": "pending"
    },
    "PreviousState": {
      "Code": 80,
      "Name": "stopped"
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [Hentikan dan Mulai Instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [StartInstances](#) di Referensi AWS CLI Perintah.

## start-network-insights-access-scope-analysis

Contoh kode berikut menunjukkan cara menggunakan `start-network-insights-access-scope-analysis`.

### AWS CLI

Untuk memulai analisis cakupan akses Network Insights

`start-network-insights-access-scope-analysis` Contoh berikut memulai analisis lingkup di AWS akun Anda.

```

aws ec2 start-network-insights-access-scope-analysis \
  --region us-east-1 \
  --network-insights-access-scope-id nis-123456789111

```

Output:

```

{
  "NetworkInsightsAccessScopeAnalysis": {
    "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
    "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-east-1:123456789012:network-insights-access-scope-analysis/nisa-123456789222",
    "NetworkInsightsAccessScopeId": "nis-123456789111",
  }
}

```

```
    "Status": "running",
    "StartDate": "2022-01-26T00:47:06.814000+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Memulai Network Access Analyzer menggunakan AWS CLI di Network Access Analyzer Guide](#).

- Untuk detail API, lihat [StartNetworkInsightsAccessScopeAnalysis](#) di Referensi AWS CLI Perintah.

## start-network-insights-analysis

Contoh kode berikut menunjukkan cara menggunakan `start-network-insights-analysis`.

### AWS CLI

Untuk menganalisis jalur

`start-network-insights-analysis` Contoh berikut menganalisis jalur antara sumber dan tujuan. Untuk melihat hasil analisis jalur, gunakan `describe-network-insights-analyses` perintah.

```
aws ec2 start-network-insights-analysis \
  --network-insights-path-id nip-0b26f224f1d131fa8
```

Output:

```
{
  "NetworkInsightsAnalysis": {
    "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a",
    "NetworkInsightsAnalysisArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-analysis/nia-02207aa13eb480c7a",
    "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
    "StartDate": "2021-01-20T22:58:37.495Z",
    "Status": "running"
  }
}
```

Untuk informasi selengkapnya, lihat [Memulai menggunakan AWS CLI](#) di Panduan Reachability Analyzer.

- Untuk detail API, lihat [StartNetworkInsightsAnalysis](#) di Referensi AWS CLI Perintah.

## **start-vpc-endpoint-service-private-dns-verification**

Contoh kode berikut menunjukkan cara menggunakan `start-vpc-endpoint-service-private-dns-verification`.

### AWS CLI

Untuk memulai proses verifikasi DNS

`start-vpc-endpoint-service-private-dns-verification` Contoh berikut memulai proses verifikasi DNS untuk layanan endpoint yang ditentukan.

```
aws ec2 start-vpc-endpoint-service-private-dns-verification \  
  --service-id vpce-svc-071afff70666e61e0
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola nama DNS](#) di Panduan AWS PrivateLink Pengguna.

- Untuk detail API, lihat [StartVpcEndpointServicePrivateDnsVerification](#) di Referensi AWS CLI Perintah.

## **stop-instances**

Contoh kode berikut menunjukkan cara menggunakan `stop-instances`.

### AWS CLI

Contoh 1: Untuk menghentikan instans Amazon EC2

Contoh `stop-instances` berikut menghentikan instans yang didukung Amazon EBS tertentu.

```
aws ec2 stop-instances \  
  --instance-ids i-1234567890abcdef0
```

Output:

```
{
```

```
"StoppingInstances": [  
  {  
    "InstanceId": "i-1234567890abcdef0",  
    "CurrentState": {  
      "Code": 64,  
      "Name": "stopping"  
    },  
    "PreviousState": {  
      "Code": 16,  
      "Name": "running"  
    }  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Hentikan dan Mulai Instans Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

Contoh 2: Untuk menghibernasi instans Amazon EC2

Contoh `stop-instances` berikut menghibernasi instans yang didukung Amazon EBS jika instans diaktifkan untuk hibernasi dan memenuhi prasyarat hibernasi. Setelah instans dimasukkan ke hibernasi, instans berhenti.

```
aws ec2 stop-instances \  
  --instance-ids i-1234567890abcdef0 \  
  --hibernate
```

Output:

```
{  
  "StoppingInstances": [  
    {  
      "CurrentState": {  
        "Code": 64,  
        "Name": "stopping"  
      },  
      "InstanceId": "i-1234567890abcdef0",  
      "PreviousState": {  
        "Code": 16,  
        "Name": "running"  
      }  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Hibernasikan instans Linux Sesuai Permintaan Anda](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

- Untuk detail API, lihat [StopInstances](#) di Referensi AWS CLI Perintah.

## **terminate-client-vpn-connections**

Contoh kode berikut menunjukkan cara menggunakan `terminate-client-vpn-connections`.

### AWS CLI

Untuk mengakhiri koneksi ke titik akhir Client VPN

`terminate-client-vpn-connections` Contoh berikut mengakhiri koneksi yang ditentukan ke titik akhir Client VPN.

```
aws ec2 terminate-client-vpn-connections \  
  --client-vpn-endpoint-id vpn-endpoint-123456789123abcde \  
  --connection-id cvpn-connection-04edd76f5201e0cb8
```

### Output:

```
{  
  "ClientVpnEndpointId": "vpn-endpoint-123456789123abcde",  
  "ConnectionStatuses": [  
    {  
      "ConnectionId": "cvpn-connection-04edd76f5201e0cb8",  
      "PreviousStatus": {  
        "Code": "active"  
      },  
      "CurrentStatus": {  
        "Code": "terminating"  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Koneksi Klien](#) di Panduan Administrator AWS Client VPN.

- Untuk detail API, lihat [TerminateClientVpnConnections](#) di Referensi AWS CLI Perintah.

## terminate-instances

Contoh kode berikut menunjukkan cara menggunakan `terminate-instances`.

### AWS CLI

Untuk mengakhiri instans Amazon EC2

Contoh ini mengakhiri instans tertentu.

Perintah:

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
  "TerminatingInstances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Instans Amazon EC2](#) di Panduan Pengguna Antarmuka Baris Perintah AWS .

- Untuk detail API, lihat [TerminateInstances](#) di Referensi AWS CLI Perintah.

## unassign-ipv6-addresses

Contoh kode berikut menunjukkan cara menggunakan `unassign-ipv6-addresses`.

## AWS CLI

Untuk membatalkan penugasan alamat IPv6 dari antarmuka jaringan

Contoh ini membatalkan penetapan alamat IPv6 yang ditentukan dari antarmuka jaringan yang ditentukan.

Perintah:

```
aws ec2 unassign-ipv6-addresses --ipv6-addresses
2001:db8:1234:1a00:3304:8879:34cf:4071 --network-interface-id eni-23c49b68
```

Output:

```
{
  "NetworkInterfaceId": "eni-23c49b68",
  "UnassignedIpv6Addresses": [
    "2001:db8:1234:1a00:3304:8879:34cf:4071"
  ]
}
```

- Untuk detail API, lihat [UnassignIpv6Alamat](#) di Referensi AWS CLI Perintah.

## unassign-private-ip-addresses

Contoh kode berikut menunjukkan cara menggunakan `unassign-private-ip-addresses`.

### AWS CLI

Untuk membatalkan penetapan alamat IP pribadi sekunder dari antarmuka jaringan

Contoh ini membatalkan penetapan alamat IP pribadi yang ditentukan dari antarmuka jaringan yang ditentukan. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

Perintah:

```
aws ec2 unassign-private-ip-addresses --network-interface-id eni-e5aa89a3 --private-
ip-addresses 10.0.0.82
```

- Untuk detail API, lihat [UnassignPrivateIpAddresses](#) di Referensi AWS CLI Perintah.



## unassign-private-nat-gateway-address

Contoh kode berikut menunjukkan cara menggunakan `unassign-private-nat-gateway-address`.

### AWS CLI

Untuk membatalkan penetapan alamat IP pribadi dari gateway NAT pribadi Anda

`unassign-private-nat-gateway-address` Contoh berikut membatalkan penetapan alamat IP tertentu dari gateway NAT pribadi yang ditentukan.

```
aws ec2 unassign-private-nat-gateway-address \
  --nat-gateway-id nat-1234567890abcdef0 \
  --private-ip-addresses 10.0.20.197
```

### Output:

```
{
  "NatGatewayId": "nat-0ee3edd182361f662",
  "NatGatewayAddresses": [
    {
      "NetworkInterfaceId": "eni-0065a61b324d1897a",
      "PrivateIp": "10.0.20.197",
      "IsPrimary": false,
      "Status": "unassigning"
    }
  ]
}
```

Untuk informasi lebih lanjut, lihat [Gateway NAT](#) dalam Panduan Pengguna Amazon VPC.

- Untuk detail API, lihat [UnassignPrivateNatGatewayAddress](#) di Referensi AWS CLI Perintah.

## unmonitor-instances

Contoh kode berikut menunjukkan cara menggunakan `unmonitor-instances`.

### AWS CLI

Untuk menonaktifkan pemantauan yang mendetail untuk instans

Perintah contoh ini menonaktifkan pemantauan yang mendetail untuk instans tertentu.

Perintah:

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
  "InstanceMonitorings": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "Monitoring": {
        "State": "disabling"
      }
    }
  ]
}
```

- Untuk detail API, lihat [UnmonitorInstances](#) di Referensi AWS CLI Perintah.

## update-security-group-rule-descriptions-egress

Contoh kode berikut menunjukkan cara menggunakan `update-security-group-rule-descriptions-egress`.

AWS CLI

Untuk memperbarui deskripsi aturan grup keamanan keluar

`update-security-group-rule-descriptions-egress` Contoh berikut memperbarui deskripsi untuk aturan grup keamanan untuk port yang ditentukan dan rentang alamat IPv4. Deskripsi 'Outbound HTTP access to server 2' menggantikan deskripsi yang ada untuk aturan.

```
aws ec2 update-security-group-rule-descriptions-egress \
  --group-id sg-02f0d35a850ba727f \
  --ip-permissions
  IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=203.0.113.0/24,Description="Outbound
  HTTP access to server 2"}]
```

Output:

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Aturan grup keamanan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [UpdateSecurityGroupRuleDescriptionsEgress](#) di Referensi AWS CLI Perintah.

## update-security-group-rule-descriptions-ingress

Contoh kode berikut menunjukkan cara menggunakan `update-security-group-rule-descriptions-ingress`.

AWS CLI

Contoh 1: Untuk memperbarui deskripsi aturan grup keamanan masuk dengan sumber CIDR

`update-security-group-rule-descriptions-ingress` Contoh berikut memperbarui deskripsi untuk aturan grup keamanan untuk port yang ditentukan dan rentang alamat IPv4. Deskripsi 'SSH access from ABC office' menggantikan deskripsi yang ada untuk aturan.

```
aws ec2 update-security-group-rule-descriptions-ingress \
  --group-id sg-02f0d35a850ba727f \
  --ip-permissions
  IpProtocol=tcp,FromPort=22,ToPort=22,IpRanges='[{"CidrIp=203.0.113.0/16,Description="SSH
  access from corpnet"}]'
```

Output:

```
{
  "Return": true
}
```

Untuk informasi selengkapnya, lihat [Aturan grup keamanan](#) di Panduan Pengguna Amazon EC2.

Contoh 2: Untuk memperbarui deskripsi aturan grup keamanan masuk dengan sumber daftar awalan

`update-security-group-rule-descriptions-ingress` Contoh berikut memperbarui deskripsi untuk aturan grup keamanan untuk port dan daftar awalan yang ditentukan. Deskripsi 'SSH access from ABC office' menggantikan deskripsi yang ada untuk aturan.

```
aws ec2 update-security-group-rule-descriptions-ingress \  
  --group-id sg-02f0d35a850ba727f \  
  --ip-permissions  
  IpProtocol=tcp,FromPort=22,ToPort=22,PrefixListIds='[{"PrefixListId=pl-12345678,Description=  
  access from corpnet"}]'
```

Output:

```
{  
  "Return": true  
}
```

Untuk informasi selengkapnya, lihat [Aturan grup keamanan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [UpdateSecurityGroupRuleDescriptionsIngress](#) di Referensi AWS CLI Perintah.

## **withdraw-byoip-cidr**

Contoh kode berikut menunjukkan cara menggunakan `withdraw-byoip-cidr`.

AWS CLI

Untuk berhenti mengiklankan rentang alamat

`withdraw-byoip-cidr` Contoh berikut berhenti mengiklankan rentang alamat yang ditentukan.

```
aws ec2 withdraw-byoip-cidr  
  --cidr 203.0.113.25/24
```

Output:

```
{  
  "ByoipCidr": {  
    "Cidr": "203.0.113.25/24",  
    "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",  
    "State": "advertised"  }  
}
```

```
}  
}
```

- Untuk detail API, lihat [WithdrawByoipCidr](#) di Referensi AWS CLI Perintah.

## Contoh Amazon EC2 Instance Connect menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum menggunakan Instance Connect AWS Command Line Interface with Amazon EC2.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **send-ssh-public-key**

Contoh kode berikut menunjukkan cara menggunakan `send-ssh-public-key`.

### AWS CLI

Untuk mengirim kunci publik SSH ke sebuah instans

`send-ssh-public-key` Contoh berikut mengirimkan SSH kunci publik tertentu untuk contoh tertentu. Kuncinya digunakan untuk mengautentikasi pengguna yang ditentukan.

```
aws ec2-instance-connect send-ssh-public-key \  
  --instance-id i-1234567890abcdef0 \  
  --instance-os-user ec2-user \  
  --availability-zone us-east-2b \  
  --public-key-path /path/to/public-key
```

```
--ssh-public-key file://path/my-rsa-key.pub
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SendSshPublicKey](#) di Referensi AWS CLI Perintah.

## Contoh Amazon ECR menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon ECR.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-check-layer-availability**

Contoh kode berikut menunjukkan cara menggunakan `batch-check-layer-availability`.

AWS CLI

Untuk memeriksa ketersediaan lapisan

`batch-check-layer-availability` Contoh berikut memeriksa ketersediaan lapisan dengan intisari

`sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed`  
di `cluster-autoscaler` repositori.

```
aws ecr batch-check-layer-availability \  
  --repository-name cluster-autoscaler \  
  --sha256 sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

```
--layer-digests
sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

**Output:**

```
{
  "layers": [
    {
      "layerDigest":
"sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed",
      "layerAvailability": "AVAILABLE",
      "layerSize": 2777,
      "mediaType": "application/vnd.docker.container.image.v1+json"
    }
  ],
  "failures": []
}
```

- Untuk detail API, lihat [BatchCheckLayerAvailability](#) di Referensi AWS CLI Perintah.

**batch-delete-image**

Contoh kode berikut menunjukkan cara menggunakan `batch-delete-image`.

**AWS CLI**

Contoh 1: Untuk menghapus gambar

`batch-delete-image` Contoh berikut menghapus gambar dengan tag `precise` di repositori yang ditentukan dalam registri default untuk akun.

```
aws ecr batch-delete-image \
  --repository-name ubuntu \
  --image-ids imageTag=precise
```

**Output:**

```
{
  "failures": [],
  "imageIds": [
    {
      "imageTag": "precise",
```

```
        "imageDigest":  
        "sha256:19665f1e6d1e504117a1743c0a3d3753086354a38375961f2e665416ef4b1b2f"  
    }  
]  
}
```

Contoh 2: Untuk menghapus beberapa gambar

`batch-delete-image` berikut menghapus semua gambar yang ditandai dengan `prod` dan `team1` di repositori yang ditentukan.

```
aws ecr batch-delete-image \  
  --repository-name MyRepository \  
  --image-ids imageTag=prod imageTag=team1
```

Output:

```
{  
  "imageIds": [  
    {  
      "imageDigest": "sha256:123456789012",  
      "imageTag": "prod"  
    },  
    {  
      "imageDigest": "sha256:567890121234",  
      "imageTag": "team1"  
    }  
  ],  
  "failures": []  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [BatchDeleteImage](#) di Referensi AWS CLI Perintah.

## **batch-get-image**

Contoh kode berikut menunjukkan cara menggunakan `batch-get-image`.

### AWS CLI

Contoh 1: Untuk mendapatkan gambar



`batch-get-image` Contoh berikut mendapatkan gambar dengan tag `v1.13.6` dalam repositori yang disebut `cluster-autoscaler` dalam registri default untuk akun.

```
aws ecr batch-get-image \
  --repository-name cluster-autoscaler \
  --image-ids imageTag=v1.13.6
```

Output:

```
{
  "images": [
    {
      "registryId": "012345678910",
      "repositoryName": "cluster-autoscaler",
      "imageId": {
        "imageDigest":
"sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
        "imageTag": "v1.13.6"
      },
      "imageManifest": "{\n  \"schemaVersion\": 2,\n
\n  \"mediaType\": \"application/vnd.docker.distribution.manifest.v2+json
\n\",\n  \"config\": {\n    \"mediaType\": \"application/
vnd.docker.container.image.v1+json\",\n    \"size\": 2777,\n    \"digest
\": \"sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
\n\n  },\n  \"layers\": [\n    {\n      \"mediaType
\": \"application/vnd.docker.image.rootfs.diff.tar.gzip
\n\",\n      \"size\": 17743696,\n      \"digest\":
\"sha256:39fafc05754f195f134ca11ecdb1c9a691ab0848c697fffeb5a85f900caaf6e1\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 2565026,\n
      \"digest\":
\"sha256:8c8a779d3a537b767ae1091fe6e00c2590afd16767aa6096d1b318d75494819f
\n\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 28005981,\n
      \"digest\":
\"sha256:c44ba47496991c9982ee493b47fd25c252caabf2b4ae7dd679c9a27b6a3c8fb7\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 775,\n      \"digest
\": \"sha256:e2c388b44226544363ca007be7b896bcce1baebea04da23cbd165eac30be650f\"\n
    }\n  ]\n}"
    },
    "failures": []
  ]
}
```

```
}
```

Contoh 2: Untuk mendapatkan banyak gambar

`batch-get-image` Contoh berikut menampilkan rincian semua gambar yang ditandai dengan `prod` dan `team1` dalam repositori yang ditentukan.

```
aws ecr batch-get-image \  
  --repository-name MyRepository \  
  --image-ids imageTag=prod imageTag=team1
```

Output:

```
{  
  "images": [  
    {  
      "registryId": "123456789012",  
      "repositoryName": "MyRepository",  
      "imageId": {  
        "imageDigest": "sha256:123456789012",  
        "imageTag": "prod"  
      },  
      "imageManifest": "manifestExample1"  
    },  
    {  
      "registryId": "567890121234",  
      "repositoryName": "MyRepository",  
      "imageId": {  
        "imageDigest": "sha256:123456789012",  
        "imageTag": "team1"  
      },  
      "imageManifest": "manifestExample2"  
    }  
  ],  
  "failures": []  
}
```

Untuk informasi selengkapnya, lihat [Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [BatchGetImage](#) di Referensi AWS CLI Perintah.

## complete-layer-upload

Contoh kode berikut menunjukkan cara menggunakan `complete-layer-upload`.

### AWS CLI

Untuk menyelesaikan unggahan lapisan gambar

`complete-layer-upload` Contoh berikut melengkapi upload layer gambar ke `layer-test` repositori.

```
aws ecr complete-layer-upload \  
  --repository-name layer-test \  
  --upload-id 6cb64b8a-9378-0e33-2ab1-b780fab8a9e9 \  
  --layer-digests 6cb64b8a-9378-0e33-2ab1-  
b780fab8a9e9:48074e6d3a68b39aad8ccc002cdad912d4148c0f92b3729323e
```

Output:

```
{  
  "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9",  
  "layerDigest":  
    "sha256:9a77f85878aa1906f2020a0ecdf7a7e962d57e882250acd773383224b3fe9a02",  
  "repositoryName": "layer-test",  
  "registryId": "130757420319"  
}
```

- Untuk detail API, lihat [CompleteLayerUpload](#) di Referensi AWS CLI Perintah.

## create-repository

Contoh kode berikut menunjukkan cara menggunakan `create-repository`.

### AWS CLI

Contoh 1: Untuk membuat repositori

`create-repository` Contoh berikut membuat repositori di dalam namespace yang ditentukan dalam registri default untuk akun.

```
aws ecr create-repository \  

```

```
--repository-name project-a/nginx-web-app
```

Output:

```
{
  "repository": {
    "registryId": "123456789012",
    "repositoryName": "sample-repo",
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/project-a/nginx-web-app"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Repositori](#) di Panduan Pengguna Amazon ECR.

Contoh 2: Untuk membuat repositori yang dikonfigurasi dengan immutabilitas tag gambar

`create-repository` Contoh berikut membuat repositori dikonfigurasi untuk kekekalan tag dalam registri default untuk akun.

```
aws ecr create-repository \
  --repository-name sample-repo \
  --image-tag-mutability IMMUTABLE
```

Output:

```
{
  "repository": {
    "registryId": "123456789012",
    "repositoryName": "sample-repo",
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/sample-repo",
    "imageTagMutability": "IMMUTABLE"
  }
}
```

Untuk informasi selengkapnya, lihat [Mutabilitas Tag Gambar](#) di Panduan Pengguna Amazon ECR.

Contoh 3: Untuk membuat repositori yang dikonfigurasi dengan konfigurasi pemindaian

`create-repository` Contoh berikut membuat repositori yang dikonfigurasi untuk melakukan pemindaian kerentanan pada push gambar di registri default untuk akun.

```
aws ecr create-repository \  
  --repository-name sample-repo \  
  --image-scanning-configuration scanOnPush=true
```

Output:

```
{  
  "repository": {  
    "registryId": "123456789012",  
    "repositoryName": "sample-repo",  
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/sample-  
repo",  
    "imageScanningConfiguration": {  
      "scanOnPush": true  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Pemindaian Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [CreateRepository](#) di Referensi AWS CLI Perintah.

## delete-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-lifecycle-policy`.

AWS CLI

Untuk menghapus kebijakan siklus hidup untuk repositori

`delete-lifecycle-policy` Contoh berikut menghapus kebijakan siklus hidup untuk repositori.  
`hello-world`

```
aws ecr delete-lifecycle-policy \  
  --repository-name hello-world
```

Output:

```
{  
  "registryId": "012345678910",
```

```

    "repositoryName": "hello-world",
    "lifecyclePolicyText": "{\"rules\": [{\"rulePriority\": 1, \"description\": \"Remove
untagged images.\", \"selection\": {\"tagStatus\": \"untagged\", \"countType\":
\"sinceImagePushed\", \"countUnit\": \"days\", \"countNumber\": 10}, \"action\": {\"type
\": \"expire\"}}]}",
    "lastEvaluatedAt": 0.0
}

```

- Untuk detail API, lihat [DeleteLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## delete-repository-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-repository-policy`.

### AWS CLI

Untuk menghapus kebijakan repositori untuk repositori

`delete-repository-policy` Contoh berikut menghapus kebijakan repositori untuk repositori.  
`cluster-autoscaler`

```

aws ecr delete-repository-policy \
  --repository-name cluster-autoscaler

```

Output:

```

{
  "registryId": "012345678910",
  "repositoryName": "cluster-autoscaler",
  "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
  \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :
  \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage
  \", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}

```

- Untuk detail API, lihat [DeleteRepositoryPolicy](#) di Referensi AWS CLI Perintah.

## delete-repository

Contoh kode berikut menunjukkan cara menggunakan `delete-repository`.

## AWS CLI

Untuk menghapus repositori

`delete-repository` Contoh perintah berikut menghapus repositori yang ditentukan dalam registri default untuk akun. `--force` Bendera diperlukan jika repositori berisi gambar.

```
aws ecr delete-repository \  
  --repository-name ubuntu \  
  --force
```

Output:

```
{  
  "repository": {  
    "registryId": "123456789012",  
    "repositoryName": "ubuntu",  
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/ubuntu"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Repositori di Panduan Pengguna Amazon ECR](#).

- Untuk detail API, lihat [DeleteRepository](#) di Referensi AWS CLI Perintah.

## `describe-image-scan-findings`

Contoh kode berikut menunjukkan cara menggunakan `describe-image-scan-findings`.

### AWS CLI

Untuk menggambarkan temuan pemindaian untuk sebuah gambar

`describe-image-scan-findings` Contoh berikut mengembalikan temuan pemindaian gambar untuk gambar menggunakan intisari gambar di repositori tertentu dalam registri default untuk akun.

```
aws ecr describe-image-scan-findings \  
  --repository-name sample-repo \  
  --image-id  
  imageDigest=sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6
```

## Output:

```
{
  "imageScanFindings": {
    "findings": [
      {
        "name": "CVE-2019-5188",
        "description": "A code execution vulnerability exists in the directory rehashing functionality of E2fsprogs e2fsck 1.45.4. A specially crafted ext4 directory can cause an out-of-bounds write on the stack, resulting in code execution. An attacker can corrupt a partition to trigger this vulnerability.",
        "uri": "http://people.ubuntu.com/~ubuntu-security/cve/CVE-2019-5188",
        "severity": "MEDIUM",
        "attributes": [
          {
            "key": "package_version",
            "value": "1.44.1-1ubuntu1.1"
          },
          {
            "key": "package_name",
            "value": "e2fsprogs"
          },
          {
            "key": "CVSS2_VECTOR",
            "value": "AV:L/AC:L/Au:N/C:P/I:P/A:P"
          },
          {
            "key": "CVSS2_SCORE",
            "value": "4.6"
          }
        ]
      }
    ],
    "imageScanCompletedAt": 1579839105.0,
    "vulnerabilitySourceUpdatedAt": 1579811117.0,
    "findingSeverityCounts": {
      "MEDIUM": 1
    }
  },
  "registryId": "123456789012",
  "repositoryName": "sample-repo",
  "imageId": {
    "imageDigest":
"sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
  }
}
```



```
  },
  "imageScanStatus": {
    "status": "COMPLETE",
    "description": "The scan was completed successfully."
  }
}
```

Untuk informasi selengkapnya, lihat [Pemindaian Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [DescribeImageScanFindings](#) di Referensi AWS CLI Perintah.

## describe-images

Contoh kode berikut menunjukkan cara menggunakan `describe-images`.

### AWS CLI

Untuk menggambarkan gambar dalam repositori

`describe-images` Contoh berikut menampilkan detail tentang gambar di `cluster-autoscaler` repositori dengan tag `v1.13.6`

```
aws ecr describe-images \
  --repository-name cluster-autoscaler \
  --image-ids imageTag=v1.13.6
```

Output:

```
{
  "imageDetails": [
    {
      "registryId": "012345678910",
      "repositoryName": "cluster-autoscaler",
      "imageDigest":
"sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
      "imageTags": [
        "v1.13.6"
      ],
      "imageSizeInBytes": 48318255,
      "imagePushedAt": 1565128275.0
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [DescribeImages](#) di Referensi AWS CLI Perintah.

## describe-repositories

Contoh kode berikut menunjukkan cara menggunakan `describe-repositories`.

### AWS CLI

Untuk menggambarkan repositori dalam registri

Contoh ini menjelaskan repositori dalam registri default untuk akun.

Perintah:

```
aws ecr describe-repositories
```

Output:

```
{
  "repositories": [
    {
      "registryId": "012345678910",
      "repositoryName": "ubuntu",
      "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/ubuntu"
    },
    {
      "registryId": "012345678910",
      "repositoryName": "test",
      "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/test"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeRepositories](#) di Referensi AWS CLI Perintah.

## get-authorization-token

Contoh kode berikut menunjukkan cara menggunakan `get-authorization-token`.

## AWS CLI

Untuk mendapatkan token otorisasi untuk registri default Anda

`get-authorization-token` Contoh perintah berikut mendapatkan token otorisasi untuk registri default Anda.

```
aws ecr get-authorization-token
```

Output:

```
{
  "authorizationData": [
    {
      "authorizationToken": "QVdT0kN...",
      "expiresAt": 1448875853.241,
      "proxyEndpoint": "https://123456789012.dkr.ecr.us-west-2.amazonaws.com"
    }
  ]
}
```

- Untuk detail API, lihat [GetAuthorizationToken](#) di Referensi AWS CLI Perintah.

## `get-download-url-for-layer`

Contoh kode berikut menunjukkan cara menggunakan `get-download-url-for-layer`.

## AWS CLI

Untuk mendapatkan URL unduhan lapisan

`get-download-url-for-layer` Contoh berikut menampilkan URL unduhan lapisan dengan intisari

`sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed` di `cluster-autoscaler` repositori.

```
aws ecr get-download-url-for-layer \
  --repository-name cluster-autoscaler \
  --layer-digest
sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

**Output:**

```
{
  "downloadUrl": "https://prod-us-west-2-starport-layer-bucket.s3.us-
west-2.amazonaws.com/e501-012345678910-9cb60dc0-7284-5643-3987-
da6dac0465f0/04620aac-66a5-4167-8232-55ee7ef6d565?X-Amz-Algorithm=AWS4-HMAC-
SHA256&X-Amz-Date=20190814T220617Z&X-Amz-SignedHeaders=host&X-Amz-Expires=3600&X-
Amz-Credential=AKIA32P3D2JDNMVAJLGF%2F20190814%2Fus-west-2%2Fs3%2Faws4_request&X-
Amz-Signature=9161345894947a1672467a0da7a1550f2f7157318312fe4941b59976239c3337",
  "layerDigest":
  "sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed"
}
```

- Untuk detail API, lihat [GetDownloadUrlForLayer](#) di Referensi AWS CLI Perintah.

**get-lifecycle-policy-preview**

Contoh kode berikut menunjukkan cara menggunakan `get-lifecycle-policy-preview`.

**AWS CLI**

Untuk mengambil detail untuk pratinjau kebijakan siklus hidup

`get-lifecycle-policy-preview` Contoh berikut mengambil hasil pratinjau kebijakan siklus hidup untuk repositori yang ditentukan dalam registri default untuk akun.

**Perintah:**

```
aws ecr get-lifecycle-policy-preview \
  --repository-name "project-a/amazon-ecs-sample"
```

**Output:**

```
{
  "registryId": "012345678910",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n\n      \"rulePriority\": 1,\n      \"description\": \"Expire images older than 14\n      days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\n      \": \"days\",\n      \"countNumber\": 14\n    },\n  ],\n}
```

```

  \"action\": {\n
    \n
    \"type\": \"expire\"\n
  }\n
  }\n
  ]\n}\n",
  "status": "COMPLETE",
  "previewResults": [],
  "summary": {
    "expiringImageTotalCount": 0
  }
}

```

Untuk informasi selengkapnya, lihat [Kebijakan Siklus Hidup](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [GetLifecyclePolicyPreview](#) di Referensi AWS CLI Perintah.

## get-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `get-lifecycle-policy`.

### AWS CLI

Untuk mengambil kebijakan siklus hidup

`get-lifecycle-policy` Contoh berikut menampilkan rincian kebijakan siklus hidup untuk repositori yang ditentukan dalam registri default untuk akun.

```

aws ecr get-lifecycle-policy \
  --repository-name "project-a/amazon-ecs-sample"

```

Output:

```

{
  "registryId": "123456789012",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\": 1,\n      \"description\":\n        \"Expire images older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}",
  "lastEvaluatedAt": 1504295007.0
}

```

Untuk informasi selengkapnya, lihat [Kebijakan Siklus Hidup](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [GetLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## get-login-password

Contoh kode berikut menunjukkan cara menggunakan `get-login-password`.

### AWS CLI

Untuk mengambil kata sandi untuk mengautentikasi ke registri

Berikut ini `get-login-password` menampilkan kata sandi yang dapat Anda gunakan dengan klien kontainer pilihan Anda untuk mengautentikasi ke registri ECR Amazon mana pun yang dapat diakses oleh kepala sekolah IAM Anda.

```
aws ecr get-login-password
```

Output:

```
<password>
```

Untuk digunakan dengan CLI Docker, pipa output perintah `get-login-password` ke `docker login` perintah. Saat mengambil kata sandi, pastikan Anda menentukan Wilayah yang sama dengan registri ECR Amazon Anda.

```
aws ecr get-login-password \
  --region <region> \
| docker login \
  --username AWS \
  --password-stdin <aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Untuk informasi selengkapnya, lihat [Otentikasi Registri](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [GetLoginPassword](#) di Referensi AWS CLI Perintah.

## get-login

Contoh kode berikut menunjukkan cara menggunakan `get-login`.

### AWS CLI

Untuk mengambil perintah login Docker ke registri default Anda

Contoh ini mencetak perintah yang dapat Anda gunakan untuk masuk ke registri Amazon ECR default Anda.

**Perintah:**

```
aws ecr get-login
```

**Output:**

```
docker login -u AWS -p <password> -e none https://  
<aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Untuk masuk ke registri akun lain

Contoh ini mencetak satu atau beberapa perintah yang dapat Anda gunakan untuk masuk ke pendaftar Amazon ECR yang terkait dengan akun lain.

**Perintah:**

```
aws ecr get-login --registry-ids 012345678910 023456789012
```

**Output:**

```
docker login -u <username> -p <token-1> -e none <endpoint-1>  
docker login -u <username> -p <token-2> -e none <endpoint-2>
```

- Untuk detail API, lihat [GetLogin](#) di Referensi AWS CLI Perintah.

## get-repository-policy

Contoh kode berikut menunjukkan cara menggunakan `get-repository-policy`.

### AWS CLI

Untuk mengambil kebijakan repositori untuk repositori

`get-repository-policy` Contoh berikut menampilkan rincian tentang kebijakan repositori untuk repositori. `cluster-autoscaler`

```
aws ecr get-repository-policy \  
  --repository-name cluster-autoscaler
```

**Output:**

```
{
  "registryId": "012345678910",
  "repositoryName": "cluster-autoscaler",
  "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :\n    \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\",\n    \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
```

- Untuk detail API, lihat [GetRepositoryPolicy](#) di Referensi AWS CLI Perintah.

## initiate-layer-upload

Contoh kode berikut menunjukkan cara menggunakan `initiate-layer-upload`.

### AWS CLI

Untuk memulai unggahan lapisan gambar

`initiate-layer-upload` Contoh berikut memulai upload layer gambar ke `layer-test` repositori.

```
aws ecr initiate-layer-upload \
  --repository-name layer-test
```

Output:

```
{
  "partSize": 10485760,
  "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9"
}
```

- Untuk detail API, lihat [InitiateLayerUpload](#) di Referensi AWS CLI Perintah.

## list-images

Contoh kode berikut menunjukkan cara menggunakan `list-images`.

### AWS CLI

Untuk daftar gambar dalam repositori



`list-images` Contoh berikut menampilkan daftar gambar dalam `cluster-autoscaler` repositori.

```
aws ecr list-images \  
  --repository-name cluster-autoscaler
```

Output:

```
{  
  "imageIds": [  
    {  
      "imageDigest":  
"sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",  
      "imageTag": "v1.13.8"  
    },  
    {  
      "imageDigest":  
"sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",  
      "imageTag": "v1.13.7"  
    },  
    {  
      "imageDigest":  
"sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",  
      "imageTag": "v1.13.6"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListImages](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk repositori

`list-tags-for-resource` Contoh berikut menampilkan daftar tag yang terkait dengan `hello-world` repositori.

```
aws ecr list-tags-for-resource \  
  --repository-name cluster-autoscaler
```

```
--resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world
```

Output:

```
{
  "tags": [
    {
      "Key": "Stage",
      "Value": "Integ"
    }
  ]
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-image-scanning-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-image-scanning-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi pemindaian gambar untuk repositori

`put-image-scanning-configuration` Contoh berikut memperbarui konfigurasi pemindaian gambar untuk repositori yang ditentukan.

```
aws ecr put-image-scanning-configuration \
  --repository-name sample-repo \
  --image-scanning-configuration scanOnPush=true
```

Output:

```
{
  "registryId": "012345678910",
  "repositoryName": "sample-repo",
  "imageScanningConfiguration": {
    "scanOnPush": true
  }
}
```

Untuk informasi selengkapnya, lihat [Pemindaian Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [PutImageScanningConfiguration](#) di Referensi AWS CLI Perintah.

## put-image-tag-mutability

Contoh kode berikut menunjukkan cara menggunakan `put-image-tag-mutability`.

### AWS CLI

Untuk memperbarui pengaturan mutabilitas tag gambar untuk repositori

`put-image-tag-mutability` Contoh berikut mengkonfigurasi repositori yang ditentukan untuk kekekalan tag. Ini mencegah semua tag gambar dalam repositori ditimpa.

```
aws ecr put-image-tag-mutability \  
  --repository-name hello-repository \  
  --image-tag-mutability IMMUTABLE
```

Output:

```
{  
  "registryId": "012345678910",  
  "repositoryName": "sample-repo",  
  "imageTagMutability": "IMMUTABLE"  
}
```

Untuk informasi selengkapnya, lihat [Mutabilitas Tag Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [PutImageTagMutability](#) di Referensi AWS CLI Perintah.

## put-image

Contoh kode berikut menunjukkan cara menggunakan `put-image`.

### AWS CLI

Untuk menandai ulang gambar dengan manifestnya

`put-image` Contoh berikut membuat tag baru di `hello-world` repositori dengan manifest gambar yang ada.

```
aws ecr put-image \  
  --repository-name hello-world
```

```
--repository-name hello-world \  
--image-tag 2019.08 \  
--image-manifest file://hello-world.manifest.json
```

Isi dari `hello-world.manifest.json`:

```
{  
  "schemaVersion": 2,  
  "mediaType": "application/vnd.docker.distribution.manifest.v2+json",  
  "config": {  
    "mediaType": "application/vnd.docker.container.image.v1+json",  
    "size": 5695,  
    "digest":  
"sha256:cea5fe7701b7db3dd1c372f3cea6f43cdda444fcc488f530829145e426d8b980"  
  },  
  "layers": [  
    {  
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",  
      "size": 39096921,  
      "digest":  
"sha256:d8868e50ac4c7104d2200d42f432b661b2da8c1e417ccfae217e6a1e04bb9295"  
    },  
    {  
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",  
      "size": 57938,  
      "digest":  
"sha256:83251ac64627fc331584f6c498b3aba5badc01574e2c70b2499af3af16630eed"  
    },  
    {  
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",  
      "size": 423,  
      "digest":  
"sha256:589bba2f1b36ae56f0152c246e2541c5aa604b058febfcf2be32e9a304fec610"  
    },  
    {  
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",  
      "size": 680,  
      "digest":  
"sha256:d62ecaceda3964b735cdd2af613d6bb136a52c1da0838b2ff4b4dab4212bcb1c"  
    },  
    {  
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",  
      "size": 162,
```

```
    "digest":
"sha256:6d93b41cfc6bf0d2522b7cf61588de4cd045065b36c52bd3aec2ba0622b2b22b"
  },
  {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 28268840,
    "digest":
"sha256:6986b4d4c07932c680b3587f2eac8b0e013568c003cc23b04044628a5c5e599f"
  },
  {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 35369152,
    "digest":
"sha256:8c5ec60f10102dc8da0649d866c7c2f706e459d0bdc25c83ad2de86f4996c276"
  },
  {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 155,
    "digest":
"sha256:cde50b1c594539c5f67cbede9aef95c9ae321ccfb857f7b251b45b84198adc85"
  },
  {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 28737,
    "digest":
"sha256:2e102807ab72a73fc9abf53e8c50e421bdc337a0a8afcb242176edeec65977e4"
  },
  {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 190,
    "digest":
"sha256:fc379bbd5ed37808772bef016553a297356c59b8f134659e6ee4ecb563c2f5a7"
  },
  {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 28748,
    "digest":
"sha256:021db240dfccf5a1aff19507d17c0177e5888e518acf295b52204b1825e8b7ee"
  }
]
}
```

Output:

```

{
  "image": {
    "registryId": "130757420319",
    "repositoryName": "hello-world",
    "imageId": {
      "imageDigest":
"sha256:8ece96b74f87652876199d83bd107d0435a196133af383ac54cb82b6cc5283ae",
      "imageTag": "2019.08"
    },
    "imageManifest": "{\n  \"schemaVersion\": 2,\n  \"mediaType
\": \"application/vnd.docker.distribution.manifest.v2+json
\",,\n  \"config\": {\n    \"mediaType\": \"application/
vnd.docker.container.image.v1+json\",,\n    \"size\": 5695,\n    \"digest\":
\"sha256:cea5fe7701b7db3dd1c372f3cea6f43cdda444fcc488f530829145e426d8b980\"\n
  },\n  \"layers\": [\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 39096921,\n      \"digest
\": \"sha256:d8868e50ac4c7104d2200d42f432b661b2da8c1e417ccfae217e6a1e04bb9295\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 57938,\n      \"digest
\": \"sha256:83251ac64627fc331584f6c498b3aba5badc01574e2c70b2499af3af16630eed
\"\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 423,\n      \"digest\":
\"sha256:589bba2f1b36ae56f0152c246e2541c5aa604b058febfcf2be32e9a304fec610\"\n
    },\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",,\n
      \"size\": 680,\n      \"digest\":
\"sha256:d62ecaceda3964b735cdd2af613d6bb136a52c1da0838b2ff4b4dab4212bcb1c
\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 162,\n      \"digest
\": \"sha256:6d93b41cfc6bf0d2522b7cf61588de4cd045065b36c52bd3aec2ba0622b2b22b
\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 28268840,\n      \"digest
\": \"sha256:6986b4d4c07932c680b3587f2eac8b0e013568c003cc23b04044628a5c5e599f
\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 35369152,\n      \"digest
\": \"sha256:8c5ec60f10102dc8da0649d866c7c2f706e459d0bdc25c83ad2de86f4996c276\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n      \"size\": 155,\n      \"digest\":
\"sha256:cde50b1c594539c5f67cbede9aef95c9ae321ccfb857f7b251b45b84198adc85\"\n
    },\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",,\n
      \"size\": 28737,\n      \"digest\":
\"sha256:2e102807ab72a73fc9abf53e8c50e421bdc337a0a8afcb242176edeec65977e4\"\n
    },\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",,\n
      \"size\": 190,\n      \"digest\":

```

```

  \"sha256:fc379bbd5ed37808772bef016553a297356c59b8f134659e6ee4ecb563c2f5a7\"\\n    },
\\n    {\\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\\n      \"size\": 28748,\\n      \"digest\":
  \"sha256:021db240dfccf5a1aff19507d17c0177e5888e518acf295b52204b1825e8b7ee\"\\n
    }\\n  ]\\n}\\n"
  }
}

```

- Untuk detail API, lihat [PutImage](#) di Referensi AWS CLI Perintah.

## put-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `put-lifecycle-policy`.

### AWS CLI

Untuk membuat kebijakan siklus hidup

`put-lifecycle-policy` Contoh berikut membuat kebijakan siklus hidup untuk repositori yang ditentukan dalam registri default untuk akun.

```

aws ecr put-lifecycle-policy \
  --repository-name "project-a/amazon-ecs-sample" \
  --lifecycle-policy-text "file://policy.json"

```

Isi dari `policy.json`:

```

{
  "rules": [
    {
      "rulePriority": 1,
      "description": "Expire images older than 14 days",
      "selection": {
        "tagStatus": "untagged",
        "countType": "sinceImagePushed",
        "countUnit": "days",
        "countNumber": 14
      },
      "action": {
        "type": "expire"
      }
    }
  ]
}

```

```
]
}
```

**Output:**

```
{
  "registryId": "<aws_account_id>",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\"rules\": [{\"rulePriority\": 1, \"description\": \"Expire images older than 14 days\", \"selection\": {\"tagStatus\": \"untagged\", \"countType\": \"sinceImagePushed\", \"countUnit\": \"days\", \"countNumber\": 14}, \"action\": {\"type\": \"expire\"}}]}"
}
```

Untuk informasi selengkapnya, lihat [Kebijakan Siklus Hidup](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [PutLifecyclePolicy](#) di Referensi AWS CLI Perintah.

**set-repository-policy**

Contoh kode berikut menunjukkan cara menggunakan `set-repository-policy`.

**AWS CLI**

Untuk menetapkan kebijakan repositori untuk repositori

`set-repository-policy` Contoh berikut melampirkan kebijakan repositori yang terkandung dalam file ke repositori. `cluster-autoscaler`

```
aws ecr set-repository-policy \
  --repository-name cluster-autoscaler \
  --policy-text file://my-policy.json
```

Isi dari `my-policy.json`:

```
{
  "Version" : "2008-10-17",
  "Statement" : [
    {
      "Sid" : "allow public pull",
      "Effect" : "Allow",
      "Principal" : "*",
```



```

        "Action" : [
            "ecr:BatchCheckLayerAvailability",
            "ecr:BatchGetImage",
            "ecr:GetDownloadUrlForLayer"
        ]
    }
]
}

```

Output:

```

{
  "registryId": "012345678910",
  "repositoryName": "cluster-autoscaler",
  "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n\n    \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :\n    \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\n    \", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}

```

- Untuk detail API, lihat [SetRepositoryPolicy](#) di Referensi AWS CLI Perintah.

## start-image-scan

Contoh kode berikut menunjukkan cara menggunakan `start-image-scan`.

### AWS CLI

Untuk memulai pemindaian kerentanan gambar

`start-image-scan` Contoh berikut memulai pemindaian gambar dan ditentukan oleh intisari gambar di repositori yang ditentukan.

```

aws ecr start-image-scan \
  --repository-name sample-repo \
  --image-id
imageDigest=sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6

```

Output:

```

{

```

```
"registryId": "012345678910",
"repositoryName": "sample-repo",
"imageId": {
  "imageDigest":
"sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
},
"imageScanStatus": {
  "status": "IN_PROGRESS"
}
}
```

Untuk informasi selengkapnya, lihat [Pemindaian Gambar](#) di Panduan Pengguna Amazon ECR.

- Untuk detail API, lihat [StartImageScandi](#) Referensi AWS CLI Perintah.

## start-lifecycle-policy-preview

Contoh kode berikut menunjukkan cara menggunakan `start-lifecycle-policy-preview`.

### AWS CLI

Untuk membuat pratinjau kebijakan siklus hidup

`start-lifecycle-policy-preview` Contoh berikut membuat pratinjau kebijakan siklus hidup yang ditentukan oleh file JSON untuk repositori tertentu.

```
aws ecr start-lifecycle-policy-preview \
  --repository-name "project-a/amazon-ecs-sample" \
  --lifecycle-policy-text "file://policy.json"
```

Isi dari `policy.json`:

```
{
  "rules": [
    {
      "rulePriority": 1,
      "description": "Expire images older than 14 days",
      "selection": {
        "tagStatus": "untagged",
        "countType": "sinceImagePushed",
        "countUnit": "days",
        "countNumber": 14
      }
    },
  ],
}
```

```

        "action": {
            "type": "expire"
        }
    ]
}

```

### Output:

```

{
  "registryId": "012345678910",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n\n      \"rulePriority\": 1,\n      \"description\": \"Expire images older than 14\n      days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\n\n\": \"days\",\n        \"countNumber\": 14\n      },\n\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ],\n  \"status\": \"IN_PROGRESS\"
}

```

- Untuk detail API, lihat [StartLifecyclePolicyPreview](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai repositori

`tag-resource` Contoh berikut menetapkan tag dengan kunci `Stage` dan nilai `Integ` pada `hello-world` repositori.

```

aws ecr tag-resource \
  --resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world \
  --tags Key=Stage,Value=Integ

```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag repositori

`untag-resource` Contoh berikut menghapus tag dengan kunci `Stage` dari `hello-world` repositori.

```
aws ecr untag-resource \  
  --resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world \  
  --tag-keys Stage
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## upload-layer-part

Contoh kode berikut menunjukkan cara menggunakan `upload-layer-part`.

### AWS CLI

Untuk mengunggah bagian lapisan

Berikut ini `upload-layer-part` mengunggah bagian layer gambar ke `layer-test` repositori.

```
aws ecr upload-layer-part \  
  --repository-name layer-test \  
  --upload-id 6cb64b8a-9378-0e33-2ab1-b780fab8a9e9 \  
  --part-first-byte 0 \  
  --part-last-byte 8323314 \  
  --layer-part-blob file:///var/lib/docker/image/overlay2/layerdb/sha256/  
ff986b10a018b48074e6d3a68b39aad8ccc002cdad912d4148c0f92b3729323e/layer.b64
```

Output:

```
{  
  "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9",  
  "registryId": "012345678910",  
  "lastByteReceived": 8323314,
```

```
"repositoryName": "layer-test"
}
```

- Untuk detail API, lihat [UploadLayerPart](#) di Referensi AWS CLI Perintah.

## Contoh Amazon ECS menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan Amazon ECS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-capacity-provider**

Contoh kode berikut menunjukkan cara menggunakan `create-capacity-provider`.

### AWS CLI

Untuk membuat penyedia kapasitas

`create-capacity-provider` Contoh berikut membuat penyedia kapasitas yang menggunakan grup Auto Scaling bernama `myASG`, telah mengelola penskalaan dan perlindungan terminasi terkelola yang diaktifkan. Konfigurasi ini digunakan untuk auto scaling klaster Amazon ECS.

```
aws ecs create-capacity-provider \
  --name "MyCapacityProvider" \
  --auto-scaling-group-provider "autoScalingGroupArn=arn:aws:autoscaling:us-
east-1:123456789012:autoScalingGroup:57ffcb94-11f0-4d6d-
```

```
bf60-3bac5EXAMPLE:autoScalingGroupName/
MyASG,managedScaling={status=ENABLED,targetCapacity=100},managedTerminationProtection=ENABLED
```

### Output:

```
{
  "capacityProvider": {
    "capacityProviderArn": "arn:aws:ecs:us-east-1:123456789012:capacity-provider/
MyCapacityProvider",
    "name": "MyCapacityProvider",
    "status": "ACTIVE",
    "autoScalingGroupProvider": {
      "autoScalingGroupArn": "arn:aws:autoscaling:us-
east-1:132456789012:autoScalingGroup:57ffcb94-11f0-4d6d-
bf60-3bac5EXAMPLE:autoScalingGroupName/MyASG",
      "managedScaling": {
        "status": "ENABLED",
        "targetCapacity": 100,
        "minimumScalingStepSize": 1,
        "maximumScalingStepSize": 10000,
        "instanceWarmupPeriod": 300
      },
      "managedTerminationProtection": "ENABLED"
    },
    "tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [penskalaan otomatis kluster Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [CreateCapacityProvider](#) di Referensi AWS CLI Perintah.

## create-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-cluster`.

### AWS CLI

Contoh 1: Untuk membuat cluster baru

`create-cluster` Contoh berikut menciptakan sebuah cluster.

```
aws ecs create-cluster \
```

```
--cluster-name MyCluster
```

Output:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "pendingTasksCount": 0,
    "runningTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
    "tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Cluster](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk membuat cluster baru menggunakan penyedia kapasitas

`create-cluster` Contoh berikut membuat cluster dan mengaitkan dua penyedia kapasitas yang ada dengannya. `create-capacity-provider` Perintah ini digunakan untuk membuat penyedia kapasitas. Menentukan strategi penyedia kapasitas default adalah opsional, tetapi disarankan. Dalam contoh ini, kita membuat sebuah cluster bernama `MyCluster` dan mengasosiasikan `MyCapacityProvider1` dan penyedia `MyCapacityProvider2` kapasitas dengannya. Strategi penyedia kapasitas default ditentukan yang menyebarkan tugas secara merata di kedua penyedia kapasitas.

```
aws ecs create-cluster --cluster-name MyCluster --capacity-providers 1 2 -- CapacityProvider=
1, berat = 1 kapasitasPenyedia= 2, berat = MyCapacityProvider 1 MyCapacityProvider default-
capacity-provider-strategy MyCapacityProvider MyCapacityProvider
```

Output:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "PROVISIONING",
    "registeredContainerInstancesCount": 0,
```

```
"pendingTasksCount": 0,
"runningTasksCount": 0,
"activeServicesCount": 0,
"statistics": [],
"settings": [
  {
    "name": "containerInsights",
    "value": "enabled"
  }
],
"capacityProviders": [
  "MyCapacityProvider1",
  "MyCapacityProvider2"
],
"defaultCapacityProviderStrategy": [
  {
    "capacityProvider": "MyCapacityProvider1",
    "weight": 1,
    "base": 0
  },
  {
    "capacityProvider": "MyCapacityProvider2",
    "weight": 1,
    "base": 0
  }
],
"attachments": [
  {
    "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
    "type": "asp",
    "status": "PRECREATED",
    "details": [
      {
        "name": "capacityProviderName",
        "value": "MyCapacityProvider1"
      },
      {
        "name": "scalingPlanName",
        "value": "ECManagedAutoScalingPlan-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
      }
    ]
  }
],
{
```



```

        "id": "ae592060-2382-4663-9476-b015c685593c",
        "type": "asp",
        "status": "PRECREATED",
        "details": [
            {
                "name": "capacityProviderName",
                "value": "MyCapacityProvider2"
            },
            {
                "name": "scalingPlanName",
                "value": "ECSManagedAutoScalingPlan-a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222"
            }
        ]
    },
    "attachmentsStatus": "UPDATE_IN_PROGRESS"
}
}

```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas kluster](#) di Panduan Pengembang Amazon ECS.

Contoh 3: Untuk membuat cluster baru dengan beberapa tag

`create-cluster` Contoh berikut membuat cluster dengan beberapa tag. Untuk informasi selengkapnya tentang menambahkan tag menggunakan sintaks singkatan, lihat [Menggunakan Sintaks Shorthand dengan Antarmuka AWS Baris Perintah di Panduan Pengguna CLI.AWS](#)

```

aws ecs create-cluster \
  --cluster-name MyCluster \
  --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3

```

Output:

```

{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "pendingTasksCount": 0,
  }
}

```

```
    "runningTasksCount": 0,  
    "activeServicesCount": 0,  
    "statistics": [],  
    "tags": [  
      {  
        "key": "key1",  
        "value": "value1"  
      },  
      {  
        "key": "key2",  
        "value": "value2"  
      },  
      {  
        "key": "key3",  
        "value": "value3"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Cluster](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [CreateCluster](#) di Referensi AWS CLI Perintah.

## create-service

Contoh kode berikut menunjukkan cara menggunakan `create-service`.

### AWS CLI

Contoh 1: Untuk membuat layanan dengan tugas Fargate

`create-service` Contoh berikut menunjukkan cara membuat layanan menggunakan tugas Fargate.

```
aws ecs create-service \  
  --cluster MyCluster \  
  --service-name MyService \  
  --task-definition sample-fargate:1 \  
  --desired-count 2 \  
  --launch-type FARGATE \  
  --platform-version LATEST \  
  --tags Key=Value
```

```

--network-configuration
"awsvpcConfiguration={subnets=[subnet-12344321],securityGroups=[sg-12344321],assignPublicIp
\
--tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3

```

### Output:

```

{
  "service": {
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService",
    "serviceName": "MyService",
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "loadBalancers": [],
    "serviceRegistries": [],
    "status": "ACTIVE",
    "desiredCount": 2,
    "runningCount": 0,
    "pendingCount": 0,
    "launchType": "FARGATE",
    "platformVersion": "LATEST",
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/sample-fargate:1",
    "deploymentConfiguration": {
      "maximumPercent": 200,
      "minimumHealthyPercent": 100
    },
    "deployments": [
      {
        "id": "ecs-svc/1234567890123456789",
        "status": "PRIMARY",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/sample-fargate:1",
        "desiredCount": 2,
        "pendingCount": 0,
        "runningCount": 0,
        "createdAt": 1557119253.821,
        "updatedAt": 1557119253.821,
        "launchType": "FARGATE",
        "platformVersion": "1.3.0",
        "networkConfiguration": {
          "awsvpcConfiguration": {
            "subnets": [

```

```
        "subnet-12344321"
      ],
      "securityGroups": [
        "sg-12344321"
      ],
      "assignPublicIp": "ENABLED"
    }
  }
},
"roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
"events": [],
"createdAt": 1557119253.821,
"placementConstraints": [],
"placementStrategy": [],
"networkConfiguration": {
  "awsvpcConfiguration": {
    "subnets": [
      "subnet-12344321"
    ],
    "securityGroups": [
      "sg-12344321"
    ],
    "assignPublicIp": "ENABLED"
  }
},
"schedulingStrategy": "REPLICA",
"tags": [
  {
    "key": "key1",
    "value": "value1"
  },
  {
    "key": "key2",
    "value": "value2"
  },
  {
    "key": "key3",
    "value": "value3"
  }
],
"enableECSTags": false,
"propagateTags": "NONE"
```

```
}  
}
```

Contoh 2: Untuk membuat layanan menggunakan tipe peluncuran EC2

`create-service` Contoh berikut menunjukkan cara membuat layanan yang dipanggil `ecs-simple-service` dengan tugas yang menggunakan tipe peluncuran EC2. Layanan ini menggunakan definisi `sleep360` tugas dan mempertahankan 1 instantiasi tugas.

```
aws ecs create-service \  
  --cluster MyCluster \  
  --service-name ecs-simple-service \  
  --task-definition sleep360:2 \  
  --desired-count 1
```

Output:

```
{  
  "service": {  
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/ecs-  
simple-service",  
    "serviceName": "ecs-simple-service",  
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",  
    "loadBalancers": [],  
    "serviceRegistries": [],  
    "status": "ACTIVE",  
    "desiredCount": 1,  
    "runningCount": 0,  
    "pendingCount": 0,  
    "launchType": "EC2",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
sleep360:2",  
    "deploymentConfiguration": {  
      "maximumPercent": 200,  
      "minimumHealthyPercent": 100  
    },  
    "deployments": [  
      {  
        "id": "ecs-svc/1234567890123456789",  
        "status": "PRIMARY",  
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-  
definition/sleep360:2",
```

```

        "desiredCount": 1,
        "pendingCount": 0,
        "runningCount": 0,
        "createdAt": 1557206498.798,
        "updatedAt": 1557206498.798,
        "launchType": "EC2"
    }
],
"events": [],
"createdAt": 1557206498.798,
"placementConstraints": [],
"placementStrategy": [],
"schedulingStrategy": "REPLICA",
"enableECSTags": false,
"propagateTags": "NONE"
}
}

```

Contoh 3: Untuk membuat layanan yang menggunakan pengontrol penyebaran eksternal

`create-service` Contoh berikut membuat layanan yang menggunakan controller deployment eksternal.

```

aws ecs create-service \
  --cluster MyCluster \
  --service-name MyService \
  --deployment-controller type=EXTERNAL \
  --desired-count 1

```

Output:

```

{
  "service": {
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService",
    "serviceName": "MyService",
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "loadBalancers": [],
    "serviceRegistries": [],
    "status": "ACTIVE",
    "desiredCount": 1,
    "runningCount": 0,
    "pendingCount": 0,

```

```

    "launchType": "EC2",
    "deploymentConfiguration": {
      "maximumPercent": 200,
      "minimumHealthyPercent": 100
    },
    "taskSets": [],
    "deployments": [],
    "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
    "events": [],
    "createdAt": 1557128207.101,
    "placementConstraints": [],
    "placementStrategy": [],
    "schedulingStrategy": "REPLICA",
    "deploymentController": {
      "type": "EXTERNAL"
    },
    "enableECSTags": false,
    "propagateTags": "NONE"
  }
}

```

#### Contoh 4: Untuk membuat layanan baru di belakang penyeimbang beban

`create-service` Contoh berikut menunjukkan cara membuat layanan yang berada di belakang penyeimbang beban. Anda harus memiliki penyeimbang beban yang dikonfigurasi di Wilayah yang sama dengan instance container Anda. Contoh ini menggunakan `--cli-input-json` opsi dan file input JSON yang disebut `ecs-simple-service-elb.json` dengan konten berikut:

```

{
  "serviceName": "ecs-simple-service-elb",
  "taskDefinition": "ecs-demo",
  "loadBalancers": [
    {
      "loadBalancerName": "EC2Contai-EcsElast-123456789012",
      "containerName": "simple-demo",
      "containerPort": 80
    }
  ],
  "desiredCount": 10,
  "role": "ecsServiceRole"
}

```

**Perintah:**

```
aws ecs create-service \  
  --cluster MyCluster \  
  --service-name ecs-simple-service-elb \  
  --cli-input-json file://ecs-simple-service-elb.json
```

**Output:**

```
{  
  "service": {  
    "status": "ACTIVE",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/ecs-  
demo:1",  
    "pendingCount": 0,  
    "loadBalancers": [  
      {  
        "containerName": "ecs-demo",  
        "containerPort": 80,  
        "loadBalancerName": "EC2Contai-EcsElast-123456789012"  
      }  
    ],  
    "roleArn": "arn:aws:iam::123456789012:role/ecsServiceRole",  
    "desiredCount": 10,  
    "serviceName": "ecs-simple-service-elb",  
    "clusterArn": "arn:aws:ecs:<us-west-2:123456789012:cluster/MyCluster",  
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/ecs-simple-  
service-elb",  
    "deployments": [  
      {  
        "status": "PRIMARY",  
        "pendingCount": 0,  
        "createdAt": 1428100239.123,  
        "desiredCount": 10,  
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-  
definition/ecs-demo:1",  
        "updatedAt": 1428100239.123,  
        "id": "ecs-svc/1234567890123456789",  
        "runningCount": 0  
      }  
    ],  
    "events": [],  
    "runningCount": 0  
  }  
}
```



```
}  
}
```

Untuk informasi selengkapnya, lihat [Membuat Layanan](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [CreateService](#) di Referensi AWS CLI Perintah.

## create-task-set

Contoh kode berikut menunjukkan cara menggunakan `create-task-set`.

### AWS CLI

Untuk membuat set tugas

`create-task-set` Contoh berikut membuat tugas yang ditetapkan dalam layanan yang menggunakan controller deployment eksternal.

```
aws ecs create-task-set \  
  --cluster MyCluster \  
  --service MyService \  
  --task-definition MyTaskDefinition:2 \  
  --network-configuration  
  "awsvpcConfiguration={subnets=[subnet-12344321],securityGroups=[sg-12344321]}"
```

Output:

```
{  
  "taskSet": {  
    "id": "ecs-svc/1234567890123456789",  
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
    "status": "ACTIVE",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
MyTaskDefinition:2",  
    "computedDesiredCount": 0,  
    "pendingCount": 0,  
    "runningCount": 0,  
    "createdAt": 1557128360.711,  
    "updatedAt": 1557128360.711,  
    "launchType": "EC2",  
    "networkConfiguration": {
```

```
    "awsvpcConfiguration": {
      "subnets": [
        "subnet-12344321"
      ],
      "securityGroups": [
        "sg-12344321"
      ],
      "assignPublicIp": "DISABLED"
    }
  },
  "loadBalancers": [],
  "serviceRegistries": [],
  "scale": {
    "value": 0.0,
    "unit": "PERCENT"
  },
  "stabilityStatus": "STABILIZING",
  "stabilityStatusAt": 1557128360.711
}
}
```

- Untuk detail API, lihat [CreateTaskSet](#) di Referensi AWS CLI Perintah.

## delete-account-setting

Contoh kode berikut menunjukkan cara menggunakan `delete-account-setting`.

### AWS CLI

Untuk menghapus pengaturan akun untuk pengguna IAM tertentu atau peran IAM

Contoh berikut `delete-account-setting` menghapus pengaturan akun untuk pengguna IAM tertentu atau peran IAM.

```
aws ecs delete-account-setting \
  --name serviceLongArnFormat \
  --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Output:

```
{
```

```
"setting": {
  "name": "serviceLongArnFormat",
  "value": "enabled",
  "principalArn": "arn:aws:iam::123456789012:user/MyUser"
}
}
```

Untuk informasi selengkapnya, lihat [Nama Sumber Daya Amazon \(ARN\) dan ID](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DeleteAccountSetting](#) di Referensi AWS CLI Perintah.

## delete-attributes

Contoh kode berikut menunjukkan cara menggunakan `delete-attributes`.

### AWS CLI

Untuk menghapus satu atau beberapa atribut kustom dari sumber daya Amazon ECS

Berikut ini `delete-attributes` menghapus atribut dengan nama `stack` dari instance container.

```
aws ecs delete-attributes \
  --attributes name=stack,targetId=arn:aws:ecs:us-west-2:130757420319:container-
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34
```

Output:

```
{
  "attributes": [
    {
      "name": "stack",
      "targetId": "arn:aws:ecs:us-west-2:130757420319:container-
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",
      "value": "production"
    }
  ]
}
```

- Untuk detail API, lihat [DeleteAttributes](#) di Referensi AWS CLI Perintah.

## delete-capacity-provider

Contoh kode berikut menunjukkan cara menggunakan `delete-capacity-provider`.

### AWS CLI

Contoh 1: Untuk menghapus penyedia kapasitas menggunakan Amazon Resource Name (ARN)

`delete-capacity-provider` Contoh berikut menghapus penyedia kapasitas dengan menentukan Amazon Resource Name (ARN) dari penyedia kapasitas. ARN serta status penghapusan penyedia kapasitas dapat diambil menggunakan perintah `describe-capacity-providers`

```
aws ecs delete-capacity-provider \
  --capacity-provider arn:aws:ecs:us-west-2:123456789012:capacity-provider/
  ExampleCapacityProvider
```

### Output:

```
{
  "capacityProvider": {
    "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
    provider/ExampleCapacityProvider",
    "name": "ExampleCapacityProvider",
    "status": "ACTIVE",
    "autoScalingGroupProvider": {
      "autoScalingGroupArn": "arn:aws:autoscaling:us-
      west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
      EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
      "managedScaling": {
        "status": "ENABLED",
        "targetCapacity": 100,
        "minimumScalingStepSize": 1,
        "maximumScalingStepSize": 10000
      },
      "managedTerminationProtection": "DISABLED"
    },
    "updateStatus": "DELETE_IN_PROGRESS",
    "tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas klaster](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk menghapus penyedia kapasitas menggunakan nama

`delete-capacity-provider` Contoh berikut menghapus penyedia kapasitas dengan menentukan nama pendek penyedia kapasitas. Nama pendek serta status penghapusan penyedia kapasitas dapat diambil menggunakan perintah `describe-capacity-providers`

```
aws ecs delete-capacity-provider \  
  --capacity-provider ExampleCapacityProvider
```

Output:

```
{  
  "capacityProvider": {  
    "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-  
provider/ExampleCapacityProvider",  
    "name": "ExampleCapacityProvider",  
    "status": "ACTIVE",  
    "autoScalingGroupProvider": {  
      "autoScalingGroupArn": "arn:aws:autoscaling:us-  
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",  
      "managedScaling": {  
        "status": "ENABLED",  
        "targetCapacity": 100,  
        "minimumScalingStepSize": 1,  
        "maximumScalingStepSize": 10000  
      },  
      "managedTerminationProtection": "DISABLED"  
    },  
    "updateStatus": "DELETE_IN_PROGRESS",  
    "tags": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas klaster](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DeleteCapacityProvider](#) di Referensi AWS CLI Perintah.

## delete-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster`.

### AWS CLI

Untuk menghapus cluster kosong

`delete-cluster` Contoh berikut menghapus cluster kosong yang ditentukan.

```
aws ecs delete-cluster --cluster MyCluster
```

Output:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "status": "INACTIVE",
    "clusterName": "MyCluster",
    "registeredContainerInstancesCount": 0,
    "pendingTasksCount": 0,
    "runningTasksCount": 0,
    "activeServicesCount": 0
    "statistics": [],
    "tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus Cluster](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DeleteCluster](#) di Referensi AWS CLI Perintah.

## delete-service

Contoh kode berikut menunjukkan cara menggunakan `delete-service`.

### AWS CLI

Untuk menghapus layanan

`ecs delete-service` Contoh berikut menghapus layanan tertentu dari cluster. Anda dapat menyertakan `--force` parameter untuk menghapus layanan meskipun belum diskalakan ke nol tugas.

```
aws ecs delete-service --cluster MyCluster --service MyService1 --force
```

Untuk informasi selengkapnya, lihat [Menghapus Layanan](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DeleteService](#) di Referensi AWS CLI Perintah.

## delete-task-definitions

Contoh kode berikut menunjukkan cara menggunakan `delete-task-definitions`.

### AWS CLI

Untuk menghapus definisi tugas

`delete-task-definitions` Contoh berikut menghapus definisi tugas INACTIVE.

```
aws ecs delete-task-definitions \  
  --task-definition curltest:1
```

Output:

```
{  
  "taskDefinitions": [  
    {  
      "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/  
curltest:1",  
      "containerDefinitions": [  
        {  
          "name": "ctest",  
          "image": "mreferre/eksutils",  
          "cpu": 0,  
          "portMappings": [],  
          "essential": true,  
          "entryPoint": [  
            "sh",  
            "-c"  
          ],  
          "command": [  
            "curl ${ECS_CONTAINER_METADATA_URI_V4}/task"  
          ],  
        }  
      ]  
    }  
  ]  
}
```

```
    "environment": [],
    "mountPoints": [],
    "volumesFrom": [],
    "logConfiguration": {
      "logDriver": "awslogs",
      "options": {
        "awslogs-create-group": "true",
        "awslogs-group": "/ecs/curltest",
        "awslogs-region": "us-east-1",
        "awslogs-stream-prefix": "ecs"
      }
    }
  ],
  "family": "curltest",
  "taskRoleArn": "arn:aws:iam::123456789012:role/ecsTaskExecutionRole",
  "executionRoleArn": "arn:aws:iam::123456789012:role/ecsTaskExecutionRole",
  "networkMode": "awsvpc",
  "revision": 1,
  "volumes": [],
  "status": "DELETE_IN_PROGRESS",
  "compatibilities": [
    "EC2",
    "FARGATE"
  ],
  "requiresCompatibilities": [
    "FARGATE"
  ],
  "cpu": "256",
  "memory": "512",
  "registeredAt": "2021-09-10T12:56:24.704000+00:00",
  "deregisteredAt": "2023-03-14T15:20:59.419000+00:00",
  "registeredBy": "arn:aws:sts::123456789012:assumed-role/Admin/jdoe"
}
],
"failures": []
}
```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DeleteTaskDefinitions](#) di Referensi AWS CLI Perintah.



## delete-task-set

Contoh kode berikut menunjukkan cara menggunakan `delete-task-set`.

### AWS CLI

Untuk menghapus set tugas

`delete-task-set` Contoh berikut menunjukkan cara menghapus set tugas. Anda dapat menyertakan `--force` parameter untuk menghapus set tugas meskipun belum diskalakan ke nol.

```
aws ecs delete-task-set \
  --cluster MyCluster \
  --service MyService \
  --task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-
svc/1234567890123456789 \
  --force
```

Output:

```
{
  "taskSet": {
    "id": "ecs-svc/1234567890123456789",
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
    "status": "DRAINING",
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:2",
    "computedDesiredCount": 0,
    "pendingCount": 0,
    "runningCount": 0,
    "createdAt": 1557130260.276,
    "updatedAt": 1557130290.707,
    "launchType": "EC2",
    "networkConfiguration": {
      "awsvpcConfiguration": {
        "subnets": [
          "subnet-12345678"
        ],
        "securityGroups": [
          "sg-12345678"
        ],
        "assignPublicIp": "DISABLED"
      }
    }
  }
}
```

```

    }
  },
  "loadBalancers": [],
  "serviceRegistries": [],
  "scale": {
    "value": 0.0,
    "unit": "PERCENT"
  },
  "stabilityStatus": "STABILIZING",
  "stabilityStatusAt": 1557130290.707
}
}

```

- Untuk detail API, lihat [DeleteTaskSet](#) di Referensi AWS CLI Perintah.

## deregister-container-instance

Contoh kode berikut menunjukkan cara menggunakan `deregister-container-instance`.

### AWS CLI

Untuk membatalkan pendaftaran instance kontainer dari sebuah cluster

`deregister-container-instance` Contoh berikut deregisters instance container dari cluster tertentu. Jika masih ada tugas yang berjalan di instance container, Anda harus menghentikan tugas tersebut sebelum membatalkan pendaftaran, atau menggunakan opsi tersebut. `--force`

```

aws ecs deregister-container-instance \
  --cluster arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
  --container-instance arn:aws:ecs:us-west-2:123456789012:container-instance/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
  --force

```

Output:

```

{
  "containerInstance": {
    "remainingResources": [
      {
        "integerValue": 1024,
        "doubleValue": 0.0,
        "type": "INTEGER",

```

```
        "longValue": 0,
        "name": "CPU"
    },
    {
        "integerValue": 985,
        "doubleValue": 0.0,
        "type": "INTEGER",
        "longValue": 0,
        "name": "MEMORY"
    },
    {
        "type": "STRINGSET",
        "integerValue": 0,
        "name": "PORTS",
        "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678",
            "51679"
        ],
        "longValue": 0,
        "doubleValue": 0.0
    },
    {
        "type": "STRINGSET",
        "integerValue": 0,
        "name": "PORTS_UDP",
        "stringSetValue": [],
        "longValue": 0,
        "doubleValue": 0.0
    }
],
"agentConnected": true,
"attributes": [
    {
        "name": "ecs.capability.secrets.asm.environment-variables"
    },
    {
        "name": "com.amazonaws.ecs.capability.logging-driver.syslog"
    },
    {
        "value": "ami-01a82c3fce2c3ba58",
        "name": "ecs.ami-id"
    }
]
```

```
},
{
  "name": "ecs.capability.secrets.asm.bootstrap.log-driver"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.none"
},
{
  "name": "ecs.capability.ecr-endpoint"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.json-file"
},
{
  "value": "vpc-1234567890123467",
  "name": "ecs.vpc-id"
},
{
  "name": "ecs.capability.execution-role-awslogs"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.17"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
},
{
  "name": "ecs.capability.docker-plugin.local"
},
{
  "name": "ecs.capability.task-eni"
},
{
  "name": "ecs.capability.task-cpu-mem-limit"
},
{
  "name": "ecs.capability.secrets.ssm.bootstrap.log-driver"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.30"
},
},
```

```
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.31"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.32"
},
{
  "name": "ecs.capability.execution-role-ecr-pull"
},
{
  "name": "ecs.capability.container-health-check"
},
{
  "value": "subnet-1234567890123467",
  "name": "ecs.subnet-id"
},
{
  "value": "us-west-2a",
  "name": "ecs.availability-zone"
},
{
  "value": "t2.micro",
  "name": "ecs.instance-type"
},
{
  "name": "com.amazonaws.ecs.capability.task-iam-role-network-host"
},
{
  "name": "ecs.capability.aws-appmesh"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.24"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.25"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.26"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.27"
```

```
},
{
  "name": "com.amazonaws.ecs.capability.privileged-container"
},
{
  "name": "ecs.capability.container-ordering"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.28"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.29"
},
{
  "value": "x86_64",
  "name": "ecs.cpu-architecture"
},
{
  "value": "93f43776-2018.10.0",
  "name": "ecs.capability.cni-plugin-version"
},
{
  "name": "ecs.capability.secrets.ssm.environment-variables"
},
{
  "name": "ecs.capability.pid-ipc-namespace-sharing"
},
{
  "name": "com.amazonaws.ecs.capability.ecr-auth"
},
{
  "value": "linux",
  "name": "ecs.os-type"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.20"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.21"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.22"
},
{
```

```
        "name": "ecs.capability.task-eia"
      },
      {
        "name": "ecs.capability.private-registry-
authentication.secretsmanager"
      },
      {
        "name": "com.amazonaws.ecs.capability.task-iam-role"
      },
      {
        "name": "com.amazonaws.ecs.capability.docker-remote-api.1.23"
      }
    ],
    "pendingTasksCount": 0,
    "tags": [],
    "containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "registeredResources": [
      {
        "integerValue": 1024,
        "doubleValue": 0.0,
        "type": "INTEGER",
        "longValue": 0,
        "name": "CPU"
      },
      {
        "integerValue": 985,
        "doubleValue": 0.0,
        "type": "INTEGER",
        "longValue": 0,
        "name": "MEMORY"
      },
      {
        "type": "STRINGSET",
        "integerValue": 0,
        "name": "PORTS",
        "stringSetValue": [
          "22",
          "2376",
          "2375",
          "51678",
          "51679"
        ],
        "longValue": 0,

```

```
        "doubleValue": 0.0
      },
      {
        "type": "STRINGSET",
        "integerValue": 0,
        "name": "PORTS_UDP",
        "stringSetValue": [],
        "longValue": 0,
        "doubleValue": 0.0
      }
    ],
    "status": "INACTIVE",
    "registeredAt": 1557768075.681,
    "version": 4,
    "versionInfo": {
      "agentVersion": "1.27.0",
      "agentHash": "aabe65ee",
      "dockerVersion": "DockerVersion: 18.06.1-ce"
    },
    "attachments": [],
    "runningTasksCount": 0,
    "ec2InstanceId": "i-12345678901234678"
  }
}
```

Untuk informasi selengkapnya, lihat [Membatalkan pendaftaran Instans Kontainer di Panduan Pengembang ECS](#).

- Untuk detail API, lihat [DeregisterContainerInstance](#) di Referensi AWS CLI Perintah.

## deregister-task-definition

Contoh kode berikut menunjukkan cara menggunakan `deregister-task-definition`.

### AWS CLI

Untuk membatalkan pendaftaran definisi tugas

`deregister-task-definition` Contoh berikut membatalkan pendaftaran revisi pertama definisi `curler` tugas di wilayah default Anda.

```
aws ecs deregister-task-definition --task-definition curler:1
```



Perhatikan bahwa dalam output yang dihasilkan, status definisi tugas menunjukkan `INACTIVE`:

```
{
  "taskDefinition": {
    "status": "INACTIVE",
    "family": "curler",
    "volumes": [],
    "taskDefinitionArn": "arn:aws:ecs:us-west-2:123456789012:task-definition/
curler:1",
    "containerDefinitions": [
      {
        "environment": [],
        "name": "curler",
        "mountPoints": [],
        "image": "curl:latest",
        "cpu": 100,
        "portMappings": [],
        "entryPoint": [],
        "memory": 256,
        "command": [
          "curl -v http://example.com/"
        ],
        "essential": true,
        "volumesFrom": []
      }
    ],
    "revision": 1
  }
}
```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DeregisterTaskDefinition](#) di Referensi AWS CLI Perintah.

## **describe-capacity-providers**

Contoh kode berikut menunjukkan cara menggunakan `describe-capacity-providers`.

### AWS CLI

Contoh 1: Untuk menggambarkan semua penyedia kapasitas

`describe-capacity-providers` Contoh berikut mengambil rincian tentang semua penyedia kapasitas.

```
aws ecs describe-capacity-providers
```

Output:

```
{
  "capacityProviders": [
    {
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/MyCapacityProvider",
      "name": "MyCapacityProvider",
      "status": "ACTIVE",
      "autoScalingGroupProvider": {
        "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
        "managedScaling": {
          "status": "ENABLED",
          "targetCapacity": 100,
          "minimumScalingStepSize": 1,
          "maximumScalingStepSize": 1000
        },
        "managedTerminationProtection": "ENABLED"
      },
      "tags": []
    },
    {
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/FARGATE",
      "name": "FARGATE",
      "status": "ACTIVE",
      "tags": []
    },
    {
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/FARGATE_SPOT",
      "name": "FARGATE_SPOT",
      "status": "ACTIVE",
      "tags": []
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas klaster](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk menggambarkan penyedia kapasitas tertentu

`describe-capacity-providers` Contoh berikut mengambil rincian tentang penyedia kapasitas tertentu. Menggunakan `--include TAGS` parameter akan menambahkan tag yang terkait dengan penyedia kapasitas ke output.

```
aws ecs describe-capacity-providers \  
  --capacity-providers MyCapacityProvider \  
  --include TAGS
```

Output:

```
{  
  "capacityProviders": [  
    {  
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-  
provider/MyCapacityProvider",  
      "name": "MyCapacityProvider",  
      "status": "ACTIVE",  
      "autoScalingGroupProvider": {  
        "autoScalingGroupArn": "arn:aws:autoscaling:us-  
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",  
        "managedScaling": {  
          "status": "ENABLED",  
          "targetCapacity": 100,  
          "minimumScalingStepSize": 1,  
          "maximumScalingStepSize": 1000  
        },  
        "managedTerminationProtection": "ENABLED"  
      },  
      "tags": [  
        {  
          "key": "environment",  
          "value": "production"  
        }  
      ]  
    }  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas klaster](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DescribeCapacityProviders](#) di Referensi AWS CLI Perintah.

## describe-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-clusters`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan sebuah cluster

`describe-clusters` Contoh berikut mengambil rincian tentang cluster tertentu.

```
aws ecs describe-clusters \
  --cluster default
```

Output:

```
{
  "clusters": [
    {
      "status": "ACTIVE",
      "clusterName": "default",
      "registeredContainerInstancesCount": 0,
      "pendingTasksCount": 0,
      "runningTasksCount": 0,
      "activeServicesCount": 1,
      "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default"
    }
  ],
  "failures": []
}
```

Untuk informasi selengkapnya, lihat [Cluster Amazon ECS](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk mendeskripsikan cluster dengan opsi lampiran

`describe-clusters` Contoh berikut menentukan pilihan ATTACHMENTS. Ini mengambil rincian tentang cluster yang ditentukan dan daftar sumber daya yang melekat pada cluster dalam bentuk lampiran. Saat menggunakan penyedia kapasitas dengan klaster, sumber daya, baik AutoScaling rencana atau kebijakan penskalaan, akan direpresentasikan sebagai `asp` atau `as_policy` ATTACHMENTS.

```
aws ecs describe-clusters \  
  --include ATTACHMENTS \  
  --clusters sampleCluster
```

Output:

```
{  
  "clusters": [  
    {  
      "clusterArn": "arn:aws:ecs:af-south-1:123456789222:cluster/  
sampleCluster",  
      "clusterName": "sampleCluster",  
      "status": "ACTIVE",  
      "registeredContainerInstancesCount": 0,  
      "runningTasksCount": 0,  
      "pendingTasksCount": 0,  
      "activeServicesCount": 0,  
      "statistics": [],  
      "tags": [],  
      "settings": [],  
      "capacityProviders": [  
        "sampleCapacityProvider"  
      ],  
      "defaultCapacityProviderStrategy": [],  
      "attachments": [  
        {  
          "id": "a1b2c3d4-5678-901b-cdef-EXAMPLE22222",  
          "type": "as_policy",  
          "status": "CREATED",  
          "details": [  
            {  
              "name": "capacityProviderName",  
              "value": "sampleCapacityProvider"  
            },  
            {  
              "name": "scalingPolicyName",
```

```

        "value": "ECManagedAutoScalingPolicy-3048e262-
fe39-4eaf-826d-6f975d303188"
      }
    ]
  },
  "attachmentsStatus": "UPDATE_COMPLETE"
}
],
"failures": []
}

```

Untuk informasi selengkapnya, lihat [Cluster Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DescribeClusters](#) di Referensi AWS CLI Perintah.

## describe-container-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-container-instances`.

### AWS CLI

Untuk menggambarkan contoh kontainer

`describe-container-instances` Contoh berikut mengambil rincian untuk instance container di update cluster, menggunakan container instance UUID sebagai identifier.

```

aws ecs describe-container-instances \
  --cluster update \
  --container-instances a1b2c3d4-5678-90ab-cdef-11111EXAMPLE

```

Output:

```

{
  "failures": [],
  "containerInstances": [
    {
      "status": "ACTIVE",
      "registeredResources": [
        {
          "integerValue": 2048,
          "longValue": 0,

```

```
        "type": "INTEGER",
        "name": "CPU",
        "doubleValue": 0.0
    },
    {
        "integerValue": 3955,
        "longValue": 0,
        "type": "INTEGER",
        "name": "MEMORY",
        "doubleValue": 0.0
    },
    {
        "name": "PORTS",
        "longValue": 0,
        "doubleValue": 0.0,
        "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678"
        ],
        "type": "STRINGSET",
        "integerValue": 0
    }
],
"ec2InstanceId": "i-A1B2C3D4",
"agentConnected": true,
"containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
"pendingTasksCount": 0,
"remainingResources": [
    {
        "integerValue": 2048,
        "longValue": 0,
        "type": "INTEGER",
        "name": "CPU",
        "doubleValue": 0.0
    },
    {
        "integerValue": 3955,
        "longValue": 0,
        "type": "INTEGER",
        "name": "MEMORY",
        "doubleValue": 0.0
    }
]
```

```
    },
    {
      "name": "PORTS",
      "longValue": 0,
      "doubleValue": 0.0,
      "stringSetValue": [
        "22",
        "2376",
        "2375",
        "51678"
      ],
      "type": "STRINGSET",
      "integerValue": 0
    }
  ],
  "runningTasksCount": 0,
  "versionInfo": {
    "agentVersion": "1.0.0",
    "agentHash": "4023248",
    "dockerVersion": "DockerVersion: 1.5.0"
  }
}
]
```

Untuk informasi selengkapnya, lihat [Instans Kontainer Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DescribeContainerInstances](#) di Referensi AWS CLI Perintah.

## describe-services

Contoh kode berikut menunjukkan cara menggunakan `describe-services`.

### AWS CLI

Untuk menggambarkan layanan

`describe-services` Contoh berikut mengambil rincian untuk `my-http-service` layanan di `cluster default`.

```
aws ecs describe-services --services my-http-service
```



## Output:

```
{
  "services": [
    {
      "status": "ACTIVE",
      "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
amazon-ecs-sample:1",
      "pendingCount": 0,
      "loadBalancers": [],
      "desiredCount": 10,
      "createdAt": 1466801808.595,
      "serviceName": "my-http-service",
      "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default",
      "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/my-http-
service",
      "deployments": [
        {
          "status": "PRIMARY",
          "pendingCount": 0,
          "createdAt": 1466801808.595,
          "desiredCount": 10,
          "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/amazon-ecs-sample:1",
          "updatedAt": 1428326312.703,
          "id": "ecs-svc/1234567890123456789",
          "runningCount": 10
        }
      ],
      "events": [
        {
          "message": "(service my-http-service) has reached a steady
state.",
          "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
          "createdAt": 1466801812.435
        }
      ],
      "runningCount": 10
    }
  ],
  "failures": []
}
```

Untuk informasi selengkapnya, lihat [Layanan](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DescribeServices](#) di Referensi AWS CLI Perintah.

## describe-task-definition

Contoh kode berikut menunjukkan cara menggunakan `describe-task-definition`.

### AWS CLI

Untuk menggambarkan definisi tugas

`describe-task-definition` Contoh berikut mengambil rincian definisi tugas.

```
aws ecs describe-task-definition \  
  --task-definition hello_world:8
```

Output:

```
{  
  "tasks": [  
    {  
      "attachments": [  
        {  
          "id": "17f3dff6-a9e9-4d83-99a9-7eb5193c2634",  
          "type": "ElasticNetworkInterface",  
          "status": "ATTACHED",  
          "details": [  
            {  
              "name": "subnetId",  
              "value": "subnet-0d0eab1bb38d5ca64"  
            },  
            {  
              "name": "networkInterfaceId",  
              "value": "eni-0d542ffb4a12aa6d9"  
            },  
            {  
              "name": "macAddress",  
              "value": "0e:6d:18:f6:2d:29"  
            },  
            {  
              "name": "privateDnsName",  
              "value": "ip-10-0-1-170.ec2.internal"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```

        },
        {
            "name": "privateIPv4Address",
            "value": "10.0.1.170"
        }
    ]
}
],
"attributes": [
    {
        "name": "ecs.cpu-architecture",
        "value": "x86_64"
    }
],
"availabilityZone": "us-east-1b",
"clusterArn": "arn:aws:ecs:us-east-1:053534965804:cluster/fargate-
cluster",
"connectivity": "CONNECTED",
"connectivityAt": "2023-11-28T11:10:52.907000-05:00",
"containers": [
    {
        "containerArn": "arn:aws:ecs:us-east-1:053534965804:container/
fargate-cluster/
c524291ae4154100b601a543108b193a/772c4784-92ae-414e-8df2-03d3358e39fa",
        "taskArn": "arn:aws:ecs:us-east-1:053534965804:task/fargate-
cluster/c524291ae4154100b601a543108b193a",
        "name": "web",
        "image": "nginx",
        "imageDigest":
"sha256:10d1f5b58f74683ad34eb29287e07dab1e90f10af243f151bb50aa5dbb4d62ee",
        "runtimeId": "c524291ae4154100b601a543108b193a-265927825",
        "lastStatus": "RUNNING",
        "networkBindings": [],
        "networkInterfaces": [
            {
                "attachmentId": "17f3dff6-a9e9-4d83-99a9-7eb5193c2634",
                "privateIpv4Address": "10.0.1.170"
            }
        ],
        "healthStatus": "HEALTHY",
        "cpu": "99",
        "memory": "100"
    },
    {

```

```
        "containerArn": "arn:aws:ecs:us-east-1:053534965804:container/
fargate-cluster/c524291ae4154100b601a543108b193a/c051a779-40d2-48ca-
ad5e-6ec875ceb610",
        "taskArn": "arn:aws:ecs:us-east-1:053534965804:task/fargate-
cluster/c524291ae4154100b601a543108b193a",
        "name": "aws-guardduty-agent-FvWGoDU",
        "imageDigest":
"sha256:359b8b014e5076c625daa1056090e522631587a7afa3b2e055edda6bd1141017",
        "runtimeId": "c524291ae4154100b601a543108b193a-505093495",
        "lastStatus": "RUNNING",
        "networkBindings": [],
        "networkInterfaces": [
            {
                "attachmentId": "17f3dff6-a9e9-4d83-99a9-7eb5193c2634",
                "privateIpv4Address": "10.0.1.170"
            }
        ],
        "healthStatus": "UNKNOWN"
    }
],
"cpu": "256",
"createdAt": "2023-11-28T11:10:49.299000-05:00",
"desiredStatus": "RUNNING",
"enableExecuteCommand": false,
"group": "family:webserver",
"healthStatus": "HEALTHY",
"lastStatus": "RUNNING",
"launchType": "FARGATE",
"memory": "512"
"platformVersion": "1.4.0",
"platformFamily": "Linux",
"pullStartedAt": "2023-11-28T11:10:59.773000-05:00",
"pullStoppedAt": "2023-11-28T11:11:12.624000-05:00",
"startedAt": "2023-11-28T11:11:20.316000-05:00",
"tags": [],
"taskArn": "arn:aws:ecs:us-east-1:053534965804:task/fargate-cluster/
c524291ae4154100b601a543108b193a",
"taskDefinitionArn": "arn:aws:ecs:us-east-1:053534965804:task-
definition/webserver:5",
"version": 4,
"ephemeralStorage": {
    "sizeInGiB": 20
}
}
```

```
  ],  
  "failures": []  
}
```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DescribeTaskDefinition](#) di Referensi AWS CLI Perintah.

## describe-task-sets

Contoh kode berikut menunjukkan cara menggunakan `describe-task-sets`.

### AWS CLI

Untuk mendeskripsikan set tugas

`describe-task-sets` Contoh berikut menjelaskan tugas yang ditetapkan dalam layanan yang menggunakan deployer eksternal.

```
aws ecs describe-task-sets \  
  --cluster MyCluster \  
  --service MyService \  
  --task-sets arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-  
svc/1234567890123456789
```

Output:

```
{  
  "taskSets": [  
    {  
      "id": "ecs-svc/1234567890123456789",  
      "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
      "status": "ACTIVE",  
      "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
sample-fargate:2",  
      "computedDesiredCount": 0,  
      "pendingCount": 0,  
      "runningCount": 0,  
      "createdAt": 1557207715.195,  
      "updatedAt": 1557207740.014,  
      "launchType": "EC2",
```

```

    "networkConfiguration": {
      "awsvpcConfiguration": {
        "subnets": [
          "subnet-12344321"
        ],
        "securityGroups": [
          "sg-1234431"
        ],
        "assignPublicIp": "DISABLED"
      }
    },
    "loadBalancers": [],
    "serviceRegistries": [],
    "scale": {
      "value": 0.0,
      "unit": "PERCENT"
    },
    "stabilityStatus": "STEADY_STATE",
    "stabilityStatusAt": 1557207740.014
  }
],
"failures": []
}

```

- Untuk detail API, lihat [DescribeTaskSets](#) di Referensi AWS CLI Perintah.

## describe-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-tasks`.

### AWS CLI

Contoh 1: Untuk menggambarkan tugas tugas tunggal

`describe-tasks` Contoh berikut mengambil rincian tugas dalam sebuah cluster. Anda dapat menentukan tugas dengan menggunakan ID atau ARN penuh dari tugas tersebut. Contoh ini menggunakan ARN penuh tugas.

```

aws ecs describe-tasks \
  --cluster MyCluster \
  --tasks arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/4d590253bb114126b7afa7b58EXAMPLE

```

## Output:

```
{
  "tasks": [
    {
      "attachments": [],
      "attributes": [
        {
          "name": "ecs.cpu-architecture",
          "value": "x86_64"
        }
      ],
      "availabilityZone": "us-east-1b",
      "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
      "connectivity": "CONNECTED",
      "connectivityAt": "2021-08-11T12:21:26.681000-04:00",
      "containerInstanceArn": "arn:aws:ecs:us-east-1:123456789012:container-
instance/test/025c7e2c5e054a6790a29fc1fEXAMPLE",
      "containers": [
        {
          "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
MyCluster/4d590253bb114126b7afa7b58eea9221/a992d1cc-ea46-474a-b6e8-24688EXAMPLE",
          "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/4d590253bb114126b7afa7b58EXAMPLE",
          "name": "simple-app",
          "image": "httpd:2.4",
          "runtimeId":
"91251eed27db90006ad67b1a08187290869f216557717dd5c39b37c94EXAMPLE",
          "lastStatus": "RUNNING",
          "networkBindings": [
            {
              "bindIP": "0.0.0.0",
              "containerPort": 80,
              "hostPort": 80,
              "protocol": "tcp"
            }
          ],
          "networkInterfaces": [],
          "healthStatus": "UNKNOWN",
          "cpu": "10",
          "memory": "300"
        }
      ],
      "cpu": "10",
```

```

    "createdAt": "2021-08-11T12:21:26.681000-04:00",
    "desiredStatus": "RUNNING",
    "enableExecuteCommand": false,
    "group": "service:testupdate",
    "healthStatus": "UNKNOWN",
    "lastStatus": "RUNNING",
    "launchType": "EC2",
    "memory": "300",
    "overrides": {
      "containerOverrides": [
        {
          "name": "simple-app"
        }
      ],
      "inferenceAcceleratorOverrides": []
    },
    "pullStartedAt": "2021-08-11T12:21:28.234000-04:00",
    "pullStoppedAt": "2021-08-11T12:21:33.793000-04:00",
    "startedAt": "2021-08-11T12:21:34.945000-04:00",
    "startedBy": "ecs-svc/968695068243EXAMPLE",
    "tags": [],
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/4d590253bb114126b7afa7b58eea9221",
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/console-sample-app-static2:1",
    "version": 2
  }
],
"failures": []
}

```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk menggambarkan beberapa tugas

`describe-tasks` Contoh berikut mengambil rincian beberapa tugas dalam sebuah cluster. Anda dapat menentukan tugas dengan menggunakan ID atau ARN penuh dari tugas tersebut. Contoh ini menggunakan ID lengkap tugas.

```

aws ecs describe-tasks \
  --cluster MyCluster \
  --tasks "74de0355a10a4f979ac495c14EXAMPLE" "d789e94343414c25b9f6bd59eEXAMPLE"

```



## Output:

```
{
  "tasks": [
    {
      "attachments": [
        {
          "id": "d9e7735a-16aa-4128-bc7a-b2d51EXAMPLE",
          "type": "ElasticNetworkInterface",
          "status": "ATTACHED",
          "details": [
            {
              "name": "subnetId",
              "value": "subnet-0d0eab1bb3EXAMPLE"
            },
            {
              "name": "networkInterfaceId",
              "value": "eni-0fa40520aeEXAMPLE"
            },
            {
              "name": "macAddress",
              "value": "0e:89:76:28:07:b3"
            },
            {
              "name": "privateDnsName",
              "value": "ip-10-0-1-184.ec2.internal"
            },
            {
              "name": "privateIPv4Address",
              "value": "10.0.1.184"
            }
          ]
        }
      ],
      "attributes": [
        {
          "name": "ecs.cpu-architecture",
          "value": "x86_64"
        }
      ],
      "availabilityZone": "us-east-1b",
      "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
      "connectivity": "CONNECTED",
      "connectivityAt": "2021-12-20T12:13:37.875000-05:00",
    }
  ]
}
```

```
"containers": [
  {
    "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE/aad3ba00-83b3-4dac-84d4-11f8cEXAMPLE",
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE",
    "name": "web",
    "image": "nginx",
    "runtimeId": "74de0355a10a4f979ac495c14EXAMPLE-265927825",
    "lastStatus": "RUNNING",
    "networkBindings": [],
    "networkInterfaces": [
      {
        "attachmentId": "d9e7735a-16aa-4128-bc7a-b2d51EXAMPLE",
        "privateIpv4Address": "10.0.1.184"
      }
    ],
    "healthStatus": "UNKNOWN",
    "cpu": "99",
    "memory": "100"
  }
],
"cpu": "256",
"createdAt": "2021-12-20T12:13:20.226000-05:00",
"desiredStatus": "RUNNING",
"enableExecuteCommand": false,
"group": "service:tdsevicetag",
"healthStatus": "UNKNOWN",
"lastStatus": "RUNNING",
"launchType": "FARGATE",
"memory": "512",
"overrides": {
  "containerOverrides": [
    {
      "name": "web"
    }
  ],
  "inferenceAcceleratorOverrides": []
},
"platformVersion": "1.4.0",
"platformFamily": "Linux",
"pullStartedAt": "2021-12-20T12:13:42.665000-05:00",
"pullStoppedAt": "2021-12-20T12:13:46.543000-05:00",
"startedAt": "2021-12-20T12:13:48.086000-05:00",
```

```
    "startedBy": "ecs-svc/988401040018EXAMPLE",
    "tags": [],
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE",
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/webserver:2",
    "version": 3,
    "ephemeralStorage": {
      "sizeInGiB": 20
    }
  },
  {
    "attachments": [
      {
        "id": "214eb5a9-45cd-4bf8-87bc-57fefEXAMPLE",
        "type": "ElasticNetworkInterface",
        "status": "ATTACHED",
        "details": [
          {
            "name": "subnetId",
            "value": "subnet-0d0eab1bb3EXAMPLE"
          },
          {
            "name": "networkInterfaceId",
            "value": "eni-064c7766daEXAMPLE"
          },
          {
            "name": "macAddress",
            "value": "0e:76:83:01:17:a9"
          },
          {
            "name": "privateDnsName",
            "value": "ip-10-0-1-41.ec2.internal"
          },
          {
            "name": "privateIPv4Address",
            "value": "10.0.1.41"
          }
        ]
      }
    ],
    "attributes": [
      {
        "name": "ecs.cpu-architecture",
```

```
        "value": "x86_64"
      }
    ],
    "availabilityZone": "us-east-1b",
    "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
    "connectivity": "CONNECTED",
    "connectivityAt": "2021-12-20T12:13:35.243000-05:00",
    "containers": [
      {
        "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE/9afef792-609b-43a5-bb6a-3efdbEXAMPLE",
        "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE",
        "name": "web",
        "image": "nginx",
        "runtimeId": "d789e94343414c25b9f6bd59eEXAMPLE-265927825",
        "lastStatus": "RUNNING",
        "networkBindings": [],
        "networkInterfaces": [
          {
            "attachmentId": "214eb5a9-45cd-4bf8-87bc-57fefEXAMPLE",
            "privateIpv4Address": "10.0.1.41"
          }
        ],
        "healthStatus": "UNKNOWN",
        "cpu": "99",
        "memory": "100"
      }
    ],
    "cpu": "256",
    "createdAt": "2021-12-20T12:13:20.226000-05:00",
    "desiredStatus": "RUNNING",
    "enableExecuteCommand": false,
    "group": "service:tdsevicetag",
    "healthStatus": "UNKNOWN",
    "lastStatus": "RUNNING",
    "launchType": "FARGATE",
    "memory": "512",
    "overrides": {
      "containerOverrides": [
        {
          "name": "web"
        }
      ]
    }
  ],
```

```

        "inferenceAcceleratorOverrides": [],
    },
    "platformVersion": "1.4.0",
    "platformFamily": "Linux",
    "pullStartedAt": "2021-12-20T12:13:44.611000-05:00",
    "pullStoppedAt": "2021-12-20T12:13:48.251000-05:00",
    "startedAt": "2021-12-20T12:13:49.326000-05:00",
    "startedBy": "ecs-svc/988401040018EXAMPLE",
    "tags": [],
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/
d789e94343414c25b9f6bd59eEXAMPLE",
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/webserver:2",
    "version": 3,
    "ephemeralStorage": {
        "sizeInGiB": 20
    }
}
],
"failures": []
}

```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [DescribeTasks](#) di Referensi AWS CLI Perintah.

## execute-command

Contoh kode berikut menunjukkan cara menggunakan `execute-command`.

### AWS CLI

Untuk menjalankan perintah `/bin/sh` interaktif

`execute-command` Contoh berikut menjalankan perintah `/bin/sh` interaktif terhadap wadah bernama `MyContainer` untuk tugas dengan id dari `arn:aws:ecs:us-east-1:123456789012:task/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE`

```

aws ecs execute-command \
  --cluster MyCluster \
  --task arn:aws:ecs:us-east-1:123456789012:task/MyCluster/
d789e94343414c25b9f6bd59eEXAMPLE \

```

```
--container MyContainer \  
--interactive \  
--command "/bin/sh"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan Amazon ECS Exec untuk men-debug di Panduan Pengembang](#) Amazon ECS.

- Untuk detail API, lihat [ExecuteCommand](#) di Referensi AWS CLI Perintah.

## list-account-settings

Contoh kode berikut menunjukkan cara menggunakan `list-account-settings`.

### AWS CLI

Contoh 1: Untuk melihat pengaturan akun untuk akun

`list-account-settings` Contoh berikut menampilkan pengaturan akun yang efektif untuk akun.

```
aws ecs list-account-settings --effective-settings
```

Output:

```
{  
  "settings": [  
    {  
      "name": "containerInstanceLongArnFormat",  
      "value": "enabled",  
      "principalArn": "arn:aws:iam::123456789012:root"  
    },  
    {  
      "name": "serviceLongArnFormat",  
      "value": "enabled",  
      "principalArn": "arn:aws:iam::123456789012:root"  
    },  
    {  
      "name": "taskLongArnFormat",  
      "value": "enabled",  
      "principalArn": "arn:aws:iam::123456789012:root"  
    }  
  ]  
}
```

```

    }
  ]
}

```

Contoh 2: Untuk melihat pengaturan akun untuk pengguna IAM tertentu atau peran IAM

`list-account-settings` Contoh berikut menampilkan pengaturan akun untuk pengguna IAM tertentu atau peran IAM.

```
aws ecs list-account-settings --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Output:

```

{
  "settings": [
    {
      "name": "serviceLongArnFormat",
      "value": "enabled",
      "principalArn": "arn:aws:iam::123456789012:user/MyUser"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Nama Sumber Daya Amazon \(ARN\) dan ID](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListAccountSettings](#) di Referensi AWS CLI Perintah.

## list-attributes

Contoh kode berikut menunjukkan cara menggunakan `list-attributes`.

### AWS CLI

Untuk membuat daftar instance kontainer yang berisi atribut tertentu

Contoh berikut mencantumkan atribut untuk instance kontainer yang memiliki `stack=production` atribut di cluster default.

```
aws ecs list-attributes \
```

```
--target-type container-instance \  
--attribute-name stack \  
--attribute-value production \  
--cluster default
```

Output:

```
{  
  "attributes": [  
    {  
      "name": "stack",  
      "targetId": "arn:aws:ecs:us-west-2:130757420319:container-  
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",  
      "value": "production"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Konfigurasi Agen Kontainer Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListAttributes](#) di Referensi AWS CLI Perintah.

## list-clusters

Contoh kode berikut menunjukkan cara menggunakan `list-clusters`.

AWS CLI

Untuk membuat daftar cluster yang tersedia

`list-clusters` Contoh berikut mencantumkan semua cluster yang tersedia.

```
aws ecs list-clusters
```

Output:

```
{  
  "clusterArns": [  
    "arn:aws:ecs:us-west-2:123456789012:cluster/MyECSCluster1",  
    "arn:aws:ecs:us-west-2:123456789012:cluster/AnotherECSCluster"  
  ]  
}
```



```
]
}
```

Untuk informasi selengkapnya, lihat [Cluster Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListClusters](#) di Referensi AWS CLI Perintah.

## **list-container-instances**

Contoh kode berikut menunjukkan cara menggunakan `list-container-instances`.

### AWS CLI

Untuk membuat daftar instance kontainer dalam sebuah cluster

`list-container-instances` Contoh berikut mencantumkan semua instance kontainer yang tersedia dalam sebuah cluster.

```
aws ecs list-container-instances --cluster MyCluster
```

Output:

```
{
  "containerInstanceArns": [
    "arn:aws:ecs:us-west-2:123456789012:container-instance/MyCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "arn:aws:ecs:us-west-2:123456789012:container-instance/MyCluster/
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
  ]
}
```

Untuk informasi selengkapnya, lihat [Instans Kontainer Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListContainerInstances](#) di Referensi AWS CLI Perintah.

## **list-services-by-namespace**

Contoh kode berikut menunjukkan cara menggunakan `list-services-by-namespace`.

## AWS CLI

Untuk membuat daftar layanan di namespace

`list-services-by-namespace` Contoh berikut mencantumkan semua layanan yang dikonfigurasi untuk namespace yang ditentukan di Region default Anda.

```
aws ecs list-services-by-namespace \  
  --namespace service-connect
```

Output:

```
{  
  "serviceArns": [  
    "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService",  
    "arn:aws:ecs:us-west-2:123456789012:service/tutorial/service-connect-nginx-  
service"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Service Connect](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListServicesByNamespace](#) di Referensi AWS CLI Perintah.

## `list-services`

Contoh kode berikut menunjukkan cara menggunakan `list-services`.

## AWS CLI

Untuk membuat daftar layanan dalam klaster

`list-services` Contoh berikut menunjukkan cara membuat daftar layanan yang berjalan di cluster.

```
aws ecs list-services --cluster MyCluster
```

Output:

```
{
```

```
    "serviceArns": [  
        "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService"  
    ]  
}
```

Untuk informasi selengkapnya, lihat [Layanan](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListServices](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk cluster tertentu.

```
aws ecs list-tags-for-resource \  
    --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster
```

Output:

```
{  
  "tags": [  
    {  
      "key": "key1",  
      "value": "value1"  
    },  
    {  
      "key": "key2",  
      "value": "value2"  
    },  
    {  
      "key": "key3",  
      "value": "value3"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## list-task-definition-families

Contoh kode berikut menunjukkan cara menggunakan `list-task-definition-families`.

### AWS CLI

Contoh 1: Untuk membuat daftar keluarga definisi tugas terdaftar

`list-task-definition-families` Contoh berikut mencantumkan semua keluarga definisi tugas terdaftar.

```
aws ecs list-task-definition-families
```

Output:

```
{
  "families": [
    "node-js-app",
    "web-timer",
    "hpcc",
    "hpcc-c4-8xlarge"
  ]
}
```

Contoh 2: Untuk memfilter keluarga definisi tugas yang terdaftar

`list-task-definition-families` Contoh berikut mencantumkan revisi definisi tugas yang dimulai dengan "hpcc".

```
aws ecs list-task-definition-families --family-prefix hpcc
```

Output:

```
{
  "families": [
    "hpcc",
    "hpcc-c4-8xlarge"
  ]
}
```

Untuk informasi selengkapnya, lihat [Parameter Definisi Tugas](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListTaskDefinitionFamilies](#) di Referensi AWS CLI Perintah.

## list-task-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-task-definitions`.

### AWS CLI

Contoh 1: Untuk mencantumkan definisi tugas yang terdaftar

`list-task-definitions` Contoh berikut mencantumkan semua definisi tugas terdaftar.

```
aws ecs list-task-definitions
```

Output:

```
{
  "taskDefinitionArns": [
    "arn:aws:ecs:us-west-2:123456789012:task-definition/sleep300:2",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/sleep360:1",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:3",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:4",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:5",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:6"
  ]
}
```

Contoh 2: Untuk mencantumkan definisi tugas yang terdaftar dalam keluarga

`list-task-definitions` Contoh berikut mencantumkan revisi definisi tugas dari keluarga tertentu.

```
aws ecs list-task-definitions --family-prefix wordpress
```

Output:

```
{
  "taskDefinitionArns": [
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:3",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:4",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:5",
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:6"
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListTaskDefinitions](#) di Referensi AWS CLI Perintah.

## list-tasks

Contoh kode berikut menunjukkan cara menggunakan `list-tasks`.

### AWS CLI

Contoh 1: Untuk membuat daftar tugas dalam sebuah cluster

`list-tasks` Contoh berikut mencantumkan semua tugas dalam sebuah cluster.

```
aws ecs list-tasks --cluster default
```

Output:

```
{
  "taskArns": [
    "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-11111EXAMPLE",
    "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-22222EXAMPLE"
  ]
}
```

Contoh 2: Untuk membuat daftar tugas pada instance kontainer tertentu

`list-tasks` Contoh berikut mencantumkan tugas-tugas pada instance container, menggunakan container instance UUID sebagai filter.

```
aws ecs list-tasks --cluster default --container-instance a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE
```

Output:

```
{
  "taskArns": [
    "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-44444EXAMPLE"
  ]
}
```

Untuk informasi selengkapnya, lihat [Definisi Tugas Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [ListTasks](#) di Referensi AWS CLI Perintah.

## put-account-setting-default

Contoh kode berikut menunjukkan cara menggunakan `put-account-setting-default`.

### AWS CLI

Untuk mengubah pengaturan akun default

`put-account-setting-default` Contoh berikut memodifikasi pengaturan akun default untuk semua pengguna IAM atau peran di akun Anda. Perubahan ini berlaku untuk seluruh AWS akun kecuali pengguna atau peran IAM secara eksplisit mengesampingkan pengaturan ini untuk diri mereka sendiri.

```
aws ecs put-account-setting-default --name serviceLongArnFormat --value enabled
```

Output:

```
{
  "setting": {
    "name": "serviceLongArnFormat",
    "value": "enabled",
    "principalArn": "arn:aws:iam::123456789012:root"
  }
}
```

Untuk informasi selengkapnya, lihat [Nama Sumber Daya Amazon \(ARN\) dan ID](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [PutAccountSettingDefault](#) di Referensi AWS CLI Perintah.

## put-account-setting

Contoh kode berikut menunjukkan cara menggunakan `put-account-setting`.

### AWS CLI

Untuk mengubah pengaturan akun untuk akun pengguna IAM Anda

`put-account-setting` Contoh berikut memungkinkan pengaturan `serviceLongArnFormat` akun untuk akun pengguna IAM Anda.

```
aws ecs put-account-setting --name serviceLongArnFormat --value enabled
```

Output:

```
{
  "setting": {
    "name": "serviceLongArnFormat",
    "value": "enabled",
    "principalArn": "arn:aws:iam::130757420319:user/your_username"
  }
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Pengaturan Akun](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [PutAccountSetting](#) di Referensi AWS CLI Perintah.

## put-account-settings

Contoh kode berikut menunjukkan cara menggunakan `put-account-settings`.

### AWS CLI

Untuk mengubah pengaturan akun untuk pengguna IAM atau peran IAM

`put-account-setting` Contoh berikut memodifikasi pengaturan akun untuk pengguna IAM tertentu atau peran IAM.

```
aws ecs put-account-setting \
  --name serviceLongArnFormat \
  --value enabled \
```



```
--principal-arn arn:aws:iam::123456789012:user/MyUser
```

Output:

```
{
  "setting": {
    "name": "serviceLongArnFormat",
    "value": "enabled",
    "principalArn": "arn:aws:iam::123456789012:user/MyUser"
  }
}
```

- Untuk detail API, lihat [PutAccountSettings](#) di Referensi AWS CLI Perintah.

## put-attributes

Contoh kode berikut menunjukkan cara menggunakan `put-attributes`.

AWS CLI

Untuk membuat atribut dan mengaitkannya dengan sumber daya Amazon ECS

Berikut ini `put-attributes` menerapkan atribut dengan nama `stack` dan nilai produksi untuk instance container.

```
aws ecs put-attributes \
  --attributes name=stack,value=production,targetId=arn:aws:ecs:us-
west-2:130757420319:container-instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34
```

Output:

```
{
  "attributes": [
    {
      "name": "stack",
      "targetId": "arn:aws:ecs:us-west-2:130757420319:container-
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",
      "value": "production"
    }
  ]
}
```

- Untuk detail API, lihat [PutAttributes](#) di Referensi AWS CLI Perintah.

## put-cluster-capacity-providers

Contoh kode berikut menunjukkan cara menggunakan `put-cluster-capacity-providers`.

### AWS CLI

Contoh 1: Untuk menambahkan penyedia kapasitas yang ada ke cluster

`put-cluster-capacity-providers` Contoh berikut menambahkan penyedia kapasitas yang ada ke cluster. `create-capacity-provider` Perintah ini digunakan untuk membuat penyedia kapasitas. `describe-clusters` Perintah ini digunakan untuk menggambarkan penyedia kapasitas saat ini dan strategi penyedia kapasitas default yang terkait dengan cluster. Saat menambahkan penyedia kapasitas baru ke klaster, Anda harus menentukan semua penyedia kapasitas yang ada selain penyedia kapasitas baru yang ingin Anda kaitkan dengan cluster. Anda juga harus menentukan strategi penyedia kapasitas default untuk dikaitkan dengan cluster. Dalam contoh ini, `MyCluster` cluster memiliki penyedia `MyCapacityProvider1` kapasitas yang terkait dengannya dan Anda ingin menambahkan penyedia `MyCapacityProvider2` kapasitas dan memasukkannya ke dalam strategi penyedia kapasitas default sehingga tugas tersebar merata di kedua penyedia kapasitas.

```
aws ecs put-cluster-capacity-providers \
  --cluster MyCluster \
  --capacity-providers MyCapacityProvider1 MyCapacityProvider2 \
  --default-capacity-provider-strategy
capacityProvider=MyCapacityProvider1,weight=1
capacityProvider=MyCapacityProvider2,weight=1
```

Output:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "runningTasksCount": 0,
    "pendingTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
```

```
"tags": [],
"settings": [
  {
    "name": "containerInsights",
    "value": "enabled"
  }
],
"capacityProviders": [
  "MyCapacityProvider1",
  "MyCapacityProvider2"
],
"defaultCapacityProviderStrategy": [
  {
    "capacityProvider": "MyCapacityProvider1",
    "weight": 1,
    "base": 0
  },
  {
    "capacityProvider": "MyCapacityProvider2",
    "weight": 1,
    "base": 0
  }
],
"attachments": [
  {
    "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
    "type": "as_policy",
    "status": "ACTIVE",
    "details": [
      {
        "name": "capacityProviderName",
        "value": "MyCapacityProvider1"
      },
      {
        "name": "scalingPolicyName",
        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
      }
    ]
  },
  {
    "id": "ae592060-2382-4663-9476-b015c685593c",
    "type": "as_policy",
    "status": "ACTIVE",
```

```

        "details": [
            {
                "name": "capacityProviderName",
                "value": "MyCapacityProvider2"
            },
            {
                "name": "scalingPolicyName",
                "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
            }
        ]
    },
    "attachmentsStatus": "UPDATE_IN_PROGRESS"
}

```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas kluster](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk menghapus penyedia kapasitas dari cluster

`put-cluster-capacity-providers` Contoh berikut menghapus penyedia kapasitas dari cluster. `describe-clusters` Perintah ini digunakan untuk menggambarkan penyedia kapasitas saat ini yang terkait dengan cluster. Saat menghapus penyedia kapasitas dari kluster, Anda harus menentukan penyedia kapasitas yang ingin tetap terkait dengan kluster serta strategi penyedia kapasitas default untuk dikaitkan dengan cluster. Dalam contoh ini, cluster memiliki `MyCapacityProvider1` dan penyedia `MyCapacityProvider2` kapasitas yang terkait dengannya dan Anda ingin menghapus penyedia `MyCapacityProvider2` kapasitas, sehingga Anda hanya menentukan `MyCapacityProvider1` dalam perintah bersama dengan strategi penyedia kapasitas default yang diperbarui.

```

aws ecs put-cluster-capacity-providers \
  --cluster MyCluster \
  --capacity-providers MyCapacityProvider1 \
  --default-capacity-provider-strategy
capacityProvider=MyCapacityProvider1,weight=1,base=0

```

Output:

```
{
```

```
"cluster": {
  "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
  "clusterName": "MyCluster",
  "status": "ACTIVE",
  "registeredContainerInstancesCount": 0,
  "runningTasksCount": 0,
  "pendingTasksCount": 0,
  "activeServicesCount": 0,
  "statistics": [],
  "tags": [],
  "settings": [
    {
      "name": "containerInsights",
      "value": "enabled"
    }
  ],
  "capacityProviders": [
    "MyCapacityProvider1"
  ],
  "defaultCapacityProviderStrategy": [
    {
      "capacityProvider": "MyCapacityProvider1",
      "weight": 1,
      "base": 0
    }
  ],
  "attachments": [
    {
      "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
      "type": "as_policy",
      "status": "ACTIVE",
      "details": [
        {
          "name": "capacityProviderName",
          "value": "MyCapacityProvider1"
        },
        {
          "name": "scalingPolicyName",
          "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
        }
      ]
    },
    {
      "id": "ae592060-2382-4663-9476-b015c685593c",
      "type": "as_policy",
```

```

        "status": "DELETING",
        "details": [
            {
                "name": "capacityProviderName",
                "value": "MyCapacityProvider2"
            },
            {
                "name": "scalingPolicyName",
                "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
            }
        ]
    },
    "attachmentsStatus": "UPDATE_IN_PROGRESS"
}

```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas klaster](#) di Panduan Pengembang Amazon ECS.

Contoh 3: Untuk menghapus semua penyedia kapasitas dari cluster

`put-cluster-capacity-providers` Contoh berikut menghapus semua penyedia kapasitas yang ada dari cluster.

```

aws ecs put-cluster-capacity-providers \
  --cluster MyCluster \
  --capacity-providers [] \
  --default-capacity-provider-strategy []

```

Output:

```

{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "runningTasksCount": 0,
    "pendingTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],

```

```
"tags": [],
"settings": [
  {
    "name": "containerInsights",
    "value": "enabled"
  }
],
"capacityProviders": [],
"defaultCapacityProviderStrategy": [],
"attachments": [
  {
    "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
    "type": "as_policy",
    "status": "DELETING",
    "details": [
      {
        "name": "capacityProviderName",
        "value": "MyCapacityProvider1"
      },
      {
        "name": "scalingPolicyName",
        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
      }
    ]
  },
  {
    "id": "ae592060-2382-4663-9476-b015c685593c",
    "type": "as_policy",
    "status": "DELETING",
    "details": [
      {
        "name": "capacityProviderName",
        "value": "MyCapacityProvider2"
      },
      {
        "name": "scalingPolicyName",
        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
      }
    ]
  }
],
"attachmentsStatus": "UPDATE_IN_PROGRESS"
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Penyedia kapasitas klaster](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [PutClusterCapacityProviders](#) di Referensi AWS CLI Perintah.

## register-task-definition

Contoh kode berikut menunjukkan cara menggunakan `register-task-definition`.

### AWS CLI

Contoh 1: Untuk mendaftarkan definisi tugas dengan file JSON

`register-task-definition` Contoh berikut mendaftarkan definisi tugas ke keluarga yang ditentukan. Definisi wadah disimpan dalam format JSON di lokasi file yang ditentukan.

```
aws ecs register-task-definition \  
  --cli-input-json file://<path_to_json_file>/sleep360.json
```

Isi dari `sleep360.json`:

```
{  
  "containerDefinitions": [  
    {  
      "name": "sleep",  
      "image": "busybox",  
      "cpu": 10,  
      "command": [  
        "sleep",  
        "360"  
      ],  
      "memory": 10,  
      "essential": true  
    }  
  ],  
  "family": "sleep360"  
}
```

Output:



```
{
  "taskDefinition": {
    "status": "ACTIVE",
    "family": "sleep360",
    "placementConstraints": [],
    "compatibilities": [
      "EXTERNAL",
      "EC2"
    ],
    "volumes": [],
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/sleep360:1",
    "containerDefinitions": [
      {
        "environment": [],
        "name": "sleep",
        "mountPoints": [],
        "image": "busybox",
        "cpu": 10,
        "portMappings": [],
        "command": [
          "sleep",
          "360"
        ],
        "memory": 10,
        "essential": true,
        "volumesFrom": []
      }
    ],
    "revision": 1
  }
}
```

Untuk informasi selengkapnya, lihat [Contoh definisi tugas](#) di Panduan Pengembang Amazon ECS.

Contoh 2: Untuk mendaftarkan definisi tugas dengan parameter string JSON

`register-task-definition` Contoh berikut mendaftarkan definisi tugas menggunakan definisi kontainer yang disediakan sebagai parameter string JSON dengan tanda kutip ganda yang lolos.

```
aws ecs register-task-definition \
  --family sleep360 \
```

```
--container-definitions "[{"name":"sleep","image":"busybox","cpu":10,
  "command":["sleep","360"],"memory":10,"essential":true}]"
```

Outputnya identik dengan contoh sebelumnya.

Untuk informasi selengkapnya, lihat [Membuat Definisi Tugas](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [RegisterTaskDefinition](#) di Referensi AWS CLI Perintah.

## run-task

Contoh kode berikut menunjukkan cara menggunakan `run-task`.

### AWS CLI

Untuk menjalankan tugas di klaster default Anda

`run-task` Contoh berikut menjalankan tugas pada cluster default dan menggunakan token klien.

```
aws ecs run-task \
  --cluster default \
  --task-definition sleep360:1 \
  --client-token 550e8400-e29b-41d4-a716-446655440000
```

Output:

```
{
  "tasks": [
    {
      "attachments": [],
      "attributes": [
        {
          "name": "ecs.cpu-architecture",
          "value": "x86_64"
        }
      ],
      "availabilityZone": "us-east-1b",
      "capacityProviderName": "example-capacity-provider",
      "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/default",
      "containerInstanceArn": "arn:aws:ecs:us-east-1:123456789012:container-instance/default/bc4d2ec611d04bb7bb97e83ceEXAMPLE",
```

```
    "containers": [
      {
        "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
default/d6f51cc5bbc94a47969c92035e9f66f8/75853d2d-711e-458a-8362-0f0aEXAMPLE",
        "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/default/
d6f51cc5bbc94a47969c9203EXAMPLE",
        "name": "sleep",
        "image": "busybox",
        "lastStatus": "PENDING",
        "networkInterfaces": [],
        "cpu": "10",
        "memory": "10"
      }
    ],
    "cpu": "10",
    "createdAt": "2023-11-21T16:59:34.403000-05:00",
    "desiredStatus": "RUNNING",
    "enableExecuteCommand": false,
    "group": "family:sleep360",
    "lastStatus": "PENDING",
    "launchType": "EC2",
    "memory": "10",
    "overrides": {
      "containerOverrides": [
        {
          "name": "sleep"
        }
      ],
      "inferenceAcceleratorOverrides": []
    },
    "tags": [],
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/default/
d6f51cc5bbc94a47969c9203EXAMPLE",
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/sleep360:1",
    "version": 1
  }
],
"failures": []
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Tugas](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [RunTask](#) di Referensi AWS CLI Perintah.

## start-task

Contoh kode berikut menunjukkan cara menggunakan `start-task`.

### AWS CLI

Untuk memulai tugas baru

Berikut ini `start-task` memulai tugas menggunakan revisi terbaru dari definisi `sleep360` tugas pada instance kontainer yang ditentukan di cluster default.

```
aws ecs start-task \  
  --task-definition sleep360 \  
  --container-instances 765936fadbdd46b5991a4bd70c2a43d4
```

Output:

```
{  
  "tasks": [  
    {  
      "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/  
default/666fdccc2e2d4b6894dd422f4eeee8f8",  
      "clusterArn": "arn:aws:ecs:us-west-2:130757420319:cluster/default",  
      "taskDefinitionArn": "arn:aws:ecs:us-west-2:130757420319:task-  
definition/sleep360:3",  
      "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-  
instance/default/765936fadbdd46b5991a4bd70c2a43d4",  
      "overrides": {  
        "containerOverrides": [  
          {  
            "name": "sleep"  
          }  
        ]  
      },  
      "lastStatus": "PENDING",  
      "desiredStatus": "RUNNING",  
      "cpu": "128",  
      "memory": "128",  
      "containers": [  
        {  
          "containerArn": "arn:aws:ecs:us-  
west-2:130757420319:container/75f11ed4-8a3d-4f26-a33b-ad1db9e02d41",
```

```
        "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",
        "name": "sleep",
        "lastStatus": "PENDING",
        "networkInterfaces": [],
        "cpu": "10",
        "memory": "10"
    }
],
"version": 1,
"createdAt": 1563421494.186,
"group": "family:sleep360",
"launchType": "EC2",
"attachments": [],
"tags": []
}
],
"failures": []
}
```

- Untuk detail API, lihat [StartTask](#) di Referensi AWS CLI Perintah.

## stop-task

Contoh kode berikut menunjukkan cara menggunakan `stop-task`.

### AWS CLI

Untuk menghentikan tugas

Berikut ini `stop-task` menghentikan tugas yang ditentukan agar tidak berjalan di cluster default.

```
aws ecs stop-task \
  --task 666fdccc2e2d4b6894dd422f4eeee8f8
```

Output:

```
{
  "task": {
    "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",
    "clusterArn": "arn:aws:ecs:us-west-2:130757420319:cluster/default",
```

```

    "taskDefinitionArn": "arn:aws:ecs:us-west-2:130757420319:task-definition/
sleep360:3",
    "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
    "overrides": {
      "containerOverrides": []
    },
    "lastStatus": "STOPPED",
    "desiredStatus": "STOPPED",
    "cpu": "128",
    "memory": "128",
    "containers": [],
    "version": 2,
    "stoppedReason": "Taskfailedtostart",
    "stopCode": "TaskFailedToStart",
    "connectivity": "CONNECTED",
    "connectivityAt": 1563421494.186,
    "pullStartedAt": 1563421494.252,
    "pullStoppedAt": 1563421496.252,
    "executionStoppedAt": 1563421497,
    "createdAt": 1563421494.186,
    "stoppingAt": 1563421497.252,
    "stoppedAt": 1563421497.252,
    "group": "family:sleep360",
    "launchType": "EC2",
    "attachments": [],
    "tags": []
  }
}

```

- Untuk detail API, lihat [StopTask](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan tag tunggal ke sumber daya yang ditentukan.

```
aws ecs tag-resource \
```

```
--resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster
--tags key=key1,value=value1
```

Perintah ini tidak menghasilkan output.

Untuk menambahkan beberapa tag ke sumber daya

`tag-resource` Contoh berikut menambahkan beberapa tag ke sumber daya yang ditentukan.

```
aws ecs tag-resource \
--resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
--tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag yang terdaftar dari sumber daya yang ditentukan.

```
aws ecs untag-resource \
--resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
--tag-keys key1,key2
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-cluster-settings**

Contoh kode berikut menunjukkan cara menggunakan `update-cluster-settings`.

### AWS CLI

Untuk mengubah pengaturan untuk klaster Anda

`update-cluster-settings` Contoh berikut memungkinkan CloudWatch Container Insights untuk default cluster.

```
aws ecs update-cluster-settings \  
  --cluster default \  
  --settings name=containerInsights,value=enabled
```

Output:

```
{  
  "cluster": {  
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",  
    "clusterName": "default",  
    "status": "ACTIVE",  
    "registeredContainerInstancesCount": 0,  
    "runningTasksCount": 0,  
    "pendingTasksCount": 0,  
    "activeServicesCount": 0,  
    "statistics": [],  
    "tags": [],  
    "settings": [  
      {  
        "name": "containerInsights",  
        "value": "enabled"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Pengaturan Akun](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [UpdateClusterSettings](#) di Referensi AWS CLI Perintah.

## update-container-agent

Contoh kode berikut menunjukkan cara menggunakan `update-container-agent`.

### AWS CLI

Untuk memperbarui agen kontainer pada instans penampung Amazon ECS



`update-container-agent` Contoh berikut memperbarui agen kontainer pada instance kontainer tertentu di cluster default.

```
aws ecs update-container-agent --cluster default --container-instance
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
  "containerInstance": {
    "status": "ACTIVE",
    ...
    "agentUpdateStatus": "PENDING",
    "versionInfo": {
      "agentVersion": "1.0.0",
      "agentHash": "4023248",
      "dockerVersion": "DockerVersion: 1.5.0"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Agen Kontainer Amazon ECS](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [UpdateContainerAgent](#) di Referensi AWS CLI Perintah.

## **update-container-instances-state**

Contoh kode berikut menunjukkan cara menggunakan `update-container-instances-state`.

AWS CLI

Untuk memperbarui status instance kontainer

Berikut ini `update-container-instances-state` memperbarui status instance kontainer tertentu DRAINING yang akan menghapusnya dari cluster apakah itu terdaftar.

```
aws ecs update-container-instances-state \
  --container-instances 765936fadbdd46b5991a4bd70c2a43d4 \
  --status DRAINING
```

## Output:

```
{
  "containerInstances": [
    {
      "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
      "ec2InstanceId": "i-013d87ffbb4d513bf",
      "version": 4390,
      "versionInfo": {
        "agentVersion": "1.29.0",
        "agentHash": "a190a73f",
        "dockerVersion": "DockerVersion:18.06.1-ce"
      },
      "remainingResources": [
        {
          "name": "CPU",
          "type": "INTEGER",
          "doubleValue": 0,
          "longValue": 0,
          "integerValue": 1536
        },
        {
          "name": "MEMORY",
          "type": "INTEGER",
          "doubleValue": 0,
          "longValue": 0,
          "integerValue": 2681
        },
        {
          "name": "PORTS",
          "type": "STRINGSET",
          "doubleValue": 0,
          "longValue": 0,
          "integerValue": 0,
          "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678",
            "51679"
          ]
        }
      ]
    }
  ]
}
```

```
        "name": "PORTS_UDP",
        "type": "STRINGSET",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 0,
        "stringSetValue": []
    }
],
"registeredResources": [
    {
        "name": "CPU",
        "type": "INTEGER",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 2048
    },
    {
        "name": "MEMORY",
        "type": "INTEGER",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 3705
    },
    {
        "name": "PORTS",
        "type": "STRINGSET",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 0,
        "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678",
            "51679"
        ]
    },
    {
        "name": "PORTS_UDP",
        "type": "STRINGSET",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 0,
        "stringSetValue": []
    }
]
```

```
    }
  ],
  "status": "DRAINING",
  "agentConnected": true,
  "runningTasksCount": 2,
  "pendingTasksCount": 0,
  "attributes": [
    {
      "name": "ecs.capability.secrets.asm.environment-variables"
    },
    {
      "name": "ecs.capability.branch-cni-plugin-version",
      "value": "e0703516-"
    },
    {
      "name": "ecs.ami-id",
      "value": "ami-00e0090ac21971297"
    },
    {
      "name": "ecs.capability.secrets.asm.bootstrap.log-driver"
    },
    {
      "name": "com.amazonaws.ecs.capability.logging-driver.none"
    },
    {
      "name": "ecs.capability.ecr-endpoint"
    },
    {
      "name": "ecs.capability.docker-plugin.local"
    },
    {
      "name": "ecs.capability.task-cpu-mem-limit"
    },
    {
      "name": "ecs.capability.secrets.ssm.bootstrap.log-driver"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.30"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.31"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.32"
```

```
    },
    {
      "name": "ecs.availability-zone",
      "value": "us-west-2c"
    },
    {
      "name": "ecs.capability.aws-appmesh"
    },
    {
      "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.24"
    },
    {
      "name": "ecs.capability.task-eni-trunking"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.25"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.26"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.27"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.28"
    },
    {
      "name": "com.amazonaws.ecs.capability.privileged-container"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.29"
    },
    {
      "name": "ecs.cpu-architecture",
      "value": "x86_64"
    },
    {
      "name": "com.amazonaws.ecs.capability.ecr-auth"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.20"
```

```
    },
    {
      "name": "ecs.os-type",
      "value": "linux"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.21"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.22"
    },
    {
      "name": "ecs.capability.task-eia"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.23"
    },
    {
      "name": "ecs.capability.private-registry-
authentication.secretsmanager"
    },
    {
      "name": "com.amazonaws.ecs.capability.logging-driver.syslog"
    },
    {
      "name": "com.amazonaws.ecs.capability.logging-driver.json-file"
    },
    {
      "name": "ecs.capability.execution-role-awslogs"
    },
    {
      "name": "ecs.vpc-id",
      "value": "vpc-1234"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.17"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
    },
    {
```

```
        "name": "ecs.capability.task-eni"
      },
      {
        "name": "ecs.capability.execution-role-ecr-pull"
      },
      {
        "name": "ecs.capability.container-health-check"
      },
      {
        "name": "ecs.subnet-id",
        "value": "subnet-1234"
      },
      {
        "name": "ecs.instance-type",
        "value": "c5.large"
      },
      {
        "name": "com.amazonaws.ecs.capability.task-iam-role-network-
host"
      },
      {
        "name": "ecs.capability.container-ordering"
      },
      {
        "name": "ecs.capability.cni-plugin-version",
        "value": "91ccef8-2019.06.0"
      },
      {
        "name": "ecs.capability.pid-ipc-namespace-sharing"
      },
      {
        "name": "ecs.capability.secrets.ssm.environment-variables"
      },
      {
        "name": "com.amazonaws.ecs.capability.task-iam-role"
      }
    ],
    "registeredAt": 1560788724.507,
    "attachments": [],
    "tags": []
  }
],
"failures": []
```

```
}
```

- Untuk detail API, lihat [UpdateContainerInstancesStatedi Referensi AWS CLI Perintah](#).

## update-service-primary-task-set

Contoh kode berikut menunjukkan cara menggunakan `update-service-primary-task-set`.

### AWS CLI

Untuk memperbarui set tugas utama untuk layanan

`update-service-primary-task-set` Contoh berikut memperbarui tugas utama yang ditetapkan untuk layanan yang ditentukan.

```
aws ecs update-service-primary-task-set \  
  --cluster MyCluster \  
  --service MyService \  
  --primary-task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789
```

Output:

```
{  
  "taskSet": {  
    "id": "ecs-svc/1234567890123456789",  
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
    "status": "PRIMARY",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
sample-fargate:2",  
    "computedDesiredCount": 1,  
    "pendingCount": 0,  
    "runningCount": 0,  
    "createdAt": 1557128360.711,  
    "updatedAt": 1557129412.653,  
    "launchType": "EC2",  
    "networkConfiguration": {  
      "awsvpcConfiguration": {  
        "subnets": [  
          "subnet-12344321"  
        ],  
      },  
    },  
  },  
}
```



```
        "securityGroups": [
            "sg-12344312"
        ],
        "assignPublicIp": "DISABLED"
    }
},
"loadBalancers": [],
"serviceRegistries": [],
"scale": {
    "value": 50.0,
    "unit": "PERCENT"
},
"stabilityStatus": "STABILIZING",
"stabilityStatusAt": 1557129279.914
}
}
```

- Untuk detail API, lihat [UpdateServicePrimaryTaskSet](#) di Referensi AWS CLI Perintah.

## update-service

Contoh kode berikut menunjukkan cara menggunakan `update-service`.

### AWS CLI

Contoh 1: Untuk mengubah definisi tugas yang digunakan dalam layanan

`update-service` Contoh berikut memperbarui `my-http-service` layanan untuk menggunakan definisi `amazon-ecs-sample` tugas.

```
aws ecs update-service --service my-http-service --task-definition amazon-ecs-sample
```

Contoh 2: Untuk mengubah jumlah tugas dalam suatu layanan

`update-service` Contoh berikut memperbarui jumlah tugas yang diinginkan dari layanan `my-http-service` ke 3.

```
aws ecs update-service --service my-http-service --desired-count 3
```

Untuk informasi selengkapnya, lihat [Memperbarui Layanan](#) di Panduan Pengembang Amazon ECS.

- Untuk detail API, lihat [UpdateService](#) di Referensi AWS CLI Perintah.

## update-task-set

Contoh kode berikut menunjukkan cara menggunakan `update-task-set`.

### AWS CLI

Untuk memperbarui set tugas

`update-task-set` Contoh berikut memperbarui set tugas untuk menyesuaikan skala.

```
aws ecs update-task-set \  
  --cluster MyCluster \  
  --service MyService \  
  --task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-  
svc/1234567890123456789 \  
  --scale value=50,unit=PERCENT
```

Output:

```
{  
  "taskSet": {  
    "id": "ecs-svc/1234567890123456789",  
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
    "status": "ACTIVE",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
sample-fargate:2",  
    "computedDesiredCount": 0,  
    "pendingCount": 0,  
    "runningCount": 0,  
    "createdAt": 1557128360.711,  
    "updatedAt": 1557129279.914,  
    "launchType": "EC2",  
    "networkConfiguration": {  
      "awsvpcConfiguration": {  
        "subnets": [  
          "subnet-12344321"  
        ],  
        "securityGroups": [  
          "sg-12344321"  
        ]  
      }  
    }  
  }  
}
```

```
        ],
        "assignPublicIp": "DISABLED"
    }
},
"loadBalancers": [],
"serviceRegistries": [],
"scale": {
    "value": 50.0,
    "unit": "PERCENT"
},
"stabilityStatus": "STABILIZING",
"stabilityStatusAt": 1557129279.914
}
}
```

- Untuk detail API, lihat [UpdateTaskSet](#) di Referensi AWS CLI Perintah.

## Amazon EFS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Amazon EFS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-file-system**

Contoh kode berikut menunjukkan cara menggunakan `create-file-system`.

## AWS CLI

Untuk membuat sistem file terenkripsi

`create-file-system` Contoh berikut membuat sistem file terenkripsi menggunakan CMK default. Itu juga menambahkan `tagName=my-file-system`.

```
aws efs create-file-system \  
  --performance-mode generalPurpose \  
  --throughput-mode bursting \  
  --encrypted \  
  --tags Key=Name,Value=my-file-system
```

Output:

```
{  
  "OwnerId": "123456789012",  
  "CreationToken": "console-d7f56c5f-e433-41ca-8307-9d9c0example",  
  "FileSystemId": "fs-c7a0456e",  
  "FileSystemArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-system/  
fs-48499b4d",  
  "CreationTime": 1595286880.0,  
  "LifecycleState": "creating",  
  "Name": "my-file-system",  
  "NumberOfMountTargets": 0,  
  "SizeInBytes": {  
    "Value": 0,  
    "ValueInIA": 0,  
    "ValueInStandard": 0  
  },  
  "PerformanceMode": "generalPurpose",  
  "Encrypted": true,  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/a59b3472-e62c-42e4-  
adcf-30d92example",  
  "ThroughputMode": "bursting",  
  "Tags": [  
    {  
      "Key": "Name",  
      "Value": "my-file-system"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat sistem file Amazon EFS](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [CreateFileSystem](#) di Referensi AWS CLI Perintah.

## create-mount-target

Contoh kode berikut menunjukkan cara menggunakan `create-mount-target`.

### AWS CLI

Untuk membuat target mount

`create-mount-target` Contoh berikut membuat target mount untuk sistem file yang ditentukan.

```
aws efs create-mount-target \  
  --file-system-id fs-c7a0456e \  
  --subnet-id subnet-02bf4c428bexample \  
  --security-groups sg-068f739363example
```

Output:

```
{  
  "OwnerId": "123456789012",  
  "MountTargetId": "fsmt-f9a14450",  
  "FileSystemId": "fs-c7a0456e",  
  "SubnetId": "subnet-02bf4c428bexample",  
  "LifeCycleState": "creating",  
  "IpAddress": "10.0.1.24",  
  "NetworkInterfaceId": "eni-02d542216aexample",  
  "AvailabilityZoneId": "use2-az2",  
  "AvailabilityZoneName": "us-east-2b",  
  "VpcId": "vpc-0123456789abcdef0"  
}
```

Untuk informasi selengkapnya, lihat [Membuat target mount](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [CreateMountTarget](#) di Referensi AWS CLI Perintah.

## delete-file-system

Contoh kode berikut menunjukkan cara menggunakan `delete-file-system`.

### AWS CLI

Untuk menghapus sistem file

`delete-file-system` Contoh berikut menghapus sistem file yang ditentukan.

```
aws efs delete-file-system \  
  --file-system-id fs-c7a0456e
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus sistem file Amazon EFS](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [DeleteFileSystem](#) di Referensi AWS CLI Perintah.

## delete-mount-target

Contoh kode berikut menunjukkan cara menggunakan `delete-mount-target`.

### AWS CLI

Untuk menghapus target pemasangan

`delete-mount-target` Contoh berikut menghapus target mount yang ditentukan.

```
aws efs delete-mount-target \  
  --mount-target-id fsmt-f9a14450
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat target mount](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [DeleteMountTarget](#) di Referensi AWS CLI Perintah.

## describe-file-systems

Contoh kode berikut menunjukkan cara menggunakan `describe-file-systems`.

## AWS CLI

Untuk mendeskripsikan sistem file

`describe-file-systems` Contoh berikut menjelaskan sistem file yang ditentukan.

```
aws efs describe-file-systems \  
  --file-system-id fs-c7a0456e
```

Output:

```
{  
  "FileSystems": [  
    {  
      "OwnerId": "123456789012",  
      "CreationToken": "console-d7f56c5f-e433-41ca-8307-9d9c0example",  
      "FileSystemId": "fs-c7a0456e",  
      "FileSystemArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-  
system/fs-48499b4d",  
      "CreationTime": 1595286880.0,  
      "LifecycleState": "available",  
      "Name": "my-file-system",  
      "NumberOfMountTargets": 3,  
      "SizeInBytes": {  
        "Value": 6144,  
        "Timestamp": 1600991437.0,  
        "ValueInIA": 0,  
        "ValueInStandard": 6144  
      },  
      "PerformanceMode": "generalPurpose",  
      "Encrypted": true,  
      "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/a59b3472-e62c-42e4-  
adcf-30d92example",  
      "ThroughputMode": "bursting",  
      "Tags": [  
        {  
          "Key": "Name",  
          "Value": "my-file-system"  
        }  
      ]  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola sistem file Amazon EFS](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [DescribeFileSystems](#) di Referensi AWS CLI Perintah.

## describe-mount-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-mount-targets`.

### AWS CLI

Untuk menggambarkan target mount

`describe-mount-targets` Contoh berikut menjelaskan target mount yang ditentukan.

```
aws efs describe-mount-targets \  
  --mount-target-id fsmt-f9a14450
```

Output:

```
{  
  "MountTargets": [  
    {  
      "OwnerId": "123456789012",  
      "MountTargetId": "fsmt-f9a14450",  
      "FileSystemId": "fs-c7a0456e",  
      "SubnetId": "subnet-02bf4c428bexample",  
      "LifeCycleState": "creating",  
      "IpAddress": "10.0.1.24",  
      "NetworkInterfaceId": "eni-02d542216aexample",  
      "AvailabilityZoneId": "use2-az2",  
      "AvailabilityZoneName": "us-east-2b",  
      "VpcId": "vpc-0123456789abcdef0"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat target mount](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [DescribeMountTargets](#) di Referensi AWS CLI Perintah.



## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Untuk mendeskripsikan tag untuk sistem file

`describe-tags` Contoh berikut menjelaskan tag untuk sistem file yang ditentukan.

```
aws efs describe-tags \
  --file-system-id fs-c7a0456e
```

Output:

```
{
  "Tags": [
    {
      "Key": "Name",
      "Value": "my-file-system"
    },
    {
      "Key": "Department",
      "Value": "Business Intelligence"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola tag sistem file](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mengambil tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mengambil tag yang terkait dengan sistem file yang ditentukan.

```
aws efs list-tags-for-resource \  
  --resource-id fs-c7a0456e
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Name",  
      "Value": "my-file-system"  
    },  
    {  
      "Key": "Department",  
      "Value": "Business Intelligence"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola tag sistem file](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan tag `Department=Business Intelligence` ke sistem file yang ditentukan.

```
aws efs tag-resource \  
  --resource-id fs-c7a0456e \  
  --tags Key=Department,Value="Business Intelligence"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola tag sistem file](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dengan kunci Department tag dari sistem file yang ditentukan.

```
aws efs untag-resource \  
  --resource-id fs-c7a0456e \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola tag sistem file](#) di Panduan Pengguna Amazon Elastic File System.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Amazon EKS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface dengan Amazon EKS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **associate-encryption-config**

Contoh kode berikut menunjukkan cara menggunakan `associate-encryption-config`.

#### AWS CLI

Untuk mengaitkan konfigurasi enkripsi ke cluster yang ada

`associate-encryption-config` Contoh berikut mengaktifkan enkripsi pada kluster EKS yang ada yang belum mengaktifkan enkripsi.

```
aws eks associate-encryption-config \
  --cluster-name my-eks-cluster \
  --encryption-config '[{"resources":["secrets"],"provider":
{"keyArn":"arn:aws:kms:region-code:account:key/key"}]'
```

#### Output:

```
{
  "update": {
    "id": "3141b835-8103-423a-8e68-12c2521ffa4d",
    "status": "InProgress",
    "type": "AssociateEncryptionConfig",
    "params": [
      {
        "type": "EncryptionConfig",
        "value": "[{\\"resources\\":[\"secrets\\\"],\\"provider\\":{\\"keyArn\\":
\\\"arn:aws:kms:region-code:account:key/key\\\"}}]"
      }
    ],
    "createdAt": "2024-03-14T11:01:26.297000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan enkripsi rahasia pada kluster yang ada](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [AssociateEncryptionConfig](#) di Referensi AWS CLI Perintah.

## associate-identity-provider-config

Contoh kode berikut menunjukkan cara menggunakan `associate-identity-provider-config`.

### AWS CLI

Mengaitkan penyedia identitas ke Amazon EKS Cluster Anda

`associate-identity-provider-config` Contoh berikut mengaitkan penyedia identitas ke Amazon EKS Cluster Anda.

```
aws eks associate-identity-provider-config \
  --cluster-name my-eks-cluster \
  --oidc 'identityProviderConfigName=my-identity-provider,issuerUrl=https://
oidc.eks.us-east-2.amazonaws.com/
id/38D6A4619A0A69E342B113ED7F1A7652,clientId=kubernetes,usernameClaim=email,usernamePrefix=
username-prefix,groupsClaim=my-claim,groupsPrefix=my-groups-
prefix,requiredClaims={Claim1=value1,Claim2=value2}' \
  --tags env=dev
```

Output:

```
{
  "update": {
    "id": "8c6c1bef-61fe-42ac-a242-89412387b8e7",
    "status": "InProgress",
    "type": "AssociateIdentityProviderConfig",
    "params": [
      {
        "type": "IdentityProviderConfig",
        "value": "[{\"type\":\"oidc\",\"name\":\"my-identity-provider\"}]"
      }
    ],
    "createdAt": "2024-04-11T13:46:49.648000-04:00",
    "errors": []
  },
  "tags": {
    "env": "dev"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna untuk kluster Anda dari penyedia identitas OpenID Connect - Mengaitkan penyedia identitas OIDC di Panduan Pengguna Amazon EKS](#).

- Untuk detail API, lihat [AssociateIdentityProviderConfig](#) di Referensi AWS CLI Perintah.

## create-addon

Contoh kode berikut menunjukkan cara menggunakan `create-addon`.

### AWS CLI

Contoh 1: Untuk membuat add-on Amazon EKS dengan versi default yang kompatibel untuk masing-masing versi kluster EKS

Perintah `create-addon` contoh berikut membuat add-on Amazon EKS dengan versi default yang kompatibel untuk masing-masing versi kluster EKS.

```
aws eks create-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name my-eks-addon \  
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name
```

### Output:

```
{  
  "addon": {  
    "addonName": "my-eks-addon",  
    "clusterName": "my-eks-cluster",  
    "status": "CREATING",  
    "addonVersion": "v1.15.1-eksbuild.1",  
    "health": {  
      "issues": []  
    },  
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/1ec71ee1-b9c2-8915-4e17-e8be0a55a149",  
    "createdAt": "2024-03-14T12:20:03.264000-04:00",  
    "modifiedAt": "2024-03-14T12:20:03.283000-04:00",  
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 2: Untuk membuat add-on Amazon EKS dengan versi add-on tertentu

`create-addon` Contoh perintah berikut membuat add-on Amazon EKS dengan versi add-on tertentu.

```
aws eks create-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name my-eks-addon \  
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \  
  --addon-version v1.16.4-eksbuild.2
```

Output:

```
{  
  "addon": {  
    "addonName": "my-eks-addon",  
    "clusterName": "my-eks-cluster",  
    "status": "CREATING",  
    "addonVersion": "v1.16.4-eksbuild.2",  
    "health": {  
      "issues": []  
    },  
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/34c71ee6-7738-6c8b-c6bd-3921a176b5ff",  
    "createdAt": "2024-03-14T12:30:24.507000-04:00",  
    "modifiedAt": "2024-03-14T12:30:24.521000-04:00",  
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 3: Untuk membuat add-on Amazon EKS dengan nilai konfigurasi khusus dan menyelesaikan detail konflik

Perintah `create-addon` contoh berikut membuat add-on Amazon EKS dengan nilai konfigurasi khusus dan menyelesaikan detail konflik.

```
aws eks create-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name my-eks-addon \  
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \  
  --addon-version v1.16.4-eksbuild.2 \  
  --configuration-values '{"resources":{"limits":{"cpu":"100m"}}}' \  
  --resolve-conflicts OVERWRITE
```

### Output:

```
{  
  "addon": {  
    "addonName": "my-eks-addon",  
    "clusterName": "my-eks-cluster",  
    "status": "CREATING",  
    "addonVersion": "v1.16.4-eksbuild.2",  
    "health": {  
      "issues": []  
    },  
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/a6c71ee9-0304-9237-1be8-25af1b0f1ffb",  
    "createdAt": "2024-03-14T12:35:58.313000-04:00",  
    "modifiedAt": "2024-03-14T12:35:58.327000-04:00",  
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "tags": {},  
    "configurationValues": "{\"resources\":{\"limits\":{\"cpu\":\"100m\"}}}"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 4: Untuk membuat add-on Amazon EKS dengan file nilai konfigurasi JSON kustom

Perintah `create-addon` contoh berikut membuat add-on Amazon EKS dengan nilai konfigurasi khusus dan menyelesaikan detail konflik.

```
aws eks create-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name my-eks-addon \  
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \  
  --addon-version v1.16.4-eksbuild.2 \  
  --configuration-values '{"resources":{"limits":{"cpu":"100m"}}}' \  
  --resolve-conflicts OVERWRITE
```



```
--configuration-values 'file://configuration-values.json' \
--resolve-conflicts OVERWRITE \
--tags '{"eks-addon-key-1": "value-1" , "eks-addon-key-2": "value-2"}'
```

Isi dari `configuration-values.json`:

```
{
  "resources": {
    "limits": {
      "cpu": "150m"
    }
  },
  "env": {
    "AWS_VPC_K8S_CNI_LOGLEVEL": "ERROR"
  }
}
```

Output:

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "CREATING",
    "addonVersion": "v1.16.4-eksbuild.2",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/d8c71ef8-fbd8-07d0-fb32-6a7be19eeced",
    "createdAt": "2024-03-14T13:10:51.763000-04:00",
    "modifiedAt": "2024-03-14T13:10:51.777000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {
      "eks-addon-key-1": "value-1",
      "eks-addon-key-2": "value-2"
    },
    "configurationValues": "{\n  \"resources\": {\n    \"limits\": {\n      \"cpu\": \"150m\"\n    }\n  },\n  \"env\": {\n    \"AWS_VPC_K8S_CNI_LOGLEVEL\": \"ERROR\"\n  }\n}"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 5: Untuk membuat add-on Amazon EKS dengan file nilai konfigurasi YAMAL kustom

Perintah `create-addon` contoh berikut membuat add-on Amazon EKS dengan nilai konfigurasi khusus dan menyelesaikan detail konflik.

```
aws eks create-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon \
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
  --addon-version v1.16.4-eksbuild.2 \
  --configuration-values 'file://configuration-values.yaml' \
  --resolve-conflicts OVERWRITE \
  --tags '{"eks-addon-key-1": "value-1" , "eks-addon-key-2": "value-2"}'
```

Isi dari `configuration-values.yaml`:

```
resources:
  limits:
    cpu: '100m'
env:
  AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'
```

Output:

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "CREATING",
    "addonVersion": "v1.16.4-eksbuild.2",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/d4c71efb-3909-6f36-a548-402cd4b5d59e",
    "createdAt": "2024-03-14T13:15:45.220000-04:00",
    "modifiedAt": "2024-03-14T13:15:45.237000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {
      "eks-addon-key-3": "value-3",

```

```
        "eks-addon-key-4": "value-4"
    },
    "configurationValues": "resources:\n    limits:\n        cpu: '100m'\n    nenv:\n        AWS_VPC_K8S_CNI_LOGLEVEL: 'INFO'"
}
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [CreateAddon](#) di Referensi AWS CLI Perintah.

## create-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-cluster`.

### AWS CLI

Untuk membuat cluster baru

Perintah contoh ini membuat klaster bernama `prod` di wilayah default Anda.

Perintah:

```
aws eks create-cluster --name prod \
--role-arn arn:aws:iam::012345678910:role/eks-service-role-
AWSServiceRoleForAmazonEKS-J7ONKE3BQ4PI \
--resources-vpc-config subnetIds=subnet-6782e71e,subnet-
e7e761ac,securityGroupIds=sg-6979fe18
```

Output:

```
{
  "cluster": {
    "name": "prod",
    "arn": "arn:aws:eks:us-west-2:012345678910:cluster/prod",
    "createdAt": 1527808069.147,
    "version": "1.10",
    "roleArn": "arn:aws:iam::012345678910:role/eks-service-role-
AWSServiceRoleForAmazonEKS-J7ONKE3BQ4PI",
    "resourcesVpcConfig": {
      "subnetIds": [
        "subnet-6782e71e",
```

```

        "subnet-e7e761ac"
      ],
      "securityGroupIds": [
        "sg-6979fe18"
      ],
      "vpcId": "vpc-950809ec"
    },
    "status": "CREATING",
    "certificateAuthority": {}
  }
}

```

Untuk membuat kluster baru dengan akses endpoint pribadi dan logging diaktifkan

Perintah contoh ini membuat kluster bernama `example` di wilayah default Anda dengan akses titik akhir publik dinonaktifkan, akses titik akhir pribadi diaktifkan, dan semua jenis logging diaktifkan.

Perintah:

```

aws eks create-cluster --name example --kubernetes-version 1.12 \
--role-arn arn:aws:iam::012345678910:role/example-cluster-ServiceRole-1XWBQWYSFRE2Q \
--resources-vpc-config
  subnetIds=subnet-0a188dccd2f9a632f,subnet-09290d93da4278664,subnet-0f21dd86e0e91134a,subnet-
  \
--logging '{"clusterLogging":[{"types":
["api","audit","authenticator","controllerManager","scheduler"],"enabled":true}]}'

```

Output:

```

{
  "cluster": {
    "name": "example",
    "arn": "arn:aws:eks:us-west-2:012345678910:cluster/example",
    "createdAt": 1565804921.901,
    "version": "1.12",
    "roleArn": "arn:aws:iam::012345678910:role/example-cluster-
ServiceRole-1XWBQWYSFRE2Q",
    "resourcesVpcConfig": {
      "subnetIds": [
        "subnet-0a188dccd2f9a632f",
        "subnet-09290d93da4278664",
        "subnet-0f21dd86e0e91134a",

```

```

        "subnet-0173dead68481a583",
        "subnet-051f70a57ed6fcab6",
        "subnet-01322339c5c7de9b4"
    ],
    "securityGroupIds": [
        "sg-0c5b580845a031c10"
    ],
    "vpcId": "vpc-0f622c01f68d4afec",
    "endpointPublicAccess": false,
    "endpointPrivateAccess": true
},
"logging": {
    "clusterLogging": [
        {
            "types": [
                "api",
                "audit",
                "authenticator",
                "controllerManager",
                "scheduler"
            ],
            "enabled": true
        }
    ]
},
"status": "CREATING",
"certificateAuthority": {},
"platformVersion": "eks.3"
}
}

```

- Untuk detail API, lihat [CreateCluster](#) di Referensi AWS CLI Perintah.

## create-fargate-profile

Contoh kode berikut menunjukkan cara menggunakan `create-fargate-profile`.

### AWS CLI

Contoh 1: Buat Profil EKS Fargate untuk pemilih dengan namespace

`create-fargate-profile` Contoh berikut membuat Profil EKS Fargate untuk pemilih dengan namespace.

```
aws eks create-fargate-profile \  
  --cluster-name my-eks-cluster \  
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \  
  --fargate-profile-name my-fargate-profile \  
  --selectors '[{"namespace": "default"}]'
```

### Output:

```
{  
  "fargateProfile": {  
    "fargateProfileName": "my-fargate-profile",  
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-  
eks-cluster/my-fargate-profile/a2c72bca-318e-abe8-8ed1-27c6d4892e9e",  
    "clusterName": "my-eks-cluster",  
    "createdAt": "2024-03-19T12:38:47.368000-04:00",  
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "subnets": [  
      "subnet-09d912bb63ef21b9a",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-0e2907431c9988b72"  
    ],  
    "selectors": [  
      {  
        "namespace": "default"  
      }  
    ],  
    "status": "CREATING",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Profil AWS Fargate - Membuat profil Fargate di Panduan Pengguna Amazon EKS](#).

Contoh 2: Buat Profil EKS Fargate untuk pemilih dengan namespace dan label

`create-fargate-profile` Contoh berikut membuat Profil EKS Fargate untuk pemilih dengan namespace dan label.

```
aws eks create-fargate-profile \  
  --cluster-name my-eks-cluster \  
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \  
  --fargate-profile-name my-fargate-profile \  
  --selectors '[{"namespace": "default", "label": "my-label"}]'
```

```
--fargate-profile-name my-fargate-profile \
--selectors '[{"namespace": "default", "labels": {"labelname1":
"labelvalue1"}}]'
```

### Output:

```
{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/88c72bc7-e8a4-fa34-44e4-2f1397224bb3",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T12:33:48.125000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "default",
        "labels": {
          "labelname1": "labelvalue1"
        }
      }
    ],
    "status": "CREATING",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Profil AWS Fargate - Membuat profil Fargate di Panduan Pengguna Amazon EKS](#).

Contoh 3: Buat Profil Fargate EKS untuk pemilih dengan namespace dan label, bersama dengan ID subnet untuk meluncurkan Pod ke dalamnya.

`create-fargate-profile` Contoh berikut membuat Profil EKS Fargate untuk pemilih dengan namespace dan label, bersama dengan ID subnet untuk meluncurkan Pod ke dalamnya.

```
aws eks create-fargate-profile \
```

```
--cluster-name my-eks-cluster \  
--pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \  
--fargate-profile-name my-fargate-profile \  
--selectors '[{"namespace": "default", "labels": {"labelname1":  
"labelvalue1"}}]' \  
--subnets ["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",  
"subnet-0e2907431c9988b72"]'
```

### Output:

```
{  
  "fargateProfile": {  
    "fargateProfileName": "my-fargate-profile",  
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-  
eks-cluster/my-fargate-profile/e8c72bc8-e87b-5eb6-57cb-ed4fe57577e3",  
    "clusterName": "my-eks-cluster",  
    "createdAt": "2024-03-19T12:35:58.640000-04:00",  
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "subnets": [  
      "subnet-09d912bb63ef21b9a",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-0e2907431c9988b72"  
    ],  
    "selectors": [  
      {  
        "namespace": "default",  
        "labels": {  
          "labelname1": "labelvalue1"  
        }  
      }  
    ],  
    "status": "CREATING",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Profil AWS Fargate - Membuat profil Fargate di Panduan Pengguna Amazon EKS](#).

Contoh 4: Buat Profil Fargate EKS untuk pemilih dengan beberapa namespace dan label, bersama dengan ID subnet untuk meluncurkan Pod ke



`create-fargate-profile` Contoh berikut membuat Profil Fargate EKS untuk pemilih dengan beberapa namespace dan label, bersama dengan ID subnet untuk meluncurkan Pod ke dalamnya.

```
aws eks create-fargate-profile \  
  --cluster-name my-eks-cluster \  
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \  
  --fargate-profile-name my-fargate-profile \  
  --selectors '[{"namespace": "default1", "labels": {"labelname1": "labelvalue1",  
"labelname2": "labelvalue2"}}, {"namespace": "default2", "labels": {"labelname1":  
"labelvalue1", "labelname2": "labelvalue2"}}]' \  
  --subnets ["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",  
"subnet-0e2907431c9988b72"] \  
  --tags '{"eks-fargate-profile-key-1": "value-1" , "eks-fargate-profile-key-2":  
"value-2"}'
```

Output:

```
{  
  "fargateProfile": {  
    "fargateProfileName": "my-fargate-profile",  
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-  
eks-cluster/my-fargate-profile/4cc72bbf-b766-8ee6-8d29-e62748feb3cd",  
    "clusterName": "my-eks-cluster",  
    "createdAt": "2024-03-19T12:15:55.271000-04:00",  
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "subnets": [  
      "subnet-09d912bb63ef21b9a",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-0e2907431c9988b72"  
    ],  
    "selectors": [  
      {  
        "namespace": "default1",  
        "labels": {  
          "labelname2": "labelvalue2",  
          "labelname1": "labelvalue1"  
        }  
      },  
      {  
        "namespace": "default2",  
        "labels": {  
          "labelname2": "labelvalue2",
```

```

        "labelname1": "labelvalue1"
      }
    }
  ],
  "status": "CREATING",
  "tags": {
    "eks-fargate-profile-key-2": "value-2",
    "eks-fargate-profile-key-1": "value-1"
  }
}
}

```

Untuk informasi selengkapnya, lihat [Profil AWS Fargate - Membuat profil Fargate di Panduan Pengguna Amazon EKS](#).

Contoh 5: Buat Profil EKS Fargate dengan pemilih wildcard untuk namespace dan label, bersama dengan ID subnet untuk meluncurkan Pod ke

`create-fargate-profile` Contoh berikut membuat Profil Fargate EKS untuk pemilih dengan beberapa namespace dan label, bersama dengan ID subnet untuk meluncurkan Pod ke dalamnya.

```

aws eks create-fargate-profile \
  --cluster-name my-eks-cluster \
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
  --fargate-profile-name my-fargate-profile \
  --selectors '[{"namespace": "prod*", "labels": {"labelname*?": "*value1"}}, {"namespace": "*dev*", "labels": {"labelname*?": "*value*"}}]' \
  --subnets ["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d", "subnet-0e2907431c9988b72"] \
  --tags '{"eks-fargate-profile-key-1": "value-1" , "eks-fargate-profile-key-2": "value-2"}'

```

Output:

```

{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/e8c72bd6-5966-0bfe-b77b-1802893e5a6f",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T13:05:20.550000-04:00",
  }
}

```

```

    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "prod*",
        "labels": {
          "labelname*?": "*value1"
        }
      },
      {
        "namespace": "*dev*",
        "labels": {
          "labelname*?": "*value*"
        }
      }
    ],
    "status": "CREATING",
    "tags": {
      "eks-fargate-profile-key-2": "value-2",
      "eks-fargate-profile-key-1": "value-1"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Profil AWS Fargate - Membuat profil Fargate di Panduan Pengguna](#) Amazon EKS.

- Untuk detail API, lihat [CreateFargateProfile](#) di Referensi AWS CLI Perintah.

## create-nodegroup

Contoh kode berikut menunjukkan cara menggunakan `create-nodegroup`.

### AWS CLI

Contoh 1: Membuat grup node terkelola untuk kluster Amazon EKS

`create-nodegroup` Contoh berikut membuat grup node terkelola untuk kluster Amazon EKS.

```
aws eks create-nodegroup \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup \  
  --node-role arn:aws:iam::111122223333:role/role-name \  
  --subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d"  
"subnet-09d912bb63ef21b9a" \  
  --scaling-config minSize=1,maxSize=3,desiredSize=1 \  
  --region us-east-2
```

## Output:

```
{  
  "nodegroup": {  
    "nodegroupName": "my-eks-nodegroup",  
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-  
cluster/my-eks-nodegroup/bac7550f-b8b8-5fbb-4f3e-7502a931119e",  
    "clusterName": "my-eks-cluster",  
    "version": "1.26",  
    "releaseVersion": "1.26.12-20240329",  
    "createdAt": "2024-04-04T13:19:32.260000-04:00",  
    "modifiedAt": "2024-04-04T13:19:32.260000-04:00",  
    "status": "CREATING",  
    "capacityType": "ON_DEMAND",  
    "scalingConfig": {  
      "minSize": 1,  
      "maxSize": 3,  
      "desiredSize": 1  
    },  
    "instanceTypes": [  
      "t3.medium"  
    ],  
    "subnets": [  
      "subnet-0e2907431c9988b72", "subnet-04ad87f71c6e5ab4d",  
      "subnet-09d912bb63ef21b9a"  
    ],  
    "amiType": "AL2_x86_64",  
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",  
    "diskSize": 20,  
    "health": {  
      "issues": []  
    },  
    "updateConfig": {  
      "maxUnavailable": 1
```

```

    },
    "tags": {}
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat grup node terkelola](#) di Panduan Pengguna Amazon EKS.

Contoh 2: Membuat grup node terkelola untuk klaster Amazon EKS dengan tipe instans khusus dan ukuran disk

Contoh berikut membuat grup node terkelola untuk klaster Amazon EKS dengan tipe instans khusus dan ukuran disk.

```

aws eks create-nodegroup \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --node-role arn:aws:iam::111122223333:role/role-name \
  --subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d"
"subnet-09d912bb63ef21b9a" \
  --scaling-config minSize=1,maxSize=3,desiredSize=1 \
  --capacity-type ON_DEMAND \
  --instance-types 'm5.large' \
  --disk-size 50 \
  --region us-east-2

```

Output:

```

{
  "nodegroup": {
    "nodegroupName": "my-eks-nodegroup",
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/c0c7551b-e4f9-73d9-992c-a450fdb82322",
    "clusterName": "my-eks-cluster",
    "version": "1.26",
    "releaseVersion": "1.26.12-20240329",
    "createdAt": "2024-04-04T13:46:07.595000-04:00",
    "modifiedAt": "2024-04-04T13:46:07.595000-04:00",
    "status": "CREATING",
    "capacityType": "ON_DEMAND",
    "scalingConfig": {
      "minSize": 1,

```

```

        "maxSize": 3,
        "desiredSize": 1
    },
    "instanceTypes": [
        "m5.large"
    ],
    "subnets": [
        "subnet-0e2907431c9988b72",
        "subnet-04ad87f71c6e5ab4d",
        "subnet-09d912bb63ef21b9a"
    ],
    "amiType": "AL2_x86_64",
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",
    "diskSize": 50,
    "health": {
        "issues": []
    },
    "updateConfig": {
        "maxUnavailable": 1
    },
    "tags": {}
}
}

```

Untuk informasi selengkapnya, lihat [Membuat grup node terkelola](#) di Panduan Pengguna Amazon EKS.

Contoh 3: Membuat grup node terkelola untuk kluster Amazon EKS dengan tipe instans kustom, ukuran disk, tipe ami, tipe kapasitas, konfigurasi pembaruan, label, taint, dan tag.

`create-nodegroup` Contoh berikut membuat grup node terkelola untuk kluster Amazon EKS dengan tipe instans kustom, ukuran disk, tipe ami, tipe kapasitas, konfigurasi pembaruan, label, taints, dan tag.

```

aws eks create-nodegroup \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --node-role arn:aws:iam::111122223333:role/role-name \
  --subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d"
"subnet-09d912bb63ef21b9a" \
  --scaling-config minSize=1,maxSize=5,desiredSize=4 \
  --instance-types 't3.large' \
  --disk-size 50 \

```

```
--ami-type AL2_x86_64 \  
--capacity-type SPOT \  
--update-config maxUnavailable=2 \  
--labels '{"my-eks-nodegroup-label-1": "value-1" , "my-eks-nodegroup-label-2":  
"value-2"}' \  
--taints '{"key": "taint-key-1" , "value": "taint-value-1", "effect":  
"NO_EXECUTE"}' \  
--tags '{"my-eks-nodegroup-key-1": "value-1" , "my-eks-nodegroup-key-2":  
"value-2"}'
```

## Output:

```
{  
  "nodegroup": {  
    "nodegroupName": "my-eks-nodegroup",  
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-  
cluster/my-eks-nodegroup/88c75524-97af-0cb9-a9c5-7c0423ab5314",  
    "clusterName": "my-eks-cluster",  
    "version": "1.26",  
    "releaseVersion": "1.26.12-20240329",  
    "createdAt": "2024-04-04T14:05:07.940000-04:00",  
    "modifiedAt": "2024-04-04T14:05:07.940000-04:00",  
    "status": "CREATING",  
    "capacityType": "SPOT",  
    "scalingConfig": {  
      "minSize": 1,  
      "maxSize": 5,  
      "desiredSize": 4  
    },  
    "instanceTypes": [  
      "t3.large"  
    ],  
    "subnets": [  
      "subnet-0e2907431c9988b72",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-09d912bb63ef21b9a"  
    ],  
    "amiType": "AL2_x86_64",  
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",  
    "labels": {  
      "my-eks-nodegroup-label-2": "value-2",  
      "my-eks-nodegroup-label-1": "value-1"  
    },  
  },  
}
```

```
    "taints": [
      {
        "key": "taint-key-1",
        "value": "taint-value-1",
        "effect": "NO_EXECUTE"
      }
    ],
    "diskSize": 50,
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 2
    },
    "tags": {
      "my-eks-nodegroup-key-1": "value-1",
      "my-eks-nodegroup-key-2": "value-2"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat grup node terkelola](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [CreateNodegroup](#) di Referensi AWS CLI Perintah.

## delete-addon

Contoh kode berikut menunjukkan cara menggunakan `delete-addon`.

### AWS CLI

Contoh 1. Untuk menghapus add-on Amazon EKS tetapi mempertahankan perangkat lunak add-on di EKS Cluster

`delete-addon` Contoh perintah berikut menghapus add-on Amazon EKS tetapi mempertahankan perangkat lunak add-on di EKS Cluster.

```
aws eks delete-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon \
  --preserve
```



**Output:**

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "DELETING",
    "addonVersion": "v1.9.3-eksbuild.7",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/a8c71ed3-944e-898b-9167-c763856af4b8",
    "createdAt": "2024-03-14T11:49:09.009000-04:00",
    "modifiedAt": "2024-03-14T12:03:49.776000-04:00",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Menghapus add-on di Amazon EKS](#).

Contoh 2. Untuk menghapus add-on Amazon EKS dan juga menghapus perangkat lunak add-on dari EKS Cluster

`delete-addon` Contoh perintah berikut menghapus add-on Amazon EKS dan juga menghapus perangkat lunak add-on dari EKS Cluster.

```
aws eks delete-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon
```

**Output:**

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "DELETING",
    "addonVersion": "v1.15.1-eksbuild.1",
    "health": {
      "issues": []
    }
  }
}
```

```
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/bac71ed1-ec43-3bb6-88ea-f243cdb58954",
    "createdAt": "2024-03-14T11:45:31.983000-04:00",
    "modifiedAt": "2024-03-14T11:58:40.136000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Menghapus add-on di Amazon EKS](#).

- Untuk detail API, lihat [DeleteAddon](#) di Referensi AWS CLI Perintah.

## delete-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster`.

### AWS CLI

Hapus bidang kontrol cluster Amazon EKS

`delete-cluster` Contoh berikut menghapus bidang kontrol cluster Amazon EKS.

```
aws eks delete-cluster \
  --name my-eks-cluster
```

Output:

```
{
  "cluster": {
    "name": "my-eks-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster",
    "createdAt": "2024-03-14T11:31:44.348000-04:00",
    "version": "1.27",
    "endpoint": "https://DALSJ343KE23J3RN45653DSKJTT647TYD.y14.us-
east-2.eks.amazonaws.com",
    "roleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-cluster-
ServiceRole-zMF6CBakwwbW",
    "resourcesVpcConfig": {
      "subnetIds": [
        "subnet-0fb75d2d8401716e7",
```

```
        "subnet-02184492f67a3d0f9",
        "subnet-04098063527aab776",
        "subnet-0e2907431c9988b72",
        "subnet-04ad87f71c6e5ab4d",
        "subnet-09d912bb63ef21b9a"
    ],
    "securityGroupIds": [
        "sg-0c1327f6270afbb36"
    ],
    "clusterSecurityGroupId": "sg-01c84d09d70f39a7f",
    "vpcId": "vpc-0012b8e1cc0abb17d",
    "endpointPublicAccess": true,
    "endpointPrivateAccess": true,
    "publicAccessCidrs": [
        "0.0.0.0/0"
    ]
},
"kubernetesNetworkConfig": {
    "serviceIpv4Cidr": "10.100.0.0/16",
    "ipFamily": "ipv4"
},
"logging": {
    "clusterLogging": [
        {
            "types": [
                "api",
                "audit",
                "authenticator",
                "controllerManager",
                "scheduler"
            ],
            "enabled": true
        }
    ]
},
"identity": {
    "oidc": {
        "issuer": "https://oidc.eks.us-east-2.amazonaws.com/id/
DALSJ343KE23J3RN45653DSKJTT647TYD"
    }
},
"status": "DELETING",
"certificateAuthority": {
    "data": "XXX_CA_DATA_XXX"
}
```

```

    },
    "platformVersion": "eks.16",
    "tags": {
      "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
      "alpha.eksctl.io/cluster-name": "my-eks-cluster",
      "karpenter.sh/discovery": "my-eks-cluster",
      "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
      "auto-delete": "no",
      "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
      "EKS-Cluster-Name": "my-eks-cluster",
      "alpha.eksctl.io/cluster-oidc-enabled": "true",
      "aws:cloudformation:logical-id": "ControlPlane",
      "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
      "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
    },
    "accessConfig": {
      "authenticationMode": "API_AND_CONFIG_MAP"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Menghapus kluster Amazon EKS](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [DeleteCluster](#) di Referensi AWS CLI Perintah.

## delete-fargate-profile

Contoh kode berikut menunjukkan cara menggunakan `delete-fargate-profile`.

### AWS CLI

Contoh 1: Buat Profil EKS Fargate untuk pemilih dengan namespace

`delete-fargate-profile` Contoh berikut membuat Profil EKS Fargate untuk pemilih dengan namespace.

```

aws eks delete-fargate-profile \
  --cluster-name my-eks-cluster \
  --fargate-profile-name my-fargate-profile

```

**Output:**

```
{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/1ac72bb3-3fc6-2631-f1e1-98bff53bed62",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T11:48:39.975000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "default",
        "labels": {
          "foo": "bar"
        }
      }
    ],
    "status": "DELETING",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Profil AWS Fargate - Menghapus Fargate](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [DeleteFargateProfile](#) di Referensi AWS CLI Perintah.

**delete-nodegroup**

Contoh kode berikut menunjukkan cara menggunakan `delete-nodegroup`.

**AWS CLI**

Contoh 1: Menghapus grup node terkelola untuk kluster Amazon EKS

`delete-nodegroup` Contoh berikut menghapus grup node terkelola untuk kluster Amazon EKS.

```
aws eks delete-nodegroup \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup
```

### Output:

```
{  
  "nodegroup": {  
    "nodegroupName": "my-eks-nodegroup",  
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-  
cluster/my-eks-nodegroup/1ec75f5f-0e21-dcc0-b46e-f9c442685cd8",  
    "clusterName": "my-eks-cluster",  
    "version": "1.26",  
    "releaseVersion": "1.26.12-20240329",  
    "createdAt": "2024-04-08T13:25:15.033000-04:00",  
    "modifiedAt": "2024-04-08T13:25:31.252000-04:00",  
    "status": "DELETING",  
    "capacityType": "SPOT",  
    "scalingConfig": {  
      "minSize": 1,  
      "maxSize": 5,  
      "desiredSize": 4  
    },  
    "instanceTypes": [  
      "t3.large"  
    ],  
    "subnets": [  
      "subnet-0e2907431c9988b72",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-09d912bb63ef21b9a"  
    ],  
    "amiType": "AL2_x86_64",  
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",  
    "labels": {  
      "my-eks-nodegroup-label-2": "value-2",  
      "my-eks-nodegroup-label-1": "value-1"  
    },  
    "taints": [  
      {  
        "key": "taint-key-1",  
        "value": "taint-value-1",  
        "effect": "NO_EXECUTE"  
      }  
    ]  
  }  
}
```

```
    ],
    "diskSize": 50,
    "health": {
      "issues": []
    },
    },
    "updateConfig": {
      "maxUnavailable": 2
    },
    },
    "tags": {
      "my-eks-nodegroup-key-1": "value-1",
      "my-eks-nodegroup-key-2": "value-2"
    }
  }
}
```

- Untuk detail API, lihat [DeleteNodegroup](#) di Referensi AWS CLI Perintah.

## deregister-cluster

Contoh kode berikut menunjukkan cara menggunakan `deregister-cluster`.

### AWS CLI

Untuk membatalkan pendaftaran cluster yang terhubung untuk menghapusnya dari bidang kontrol Amazon EKS

`deregister-cluster` Contoh berikut membatalkan pendaftaran cluster yang terhubung untuk menghapusnya dari bidang kontrol Amazon EKS.

```
aws eks deregister-cluster \
  --name my-eks-anywhere-cluster
```

### Output:

```
{
  "cluster": {
    "name": "my-eks-anywhere-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-anywhere-cluster",
    "createdAt": "2024-04-12T12:38:37.561000-04:00",
    "status": "DELETING",
    "tags": {},
    "connectorConfig": {
```

```

    "activationId": "dfb5ad28-13c3-4e26-8a19-5b2457638c74",
    "activationExpiry": "2024-04-15T12:38:37.082000-04:00",
    "provider": "EKS_ANYWHERE",
    "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSCoordinatorAgentRole"
  }
}
}

```

Untuk informasi selengkapnya, lihat [Membatalkan pendaftaran kluster di Panduan Pengguna Amazon EKS](#).

- Untuk detail API, lihat [DeregisterCluster](#) di Referensi AWS CLI Perintah.

## describe-addon-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-addon-versions`.

### AWS CLI

Contoh 1: Daftar semua addon yang tersedia untuk EKS Cluster

`describe-addon-versions` Contoh berikut mencantumkan semua AWS addons yang tersedia.

```

aws eks describe-addon-versions \
  --query 'sort_by(addons &owner)[].{publisher: publisher, owner: owner,
  addonName: addonName, type: type}' \
  --output table

```

Output:

```

-----
|                                     DescribeAddonVersions                                     |
|                                     |                                     |                                     |
+-----+-----+-----+-----+
|                                     |                                     |                                     | | |
|          addonName                 |          owner                 |          publisher                 |
|          |          type            |          |          |          |
+-----+-----+-----+-----+
| vpc-cni                            | aws                            | eks                                | | | |
|   | networking                      |   |                               |   |                               |
| snapshot-controller                | aws                            | eks                                |
|   | storage                          |   |                               |   |                               |

```



kube-proxy	aws	eks
networking		
eks-pod-identity-agent	aws	eks
security		
coredns	aws	eks
networking		
aws-mountpoint-s3-csi-driver	aws	s3
storage		
aws-guardduty-agent	aws	eks
security		
aws-efs-csi-driver	aws	eks
storage		
aws-ebs-csi-driver	aws	eks
storage		
amazon-cloudwatch-observability	aws	eks
observability		
adot	aws	eks
observability		
upwind-security_upwind-operator	aws-marketplace	Upwind Security
security		
upbound_universal-crossplane	aws-marketplace	upbound
infra-management		
tetrade-io_istio-distro	aws-marketplace	tetrade-io
policy-management		
teleport_teleport	aws-marketplace	teleport
policy-management		
stormforge_optimize-live	aws-marketplace	StormForge
cost-management		
splunk_splunk-otel-collector-chart	aws-marketplace	Splunk
monitoring		
solo-io_istio-distro	aws-marketplace	Solo.io
service-mesh		
rafay-systems_rafay-operator	aws-marketplace	rafay-systems
kubernetes-management		
new-relic_kubernetes-operator	aws-marketplace	New Relic
observability		
netapp_trident-operator	aws-marketplace	NetApp Inc.
storage		
leaksignal_leakagent	aws-marketplace	leaksignal
monitoring		
kubecost_kubecost	aws-marketplace	kubecost
cost-management		
kong_konnect-ri	aws-marketplace	kong
ingress-service-type		

```

| kasten_k10 | aws-marketplace | Kasten by Veeam
| data-protection |
| haproxy-technologies_kubernetes-ingress-ee | aws-marketplace | HAProxy
Technologies | ingress-controller |
| groundcover_agent | aws-marketplace | groundcover
| monitoring |
| grafana-labs_kubernetes-monitoring | aws-marketplace | Grafana Labs
| monitoring |
| factorhouse_kpow | aws-marketplace | factorhouse
| monitoring |
| dynatrace_dynatrace-operator | aws-marketplace | dynatrace
| monitoring |
| datree_engine-pro | aws-marketplace | datree
| policy-management |
| datadog_operator | aws-marketplace | Datadog
| monitoring |
| cribl_cribledge | aws-marketplace | Cribl
| observability |
| calyptia_fluent-bit | aws-marketplace | Calyptia Inc
| observability |
| accuknox_kubearmor | aws-marketplace | AccuKnox
| security |
+-----+-----+
+-----+-----+

```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 2: Buat daftar semua addon yang tersedia untuk versi Kubernetes tertentu yang didukung untuk EKS

`describe-addon-versions` Contoh berikut mencantumkan semua addon yang tersedia untuk versi Kubernetes tertentu yang digunakan untuk EKS.

```

aws eks describe-addon-versions \
  --kubernetes-version=1.26 \
  --query 'sort_by(addons &owner)[].{publisher: publisher, owner: owner,
addonName: addonName, type: type}' \
  --output table

```

Output:

-----

DescribeAddonVersions		
addonName	owner	publisher
type		
vpc-cni	aws	eks
networking		
snapshot-controller	aws	eks
storage		
kube-proxy	aws	eks
networking		
eks-pod-identity-agent	aws	eks
security		
coredns	aws	eks
networking		
aws-mountpoint-s3-csi-driver	aws	s3
storage		
aws-guardduty-agent	aws	eks
security		
aws-efs-csi-driver	aws	eks
storage		
aws-ebs-csi-driver	aws	eks
storage		
amazon-cloudwatch-observability	aws	eks
observability		
adot	aws	eks
observability		
upwind-security_upwind-operator	aws-marketplace	Upwind Security
security		
tetrade-io_istio-distro	aws-marketplace	tetrade-io
policy-management		
stormforge_optimize-live	aws-marketplace	StormForge
cost-management		
splunk_splunk-otel-collector-chart	aws-marketplace	Splunk
monitoring		
solo-io_istio-distro	aws-marketplace	Solo.io
service-mesh		
rafay-systems_rafay-operator	aws-marketplace	rafay-systems
kubernetes-management		
new-relic_kubernetes-operator	aws-marketplace	New Relic
observability		

```

| netapp_trident-operator | aws-marketplace | NetApp Inc.
|   | storage |
| leaksignal_leakagent | aws-marketplace | leaksignal
|   | monitoring |
| kubecost_kubecost | aws-marketplace | kubecost
|   | cost-management |
| kong_konnect-ri | aws-marketplace | kong
|   | ingress-service-type |
| haproxy-technologies_kubernetes-ingress-ee | aws-marketplace | HAProxy
Technologies | ingress-controller |
| groundcover_agent | aws-marketplace | groundcover
|   | monitoring |
| grafana-labs_kubernetes-monitoring | aws-marketplace | Grafana Labs
|   | monitoring |
| dynatrace_dynatrace-operator | aws-marketplace | dynatrace
|   | monitoring |
| datadog_operator | aws-marketplace | Datadog
|   | monitoring |
| cribl_cribledge | aws-marketplace | Cribl
|   | observability |
| calyptia_fluent-bit | aws-marketplace | Calyptia Inc
|   | observability |
| accuknox_kubearmor | aws-marketplace | AccuKnox
|   | security |
+-----+-----+
+-----+-----+

```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 3: Buat daftar semua versi addons vpc-cni yang tersedia untuk versi Kubernetes tertentu yang didukung untuk EKS

`describe-addon-versions` Contoh berikut mencantumkan semua versi addons vpc-cni yang tersedia untuk versi Kubernetes tertentu yang digunakan untuk EKS.

```

aws eks describe-addon-versions \
  --kubernetes-version=1.26 \
  --addon-name=vpc-cni \
  --query='addons[].addonVersions[].addonVersion'

```

Output:

```
[  
  "v1.18.0-eksbuild.1",  
  "v1.17.1-eksbuild.1",  
  "v1.16.4-eksbuild.2",  
  "v1.16.3-eksbuild.2",  
  "v1.16.2-eksbuild.1",  
  "v1.16.0-eksbuild.1",  
  "v1.15.5-eksbuild.1",  
  "v1.15.4-eksbuild.1",  
  "v1.15.3-eksbuild.1",  
  "v1.15.1-eksbuild.1",  
  "v1.15.0-eksbuild.2",  
  "v1.14.1-eksbuild.1",  
  "v1.14.0-eksbuild.3",  
  "v1.13.4-eksbuild.1",  
  "v1.13.3-eksbuild.1",  
  "v1.13.2-eksbuild.1",  
  "v1.13.0-eksbuild.1",  
  "v1.12.6-eksbuild.2",  
  "v1.12.6-eksbuild.1",  
  "v1.12.5-eksbuild.2",  
  "v1.12.0-eksbuild.2"  
]
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Membuat add-on](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [DescribeAddonVersions](#) di Referensi AWS CLI Perintah.

## describe-addon

Contoh kode berikut menunjukkan cara menggunakan `describe-addon`.

### AWS CLI

Jelaskan secara aktif menjalankan add-on EKS di cluster Amazon EKS Anda

`describe-addon` Contoh berikut secara aktif menjalankan add-on EKS di cluster Amazon EKS Anda.

```
aws eks describe-addon \  
  --cluster-name my-eks-cluster \  
  --region us-east-1
```

```
--addon-name vpc-cni
```

### Output:

```
{
  "addon": {
    "addonName": "vpc-cni",
    "clusterName": "my-eks-cluster",
    "status": "ACTIVE",
    "addonVersion": "v1.16.4-eksbuild.2",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/vpc-cni/0ec71efc-98dd-3203-60b0-4b939b2a5e5f",
    "createdAt": "2024-03-14T13:18:45.417000-04:00",
    "modifiedAt": "2024-03-14T13:18:49.557000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-Yfakrq0C1UTm",
    "tags": {
      "eks-addon-key-3": "value-3",
      "eks-addon-key-4": "value-4"
    },
    "configurationValues": "resources:\n    limits:\n        cpu: '100m'\nenv:\n    AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'"
  }
}
```

- Untuk detail API, lihat [DescribeAddon](#) di Referensi AWS CLI Perintah.

## describe-cluster

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster`.

### AWS CLI

Jelaskan secara aktif menjalankan addon EKS di cluster Amazon EKS Anda

`describe-cluster` Contoh berikut secara aktif menjalankan addon EKS di cluster Amazon EKS Anda.

```
aws eks describe-cluster \
```

```
--cluster-name my-eks-cluster
```

**Output:**

```
{
  "cluster": {
    "name": "my-eks-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster",
    "createdAt": "2024-03-14T11:31:44.348000-04:00",
    "version": "1.26",
    "endpoint": "https://JSA79429HJDASKJDJ8223829MNDNASW.y14.us-
east-2.eks.amazonaws.com",
    "roleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-cluster-
ServiceRole-zMF6CBakwwbW",
    "resourcesVpcConfig": {
      "subnetIds": [
        "subnet-0fb75d2d8401716e7",
        "subnet-02184492f67a3d0f9",
        "subnet-04098063527aab776",
        "subnet-0e2907431c9988b72",
        "subnet-04ad87f71c6e5ab4d",
        "subnet-09d912bb63ef21b9a"
      ],
      "securityGroupIds": [
        "sg-0c1327f6270afbb36"
      ],
      "clusterSecurityGroupId": "sg-01c84d09d70f39a7f",
      "vpcId": "vpc-0012b8e1cc0abb17d",
      "endpointPublicAccess": true,
      "endpointPrivateAccess": true,
      "publicAccessCidrs": [
        "22.19.18.2/32"
      ]
    },
    "kubernetesNetworkConfig": {
      "serviceIpv4Cidr": "10.100.0.0/16",
      "ipFamily": "ipv4"
    },
    "logging": {
      "clusterLogging": [
        {
          "types": [
            "api",
```

```
        "audit",
        "authenticator",
        "controllerManager",
        "scheduler"
    ],
    "enabled": true
  }
]
},
"identity": {
  "oidc": {
    "issuer": "https://oidc.eks.us-east-2.amazonaws.com/id/
JSA79429HJDASKJDJ8223829MNDNASW"
  }
},
"status": "ACTIVE",
"certificateAuthority": {
  "data": "CA_DATA_STRING..."
},
"platformVersion": "eks.14",
"tags": {
  "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
  "alpha.eksctl.io/cluster-name": "my-eks-cluster",
  "karpenter.sh/discovery": "my-eks-cluster",
  "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
  "auto-delete": "no",
  "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
  "EKS-Cluster-Name": "my-eks-cluster",
  "alpha.eksctl.io/cluster-oidc-enabled": "true",
  "aws:cloudformation:logical-id": "ControlPlane",
  "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
  "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
},
"health": {
  "issues": []
},
"accessConfig": {
  "authenticationMode": "API_AND_CONFIG_MAP"
}
}
```



```
}
```

- Untuk detail API, lihat [DescribeCluster](#) di Referensi AWS CLI Perintah.

## describe-fargate-profile

Contoh kode berikut menunjukkan cara menggunakan `describe-fargate-profile`.

### AWS CLI

Jelaskan profil Fargate

`describe-fargate-profile` Contoh berikut menjelaskan profil Fargate.

```
aws eks describe-fargate-profile \
  --cluster-name my-eks-cluster \
  --fargate-profile-name my-fargate-profile
```

Output:

```
{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/96c766ce-43d2-f9c9-954c-647334391198",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-04-11T10:42:52.486000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-farga-FargatePodExecutionRole-1htfAaJdJUE0",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "prod*",
        "labels": {
          "labelname*?": "*value1"
        }
      },
      {
        "namespace": "*dev*",

```

```

        "labels": {
            "labelname*?": "*value*"
        }
    ],
    "status": "ACTIVE",
    "tags": {
        "eks-fargate-profile-key-2": "value-2",
        "eks-fargate-profile-key-1": "value-1"
    }
}
}

```

- Untuk detail API, lihat [DescribeFargateProfile](#) di Referensi AWS CLI Perintah.

## describe-identity-provider-config

Contoh kode berikut menunjukkan cara menggunakan `describe-identity-provider-config`.

### AWS CLI

Jelaskan konfigurasi penyedia identitas yang terkait dengan Amazon EKS Cluster

`describe-identity-provider-config` Contoh berikut menjelaskan konfigurasi penyedia identitas yang terkait dengan Cluster Amazon EKS Anda.

```

aws eks describe-identity-provider-config \
  --cluster-name my-eks-cluster \
  --identity-provider-config type=oidc,name=my-identity-provider

```

### Output:

```

{
  "identityProviderConfig": {
    "oidc": {
      "identityProviderConfigName": "my-identity-provider",
      "identityProviderConfigArn": "arn:aws:eks:us-
east-2:111122223333:identityproviderconfig/my-eks-cluster/oidc/my-identity-
provider/8ac76722-78e4-cec1-ed76-d49eea058622",
      "clusterName": "my-eks-cluster",
      "issuerUrl": "https://oidc.eks.us-east-2.amazonaws.com/
id/38D6A4619A0A69E342B113ED7F1A7652",

```

```

    "clientId": "kubernetes",
    "usernameClaim": "email",
    "usernamePrefix": "my-username-prefix",
    "groupsClaim": "my-claim",
    "groupsPrefix": "my-groups-prefix",
    "requiredClaims": {
      "Claim1": "value1",
      "Claim2": "value2"
    },
    "tags": {
      "env": "dev"
    },
    "status": "ACTIVE"
  }
}
}

```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna untuk kluster Anda dari penyedia identitas OpenID Connect](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [DescribeIdentityProviderConfig](#) di Referensi AWS CLI Perintah.

## describe-nodegroup

Contoh kode berikut menunjukkan cara menggunakan `describe-nodegroup`.

### AWS CLI

Jelaskan grup node terkelola untuk kluster Amazon EKS

`describe-nodegroup` Contoh berikut menjelaskan grup node terkelola untuk kluster Amazon EKS.

```

aws eks describe-nodegroup \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup

```

Output:

```

{
  "nodegroup": {
    "nodegroupName": "my-eks-nodegroup",

```

```
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/a8c75f2f-df78-a72f-4063-4b69af3de5b1",
    "clusterName": "my-eks-cluster",
    "version": "1.26",
    "releaseVersion": "1.26.12-20240329",
    "createdAt": "2024-04-08T11:42:10.555000-04:00",
    "modifiedAt": "2024-04-08T11:44:12.402000-04:00",
    "status": "ACTIVE",
    "capacityType": "ON_DEMAND",
    "scalingConfig": {
      "minSize": 1,
      "maxSize": 3,
      "desiredSize": 1
    },
    "instanceTypes": [
      "t3.medium"
    ],
    "subnets": [
      "subnet-0e2907431c9988b72",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-09d912bb63ef21b9a"
    ],
    "amiType": "AL2_x86_64",
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",
    "labels": {},
    "resources": {
      "autoScalingGroups": [
        {
          "name": "eks-my-eks-nodegroup-a8c75f2f-df78-
a72f-4063-4b69af3de5b1"
        }
      ]
    },
    "diskSize": 20,
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 1
    },
    "tags": {}
  }
}
```

- Untuk detail API, lihat [DescribeNodegroup](#) di Referensi AWS CLI Perintah.

## describe-update

Contoh kode berikut menunjukkan cara menggunakan `describe-update`.

### AWS CLI

Contoh 1: Untuk menjelaskan pembaruan untuk kluster

`describe-update` Contoh berikut menjelaskan pembaruan untuk cluster bernama.

```
aws eks describe-update \  
  --name my-eks-cluster \  
  --update-id 10bddb13-a71b-425a-b0a6-71cd03e59161
```

Output:

```
{  
  "update": {  
    "id": "10bddb13-a71b-425a-b0a6-71cd03e59161",  
    "status": "Successful",  
    "type": "EndpointAccessUpdate",  
    "params": [  
      {  
        "type": "EndpointPublicAccess",  
        "value": "false"  
      },  
      {  
        "type": "EndpointPrivateAccess",  
        "value": "true"  
      }  
    ],  
    "createdAt": "2024-03-14T10:01:26.297000-04:00",  
    "errors": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui versi Kubernetes cluster Amazon EKS di Panduan Pengguna Amazon EKS](#).

Contoh 2: Untuk menjelaskan pembaruan untuk kluster

`describe-update` Contoh berikut menjelaskan pembaruan untuk cluster bernama.

```
aws eks describe-update \  
  --name my-eks-cluster \  
  --update-id e4994991-4c0f-475a-a040-427e6da52966
```

Output:

```
{  
  "update": {  
    "id": "e4994991-4c0f-475a-a040-427e6da52966",  
    "status": "Successful",  
    "type": "AssociateEncryptionConfig",  
    "params": [  
      {  
        "type": "EncryptionConfig",  
        "value": "[{\"resources\": [\"secrets\"], \"provider\": {\"keyArn\":  
\\\"arn:aws:kms:region-code:account:key/key\\\"}}]"  
      }  
    ],  
    "createdAt": "2024-03-14T11:01:26.297000-04:00",  
    "errors": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui versi Kubernetes cluster Amazon EKS di Panduan Pengguna Amazon EKS](#).

Contoh 3: Untuk menjelaskan pembaruan untuk klaster

`describe-update` Contoh berikut menjelaskan pembaruan untuk cluster bernama.

```
aws eks describe-update \  
  --name my-eks-cluster \  
  --update-id b5f0ba18-9a87-4450-b5a0-825e6e84496f
```

Output:

```
{  
  "update": {  
    "id": "b5f0ba18-9a87-4450-b5a0-825e6e84496f",
```

```
    "status": "Successful",
    "type": "VersionUpdate",
    "params": [
      {
        "type": "Version",
        "value": "1.29"
      },
      {
        "type": "PlatformVersion",
        "value": "eks.1"
      }
    ],
    "createdAt": "2024-03-14T12:05:26.297000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui versi Kubernetes cluster Amazon EKS di Panduan Pengguna Amazon EKS](#).

- Untuk detail API, lihat [DescribeUpdate](#) di Referensi AWS CLI Perintah.

## disassociate-identity-provider-config

Contoh kode berikut menunjukkan cara menggunakan `disassociate-identity-provider-config`.

### AWS CLI

Pisahkan penyedia identitas ke Amazon EKS Cluster Anda

`disassociate-identity-provider-config` Contoh berikut memisahkan penyedia identitas ke Amazon EKS Cluster Anda.

```
aws eks disassociate-identity-provider-config \
  --cluster-name my-eks-cluster \
  --identity-provider-config 'type=oidc,name=my-identity-provider'
```

Output:

```
{
```

```
"update": {
  "id": "5f78d14e-c57b-4857-a3e4-cf664ae20949",
  "status": "InProgress",
  "type": "DisassociateIdentityProviderConfig",
  "params": [
    {
      "type": "IdentityProviderConfig",
      "value": "[]"
    }
  ],
  "createdAt": "2024-04-11T13:53:43.314000-04:00",
  "errors": []
}
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna untuk kluster Anda dari penyedia identitas OpenID Connect - Lepaskan penyedia identitas OIDC dari kluster Anda di Panduan Pengguna Amazon EKS](#).

- Untuk detail API, lihat [DisassociateIdentityProviderConfig](#) di Referensi AWS CLI Perintah.

## get-token

Contoh kode berikut menunjukkan cara menggunakan `get-token`.

### AWS CLI

Contoh 1: Dapatkan token otentikasi untuk Cluster Amazon EKS bernama `my-eks-cluster`

`get-token` Contoh berikut mendapatkan token otentikasi untuk Amazon EKS Cluster bernama `my-eks-cluster`.

```
aws eks get-token \
  --cluster-name my-eks-cluster
```

Output:

```
{
  "kind": "ExecCredential",
  "apiVersion": "client.authentication.k8s.io/v1beta1",
  "spec": {},
  "status": {
```



```

    "expirationTimestamp": "2024-04-11T20:59:56Z",
    "token": "k8s-aws-v1.EXAMPLE_TOKEN_DATA_STRING..."
  }
}

```

Contoh 2: Mendapatkan token otentikasi untuk Cluster Amazon EKS bernama `my-eks-cluster` dengan mengasumsikan RoLearn ini untuk kredensi saat menandatangani token

`get-token` Contoh berikut mendapatkan token otentikasi untuk Cluster Amazon EKS bernama `my-eks-cluster` dengan mengasumsikan RoLearn ini untuk kredensil saat menandatangani token.

```

aws eks get-token \
  --cluster-name my-eks-cluster \
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-
cluster-ServiceRole-j1k7AfTIQtnM

```

Output:

```

{
  "kind": "ExecCredential",
  "apiVersion": "client.authentication.k8s.io/v1beta1",
  "spec": {},
  "status": {
    "expirationTimestamp": "2024-04-11T21:05:26Z",
    "token": "k8s-aws-v1.EXAMPLE_TOKEN_DATA_STRING..."
  }
}

```

- Untuk detail API, lihat [GetToken](#) di Referensi AWS CLI Perintah.

## list-addons

Contoh kode berikut menunjukkan cara menggunakan `list-addons`.

### AWS CLI

Buat daftar semua add-on yang diinstal di cluster Amazon EKS Anda bernama `` my-eks-cluster

`list-addons` Contoh berikut mencantumkan semua add-on yang diinstal di kluster Amazon EKS Anda bernama `my-eks-cluster`.

```
aws eks list-addons \  
  --cluster-name my-eks-cluster
```

Output:

```
{  
  "addons": [  
    "kube-proxy",  
    "vpc-cni"  
  ]  
}
```

- Untuk detail API, lihat [ListAddons](#) di Referensi AWS CLI Perintah.

## list-clusters

Contoh kode berikut menunjukkan cara menggunakan `list-clusters`.

### AWS CLI

Untuk mencantumkan semua add-on yang diinstal di cluster Amazon EKS Anda bernama `my-eks-cluster`

`list-clusters` Contoh berikut mencantumkan semua add-on yang diinstal di kluster Amazon EKS Anda bernama `my-eks-cluster`.

```
aws eks list-clusters
```

Output:

```
{  
  "clusters": [  
    "prod",  
    "qa",  
    "stage",  
    "my-eks-cluster"  
  ]  
}
```

- Untuk detail API, lihat [ListClusters](#) di Referensi AWS CLI Perintah.

## list-fargate-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-fargate-profiles`.

### AWS CLI

Untuk membuat daftar semua profil fargate di cluster Amazon EKS Anda bernama `my-eks-cluster`

`list-fargate-profiles` Contoh berikut mencantumkan semua profil fargate di klaster Amazon EKS Anda bernama `my-eks-cluster`.

```
aws eks list-fargate-profiles \
  --cluster-name my-eks-cluster
```

### Output:

```
{
  "fargateProfileNames": [
    "my-fargate-profile"
  ]
}
```

- Untuk detail API, lihat [ListFargateProfiles](#) di Referensi AWS CLI Perintah.

## list-identity-provider-configs

Contoh kode berikut menunjukkan cara menggunakan `list-identity-provider-configs`.

### AWS CLI

Daftar penyedia identitas yang terkait dengan Amazon EKS Cluster

`list-identity-provider-configs` Contoh berikut mencantumkan penyedia identitas yang terkait dengan Amazon EKS Cluster.

```
aws eks list-identity-provider-configs \
  --cluster-name my-eks-cluster
```

### Output:

```
{
  "identityProviderConfigs": [
    {
      "type": "oidc",
      "name": "my-identity-provider"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna untuk klaster Anda dari penyedia identitas OpenID Connect](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [ListIdentityProviderConfigs](#) di Referensi AWS CLI Perintah.

## list-nodegroups

Contoh kode berikut menunjukkan cara menggunakan `list-nodegroups`.

### AWS CLI

Buat daftar semua grup simpul di cluster Amazon EKS

`list-nodegroups` Contoh berikut mencantumkan semua grup node dalam cluster Amazon EKS.

```
aws eks list-nodegroups \
  --cluster-name my-eks-cluster
```

Output:

```
{
  "nodegroups": [
    "my-eks-managed-node-group",
    "my-eks-nodegroup"
  ]
}
```

- Untuk detail API, lihat [ListNodegroups](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Contoh 1: Untuk daftar semua tag untuk Amazon EKS Cluster ARN

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk Amazon EKS Cluster ARN.

```
aws eks list-tags-for-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster
```

Output:

```
{
  "tags": {
    "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
    "alpha.eksctl.io/cluster-name": "my-eks-cluster",
    "karpenter.sh/discovery": "my-eks-cluster",
    "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
    "auto-delete": "no",
    "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
    "EKS-Cluster-Name": "my-eks-cluster",
    "alpha.eksctl.io/cluster-oidc-enabled": "true",
    "aws:cloudformation:logical-id": "ControlPlane",
    "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
    "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
  }
}
```

Contoh 2: Untuk mencantumkan semua tag untuk grup Amazon EKS Node ARN

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk grup Amazon EKS Node ARN.

```
aws eks list-tags-for-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-
eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c
```

Output:

```
{
  "tags": {
    "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-nodegroup-my-eks-
managed-node-group",
    "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-nodegroup-my-eks-managed-node-group/
eaa20310-e219-11ee-b851-0ab9ad8228ff",
    "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
    "EKS-Cluster-Name": "my-eks-cluster",
    "alpha.eksctl.io/nodegroup-type": "managed",
    "NodeGroup Name 1": "my-eks-managed-node-group",
    "k8s.io/cluster-autoscaler/enabled": "true",
    "nodegroup-role": "worker",
    "alpha.eksctl.io/cluster-name": "my-eks-cluster",
    "alpha.eksctl.io/nodegroup-name": "my-eks-managed-node-group",
    "karpenter.sh/discovery": "my-eks-cluster",
    "NodeGroup Name 2": "AmazonLinux-Linux-Managed-NG-v1-26-v1",
    "auto-delete": "no",
    "k8s.io/cluster-autoscaler/my-eks-cluster": "owned",
    "aws:cloudformation:logical-id": "ManagedNodeGroup",
    "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z"
  }
}
```

Contoh 3: Untuk mencantumkan semua tag pada profil Amazon EKS Fargate ARNE

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk profil Amazon EKS Fargate ARN.

```
aws eks list-tags-for-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/
my-fargate-profile/d6c76780-e541-0725-c816-36754cab734b
```

Output:

```
{
  "tags": {
    "eks-fargate-profile-key-2": "value-2",
    "eks-fargate-profile-key-1": "value-1"
  }
}
```

#### Contoh 4: Untuk mencantumkan semua tag untuk ARN Add-on Amazon EKS

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk ARN Add-on Amazon EKS.

```
aws eks list-tags-for-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/vpc-  
cni/0ec71efc-98dd-3203-60b0-4b939b2a5e5f
```

Output:

```
{  
  "tags": {  
    "eks-addon-key-2": "value-2",  
    "eks-addon-key-1": "value-1"  
  }  
}
```

#### Contoh 5: Untuk mencantumkan semua tag untuk penyedia identitas Amazon EKS OIDC ARN

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk ARN penyedia identitas Amazon EKS OIDC.

```
aws eks list-tags-for-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:identityproviderconfig/my-eks-  
cluster/oidc/my-identity-provider/8ac76722-78e4-cec1-ed76-d49eea058622
```

Output:

```
{  
  "tags": {  
    "my-identity-provider": "test"  
  }  
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **list-update**

Contoh kode berikut menunjukkan cara menggunakan `list-update`.

## AWS CLI

Contoh 1: Untuk mencantumkan pembaruan yang terkait dengan nama Amazon EKS Cluster

`list-updates` Contoh berikut mencantumkan semua ID pembaruan untuk nama Amazon EKS Cluster.

```
aws eks list-updates \  
  --name my-eks-cluster
```

Output:

```
{  
  "updateIds": [  
    "5f78d14e-c57b-4857-a3e4-cf664ae20949",  
    "760e5a3f-adad-48c7-88d3-7ac283c09c26",  
    "cd4ec863-bc55-47d5-a377-3971502f529b",  
    "f12657ce-e869-4f17-b158-a82ab8b7d937"  
  ]  
}
```

Contoh 2: Untuk mencantumkan semua ID pembaruan untuk grup Amazon EKS Node

`list-updates` Contoh berikut mencantumkan semua ID pembaruan untuk grup Amazon EKS Node.

```
aws eks list-updates \  
  --name my-eks-cluster \  
  --nodegroup-name my-eks-managed-node-group
```

Output:

```
{  
  "updateIds": [  
    "8c6c1bef-61fe-42ac-a242-89412387b8e7"  
  ]  
}
```

Contoh 3: Untuk mencantumkan semua ID pembaruan di Amazon EKS Add-one

`list-updates` Contoh berikut mencantumkan semua ID pembaruan untuk Add-on Amazon EKS.



```
aws eks list-updates \  
  --name my-eks-cluster \  
  --addon-name vpc-cni
```

Output:

```
{  
  "updateIds": [  
    "9cdba8d4-79fb-3c83-afe8-00b508d33268"  
  ]  
}
```

- Untuk detail API, lihat [ListUpdates](#) di Referensi AWS CLI Perintah.

## list-updates

Contoh kode berikut menunjukkan cara menggunakan `list-updates`.

### AWS CLI

Untuk membuat daftar pembaruan untuk kluster

Perintah contoh ini mencantumkan pembaruan saat ini untuk kluster bernama `example` di wilayah default Anda.

Perintah:

```
aws eks list-updates --name example
```

Output:

```
{  
  "updateIds": [  
    "10bddb13-a71b-425a-b0a6-71cd03e59161"  
  ]  
}
```

- Untuk detail API, lihat [ListUpdates](#) di Referensi AWS CLI Perintah.

## register-cluster

Contoh kode berikut menunjukkan cara menggunakan `register-cluster`.

### AWS CLI

Contoh 1: Daftarkan kluster Eks\_ANYWHERE Kubernetes eksternal ke Amazon EKS

`register-cluster` Contoh berikut mendaftarkan kluster Eks\_ANYWHERE Kubernetes eksternal ke Amazon EKS.

```
aws eks register-cluster \
  --name my-eks-anywhere-cluster \
  --connector-config 'roleArn=arn:aws:iam::111122223333:role/
AmazonEKSCoordinatorAgentRole,provider=EKS_ANYWHERE'
```

Output:

```
{
  "cluster": {
    "name": "my-eks-anywhere-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-anywhere-cluster",
    "createdAt": "2024-04-12T12:38:37.561000-04:00",
    "status": "PENDING",
    "tags": {},
    "connectorConfig": {
      "activationId": "xxxxxxxxACTIVATION_IDxxxxxxxx",
      "activationCode": "xxxxxxxxACTIVATION_CODExxxxxxxx",
      "activationExpiry": "2024-04-15T12:38:37.082000-04:00",
      "provider": "EKS_ANYWHERE",
      "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSCoordinatorAgentRole"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan kluster eksternal](#) di Panduan Pengguna Amazon EKS.

Contoh 2: Daftarkan kluster Kubernetes eksternal apa pun ke Amazon EKS

`register-cluster` Contoh berikut mendaftarkan kluster Eks\_ANYWHERE Kubernetes eksternal ke Amazon EKS.

```
aws eks register-cluster \  
  --name my-eks-anywhere-cluster \  
  --connector-config 'roleArn=arn:aws:iam::111122223333:role/  
AmazonEKSCoordinatorAgentRole,provider=OTHER'
```

Output:

```
{  
  "cluster": {  
    "name": "my-onprem-k8s-cluster",  
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-onprem-k8s-cluster",  
    "createdAt": "2024-04-12T12:42:10.861000-04:00",  
    "status": "PENDING",  
    "tags": {},  
    "connectorConfig": {  
      "activationId": "xxxxxxxxACTIVATION_IDxxxxxxxx",  
      "activationCode": "xxxxxxxxACTIVATION_CODExxxxxxxx",  
      "activationExpiry": "2024-04-15T12:42:10.339000-04:00",  
      "provider": "OTHER",  
      "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSCoordinatorAgentRole"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan kluster eksternal](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [RegisterCluster](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Contoh 1: Untuk menambahkan tag yang ditentukan ke Cluster Amazon EKS

`tag-resource` Contoh berikut menambahkan tag yang ditentukan ke Amazon EKS Cluster.

```
aws eks tag-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster \  
  --tag 'my-eks-cluster-test-1=test-value-1,my-eks-cluster-dev-1=dev-value-2'
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menambahkan tag yang ditentukan ke grup Amazon EKS Node

`tag-resource` Contoh berikut menambahkan tag yang ditentukan ke grup Amazon EKS Node.

```
aws eks tag-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-  
eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c \  
  --tag 'my-eks-nodegroup-test-1=test-value-1,my-eks-nodegroup-dev-1=dev-value-2'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Contoh 1: Untuk menghapus tag yang ditentukan dari Cluster Amazon EKS

`untag-resource` Contoh berikut menghapus tag yang ditentukan dari Amazon EKS Cluster.

```
aws eks untag-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster \  
  --tag-keys "my-eks-cluster-test-1" "my-eks-cluster-dev-1"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menghapus tag yang ditentukan dari grup Amazon EKS Node

`untag-resource` Contoh berikut menghapus tag yang ditentukan dari grup Amazon EKS Node.

```
aws eks untag-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-  
eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c \  
  --tag-keys "my-eks-nodegroup-test-1" "my-eks-nodegroup-dev-1"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-addon

Contoh kode berikut menunjukkan cara menggunakan `update-addon`.

### AWS CLI

Contoh 1. Untuk memperbarui add-on Amazon EKS dengan peran akun layanan ARN

`update-addon` Contoh perintah berikut memperbarui add-on Amazon EKS dengan peran akun layanan ARN.

```
aws eks update-addon \
  --cluster-name my-eks-cluster \
  --addon-name vpc-cni \
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm
```

Output:

```
{
  "update": {
    "id": "c00d2de2-c2e4-3d30-929e-46b8edec2ce4",
    "status": "InProgress",
    "type": "AddonUpdate",
    "params": [
      {
        "type": "ServiceAccountRoleArn",
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      }
    ],
    "updatedAt": "2024-04-12T16:04:55.614000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Memperbarui add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 2. Untuk memperbarui add-on Amazon EKS dengan versi add-on tertentu

`update-addon` Contoh perintah berikut memperbarui add-on Amazon EKS dengan versi add-on tertentu.

```
aws eks update-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni \  
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm \  
  --addon-version v1.16.4-eksbuild.2
```

### Output:

```
{  
  "update": {  
    "id": "f58dc0b0-2b18-34bd-bc6a-e4abc0011f36",  
    "status": "InProgress",  
    "type": "AddonUpdate",  
    "params": [  
      {  
        "type": "AddonVersion",  
        "value": "v1.16.4-eksbuild.2"  
      },  
      {  
        "type": "ServiceAccountRoleArn",  
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm"  
      }  
    ],  
    "createdAt": "2024-04-12T16:07:16.550000-04:00",  
    "errors": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Memperbarui add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 3. Untuk memperbarui add-on Amazon EKS dengan nilai konfigurasi khusus dan menyelesaikan detail konflik

update-addon Contoh perintah berikut memperbarui add-on Amazon EKS dengan nilai konfigurasi khusus dan menyelesaikan detail konflik.

```
aws eks update-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni \  
  --addon-version v1.16.4-eksbuild.2
```

```

--service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm \
--addon-version v1.16.4-eksbuild.2 \
--configuration-values '{"resources": {"limits":{"cpu":"100m"}, "requests":
{"cpu":"50m"}}}' \
--resolve-conflicts PRESERVE

```

### Output:

```

{
  "update": {
    "id": "cd9f2173-a8d8-3004-a90f-032f14326520",
    "status": "InProgress",
    "type": "AddonUpdate",
    "params": [
      {
        "type": "AddonVersion",
        "value": "v1.16.4-eksbuild.2"
      },
      {
        "type": "ServiceAccountRoleArn",
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      },
      {
        "type": "ResolveConflicts",
        "value": "PRESERVE"
      },
      {
        "type": "ConfigurationValues",
        "value": "{\"resources\": {\"limits\": {\"cpu\": \"100m\"}, \"requests
\": {\"cpu\": \"50m\"}}}"
      }
    ],
    "createdAt": "2024-04-12T16:16:27.363000-04:00",
    "errors": []
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Memperbarui add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 4. Untuk memperbarui add-on Amazon EKS dengan file nilai konfigurasi JSON kustom

update-addon Contoh perintah berikut memperbarui add-on Amazon EKS dengan nilai konfigurasi JSON kustom dan menyelesaikan detail konflik.

```
aws eks update-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni \  
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm \  
  --addon-version v1.17.1-eksbuild.1 \  
  --configuration-values 'file://configuration-values.json' \  
  --resolve-conflicts PRESERVE
```

Isi dari `configuration-values.json`:

```
{  
  "resources": {  
    "limits": {  
      "cpu": "100m"  
    },  
    "requests": {  
      "cpu": "50m"  
    }  
  },  
  "env": {  
    "AWS_VPC_K8S_CNI_LOGLEVEL": "ERROR"  
  }  
}
```

Output:

```
{  
  "update": {  
    "id": "6881a437-174f-346b-9a63-6e91763507cc",  
    "status": "InProgress",  
    "type": "AddonUpdate",  
    "params": [  
      {  
        "type": "AddonVersion",  
        "value": "v1.17.1-eksbuild.1"  
      },  
      {  
        "type": "ServiceAccountRoleArn",
```



```

        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      },
      {
        "type": "ResolveConflicts",
        "value": "PRESERVE"
      },
      {
        "type": "ConfigurationValues",
        "value": "{\n  \"resources\": {\n    \"limits\": {\n
      \"cpu\": \"100m\"\n    },\n    \"requests\": {\n      \"cpu\": \"50m
      }\n    },\n  \"env\": {\n    \"AWS_VPC_K8S_CNI_LOGLEVEL\": \"ERROR
      }\n  }\n}"
      }
    ],
    "createdAt": "2024-04-12T16:22:55.519000-04:00",
    "errors": []
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Memperbarui add-on](#) di Panduan Pengguna Amazon EKS.

Contoh 5. Untuk memperbarui add-on Amazon EKS dengan file nilai konfigurasi YAMAL kustom

update-addon Contoh perintah berikut memperbarui add-on Amazon EKS dengan nilai konfigurasi YAMAL kustom dan menyelesaikan detail konflik.

```

aws eks update-addon \
  --cluster-name my-eks-cluster \
  --addon-name vpc-cni \
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm \
  --addon-version v1.18.0-eksbuild.1 \
  --configuration-values 'file://configuration-values.yaml' \
  --resolve-conflicts PRESERVE

```

Isi dari configuration-values.yaml:

```

resources:
  limits:
    cpu: '100m'

```

```

requests:
  cpu: '50m'
env:
  AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'

```

## Output:

```

{
  "update": {
    "id": "a067a4c9-69d0-3769-ace9-d235c5b16701",
    "status": "InProgress",
    "type": "AddonUpdate",
    "params": [
      {
        "type": "AddonVersion",
        "value": "v1.18.0-eksbuild.1"
      },
      {
        "type": "ServiceAccountRoleArn",
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      },
      {
        "type": "ResolveConflicts",
        "value": "PRESERVE"
      },
      {
        "type": "ConfigurationValues",
        "value": "resources:\n      limits:\n          cpu: '100m'\n
requests:\n      cpu: '50m'\nenv:\n      AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'"
      }
    ],
    "createdAt": "2024-04-12T16:25:07.212000-04:00",
    "errors": []
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola add-on Amazon EKS - Memperbarui add-on](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [UpdateAddon](#) di Referensi AWS CLI Perintah.

## update-cluster-config

Contoh kode berikut menunjukkan cara menggunakan `update-cluster-config`.

### AWS CLI

Untuk memperbarui akses titik akhir klaster

Perintah contoh ini memperbarui cluster untuk menonaktifkan akses publik titik akhir dan mengaktifkan akses titik akhir pribadi.

Perintah:

```
aws eks update-cluster-config --name example \  
--resources-vpc-config endpointPublicAccess=false,endpointPrivateAccess=true
```

Output:

```
{  
  "update": {  
    "id": "ec883c93-2e9e-407c-a22f-8f6fa6e67d4f",  
    "status": "InProgress",  
    "type": "EndpointAccessUpdate",  
    "params": [  
      {  
        "type": "EndpointPublicAccess",  
        "value": "false"  
      },  
      {  
        "type": "EndpointPrivateAccess",  
        "value": "true"  
      }  
    ],  
    "createdAt": 1565806986.506,  
    "errors": []  
  }  
}
```

Untuk mengaktifkan logging untuk klaster

Perintah contoh ini memungkinkan semua jenis logging bidang kontrol cluster untuk sebuah cluster bernama `example`.

**Perintah:**

```
aws eks update-cluster-config --name example \  
--logging '{"clusterLogging":[{"types":  
["api","audit","authenticator","controllerManager","scheduler"],"enabled":true}]}'
```

**Output:**

```
{  
  "update": {  
    "id": "7551c64b-1d27-4b1e-9f8e-c45f056eb6fd",  
    "status": "InProgress",  
    "type": "LoggingUpdate",  
    "params": [  
      {  
        "type": "ClusterLogging",  
        "value": "{\"clusterLogging\":{\"types\":[\"api\", \"audit\",  
\"authenticator\", \"controllerManager\", \"scheduler\"], \"enabled\": true}}"  
      }  
    ],  
    "createdAt": 1565807210.37,  
    "errors": []  
  }  
}
```

- Untuk detail API, lihat [UpdateClusterConfig](#) di Referensi AWS CLI Perintah.

**update-cluster-version**

Contoh kode berikut menunjukkan cara menggunakan `update-cluster-version`.

**AWS CLI**

Untuk memperbarui cluster Amazon EKS bernama `my-eks-cluster` ke versi Kubernetes yang ditentukan

`update-cluster-version` Contoh berikut memperbarui kluster Amazon EKS ke versi Kubernetes yang ditentukan.

```
aws eks update-cluster-version \  
--name my-eks-cluster \  

```

```
--kubernetes-version 1.27
```

### Output:

```
{
  "update": {
    "id": "e4091a28-ea14-48fd-a8c7-975aeb469e8a",
    "status": "InProgress",
    "type": "VersionUpdate",
    "params": [
      {
        "type": "Version",
        "value": "1.27"
      },
      {
        "type": "PlatformVersion",
        "value": "eks.16"
      }
    ],
    "createdAt": "2024-04-12T16:56:01.082000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui versi Kubernetes cluster Amazon EKS di Panduan Pengguna Amazon EKS](#).

- Untuk detail API, lihat [UpdateClusterVersion](#) di Referensi AWS CLI Perintah.

## update-kubeconfig

Contoh kode berikut menunjukkan cara menggunakan `update-kubeconfig`.

### AWS CLI

Contoh 1: Mengonfigurasi `kubectl` Anda dengan membuat atau memperbarui `kubeconfig` sehingga Anda dapat terhubung ke Cluster Amazon EKS bernama `my-eks-cluster`

`update-kubeconfig` Contoh berikut mengkonfigurasi `kubectl` Anda dengan membuat atau memperbarui `kubeconfig` sehingga Anda dapat terhubung ke Amazon EKS Cluster bernama `my-eks-cluster`

```
aws eks update-kubeconfig \  
  --name my-eks-cluster
```

Output:

```
Updated context arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster in /Users/  
xxx/.kube/config
```

Untuk informasi selengkapnya, lihat [Membuat atau memperbarui file kubeconfig untuk klaster Amazon EKS di Panduan Pengguna Amazon EKS](#).

Contoh 2: Mengonfigurasi kubectl Anda dengan membuat atau memperbarui kubeconfig (dengan opsi role-arn untuk mengambil peran untuk otentikasi klaster) sehingga Anda dapat terhubung ke Cluster Amazon EKS bernama `` my-eks-cluster

update-kubeconfig Contoh berikut mengonfigurasi kubectl Anda dengan membuat atau memperbarui kubeconfig (dengan opsi role-arn untuk mengambil peran untuk otentikasi klaster) sehingga Anda dapat terhubung ke Cluster Amazon EKS bernama. my-eks-cluster

```
aws eks update-kubeconfig \  
  --name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-  
cluster-ServiceRole-j1k7AfTIQtnM
```

Output:

```
Updated context arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster in /Users/  
xxx/.kube/config
```

Untuk informasi selengkapnya, lihat [Membuat atau memperbarui file kubeconfig untuk klaster Amazon EKS di Panduan Pengguna Amazon EKS](#).

Contoh 3: Mengonfigurasi kubectl Anda dengan membuat atau memperbarui kubeconfig (dengan opsi role-arn untuk mengambil peran untuk otentikasi klaster bersama dengan alias cluster khusus dan alias pengguna) sehingga Anda dapat terhubung ke Cluster Amazon EKS bernama `` my-eks-cluster

update-kubeconfig Contoh berikut mengonfigurasi kubectl Anda dengan membuat atau memperbarui kubeconfig (dengan opsi role-arn untuk mengambil peran untuk otentikasi klaster

bersama dengan alias klaster kustom dan alias pengguna) sehingga Anda dapat terhubung ke Cluster Amazon EKS bernama. my-eks-cluster

```
aws eks update-kubeconfig \  
  --name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-  
cluster-ServiceRole-j1k7AfTIQtnM \  
  --alias stage-eks-cluster \  
  --user-alias john
```

Output:

```
Updated context stage-eks-cluster in /Users/dubaria/.kube/config
```

Untuk informasi selengkapnya, lihat [Membuat atau memperbarui file kubeconfig untuk klaster Amazon EKS di Panduan Pengguna Amazon EKS](#).

Contoh 4: Cetak entri file kubeconfig untuk ditinjau dan mengkonfigurasi kubectl Anda sehingga Anda dapat terhubung ke Amazon EKS Cluster bernama `` my-eks-cluster

update-kubeconfigContoh berikut mengonfigurasi kubectl Anda dengan membuat atau memperbarui kubeconfig (dengan opsi role-arn untuk mengambil peran untuk otentikasi klaster bersama dengan alias klaster kustom dan alias pengguna) sehingga Anda dapat terhubung ke Cluster Amazon EKS bernama. my-eks-cluster

```
aws eks update-kubeconfig \  
  --name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-  
cluster-ServiceRole-j1k7AfTIQtnM \  
  --alias stage-eks-cluster \  
  --user-alias john \  
  --verbose
```

Output:

```
Updated context stage-eks-cluster in /Users/dubaria/.kube/config  
Entries:  
  
context:  
cluster: arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster  
user: john
```

```

name: stage-eks-cluster

name: john
user:
exec:
  apiVersion: client.authentication.k8s.io/v1beta1
  args:
  - --region
  - us-east-2
  - eks
  - get-token
  - --cluster-name
  - my-eks-cluster
  - --output
  - json
  - --role
  - arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-
ServiceRole-j1k7AfTIQtnM
  command: aws

cluster:
certificate-authority-data: xxx_CA_DATA_xxx
server: https://DALSJ343KE23J3RN45653DSKJTT647TYD.y14.us-east-2.eks.amazonaws.com
name: arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster

```

Untuk informasi selengkapnya, lihat [Membuat atau memperbarui file kubeconfig untuk kluster Amazon EKS di Panduan Pengguna Amazon EKS](#).

- Untuk detail API, lihat [UpdateKubeconfig](#) di Referensi AWS CLI Perintah.

## update-nodegroup-config

Contoh kode berikut menunjukkan cara menggunakan `update-nodegroup-config`.

### AWS CLI

Contoh 1: Perbarui grup node terkelola untuk menambahkan label baru dan menodai node pekerja EKS untuk kluster Amazon EKS

`update-nodegroup-config` Contoh berikut memperbarui grup node terkelola untuk menambahkan label baru dan menodai node pekerja EKS untuk kluster Amazon EKS.

```
aws eks update-nodegroup-config \
```



```
--cluster-name my-eks-cluster \
--nodegroup-name my-eks-nodegroup \
--labels 'addOrUpdateLabels={my-eks-nodegroup-label-1=value-1,my-eks-nodegroup-
label-2=value-2}' \
--taints 'addOrUpdateTaints=[{key=taint-key-1,value=taint-
value-1,effect=NO_EXECUTE}]'
```

### Output:

```
{
  "update": {
    "id": "e66d21d3-bd8b-3ad1-a5aa-b196dc08c7c1",
    "status": "InProgress",
    "type": "ConfigUpdate",
    "params": [
      {
        "type": "LabelsToAdd",
        "value": "{\"my-eks-nodegroup-label-2\":\"value-2\",\"my-eks-
nodegroup-label-1\":\"value-1\"}"
      },
      {
        "type": "TaintsToAdd",
        "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",
\"key\":\"taint-key-1\"}]"
      }
    ],
    "createdAt": "2024-04-08T12:05:19.161000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui grup node terkelola](#) di Panduan Pengguna Amazon EKS.

Contoh 2: Memperbarui grup node terkelola untuk menghapus label dan menodai node pekerja EKS untuk kluster Amazon EKS

update-nodegroup-config Contoh berikut memperbarui grup node terkelola untuk menghapus label dan taint untuk node pekerja EKS untuk kluster Amazon EKS.

```
aws eks update-nodegroup-config \
--cluster-name my-eks-cluster \
```

```
--nodegroup-name my-eks-nodegroup \
--labels 'removeLabels=my-eks-nodegroup-label-1, my-eks-nodegroup-label-2' \
--taints 'removeTaints=[{key=taint-key-1,value=taint-
value-1,effect=NO_EXECUTE}]'
```

### Output:

```
{
  "update": {
    "id": "67a08692-9e59-3ace-a916-13929f44cec3",
    "status": "InProgress",
    "type": "ConfigUpdate",
    "params": [
      {
        "type": "LabelsToRemove",
        "value": "[\"my-eks-nodegroup-label-1\", \"my-eks-nodegroup-
label-2\"]"
      },
      {
        "type": "TaintsToRemove",
        "value": "[{\"effect\": \"NO_EXECUTE\", \"value\": \"taint-value-1\",
\"key\": \"taint-key-1\"}]"
      }
    ],
    "createdAt": "2024-04-08T12:17:31.817000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui grup node terkelola](#) di Panduan Pengguna Amazon EKS.

Contoh 3: Perbarui grup node terkelola untuk menghapus dan menambahkan label dan taint untuk node pekerja EKS untuk kluster Amazon EKS

update-nodegroup-configContoh berikut memperbarui grup node terkelola untuk menghapus dan menambahkan label dan taint untuk node pekerja EKS untuk kluster Amazon EKS.

```
aws eks update-nodegroup-config \
--cluster-name my-eks-cluster \
--nodegroup-name my-eks-nodegroup \
```

```
--labels 'addOrUpdateLabels={my-eks-nodegroup-new-label-1=new-value-1,my-eks-
nodegroup-new-label-2=new-value-2},removeLabels=my-eks-nodegroup-label-1, my-eks-
nodegroup-label-2' \
--taints 'addOrUpdateTaints=[{key=taint-new-key-1,value=taint-new-
value-1,effect=PREFER_NO_SCHEDULE}],removeTaints=[{key=taint-key-1,value=taint-
value-1,effect=NO_EXECUTE}]'
```

### Output:

```
{
  "update": {
    "id": "4a9c8c45-6ac7-3115-be71-d6412a2339b7",
    "status": "InProgress",
    "type": "ConfigUpdate",
    "params": [
      {
        "type": "LabelsToAdd",
        "value": "{\"my-eks-nodegroup-new-label-1\":\"new-value-1\",\"my-
eks-nodegroup-new-label-2\":\"new-value-2\"}"
      },
      {
        "type": "LabelsToRemove",
        "value": "[\"my-eks-nodegroup-label-1\",\"my-eks-nodegroup-
label-2\"]"
      },
      {
        "type": "TaintsToAdd",
        "value": "[{\"effect\":\"PREFER_NO_SCHEDULE\",\"value\":\"taint-new-
value-1\",\"key\":\"taint-new-key-1\"}]"
      },
      {
        "type": "TaintsToRemove",
        "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",
\"key\":\"taint-key-1\"}]"
      }
    ],
    "createdAt": "2024-04-08T12:30:55.486000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui grup node terkelola](#) di Panduan Pengguna Amazon EKS.

Contoh 4: Perbarui grup node terkelola untuk memperbarui scaling-config dan update-config untuk node pekerja EKS untuk klaster Amazon EKS

update-nodegroup-config Contoh berikut memperbarui grup node terkelola untuk memperbarui scaling-config dan update-config untuk node pekerja EKS untuk klaster Amazon EKS.

```
aws eks update-nodegroup-config \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup \  
  --scaling-config minSize=1,maxSize=5,desiredSize=2 \  
  --update-config maxUnavailable=2
```

Output:

```
{  
  "update": {  
    "id": "a977160f-59bf-3023-805d-c9826e460aea",  
    "status": "InProgress",  
    "type": "ConfigUpdate",  
    "params": [  
      {  
        "type": "MinSize",  
        "value": "1"  
      },  
      {  
        "type": "MaxSize",  
        "value": "5"  
      },  
      {  
        "type": "DesiredSize",  
        "value": "2"  
      },  
      {  
        "type": "MaxUnavailable",  
        "value": "2"  
      }  
    ],  
    "createdAt": "2024-04-08T12:35:17.036000-04:00",  
    "errors": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui grup node terkelola](#) di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [UpdateNodegroupConfig](#) di Referensi AWS CLI Perintah.

## update-nodegroup-version

Contoh kode berikut menunjukkan cara menggunakan `update-nodegroup-version`.

### AWS CLI

Contoh 1: Perbarui versi Kubernetes atau versi AMI dari grup node terkelola Amazon EKS

`update-nodegroup-version` Contoh berikut memperbarui versi Kubernetes atau versi AMI dari grup node terkelola Amazon EKS ke versi terbaru yang tersedia untuk klaster Kubernetes Anda.

```
aws eks update-nodegroup-version \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup \  
  --no-force
```

### Output:

```
{  
  "update": {  
    "id": "a94ebfc3-6bf8-307a-89e6-7dbaa36421f7",  
    "status": "InProgress",  
    "type": "VersionUpdate",  
    "params": [  
      {  
        "type": "Version",  
        "value": "1.26"  
      },  
      {  
        "type": "ReleaseVersion",  
        "value": "1.26.12-20240329"  
      }  
    ],  
    "createdAt": "2024-04-08T13:16:00.724000-04:00",  
    "errors": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui grup node terkelola](#) di Panduan Pengguna Amazon EKS.

Contoh 2: Perbarui versi Kubernetes atau versi AMI dari grup node terkelola Amazon EKS

update-nodegroup-version Contoh berikut memperbarui versi Kubernetes atau versi AMI dari grup node terkelola Amazon EKS ke versi rilis AMI yang ditentukan.

```
aws eks update-nodegroup-version \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --kubernetes-version '1.26' \
  --release-version '1.26.12-20240307' \
  --no-force
```

Output:

```
{
  "update": {
    "id": "4db06fe1-088d-336b-bdcd-3fdb94995fb7",
    "status": "InProgress",
    "type": "VersionUpdate",
    "params": [
      {
        "type": "Version",
        "value": "1.26"
      },
      {
        "type": "ReleaseVersion",
        "value": "1.26.12-20240307"
      }
    ],
    "createdAt": "2024-04-08T13:13:58.595000-04:00",
    "errors": []
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui grup node terkelola](https://docs.aws.amazon.com/eks/latest/userguide/update-managed-node-group.html) - < [https://docs.aws.amazon.com/eks/latest/userguide/update-managed-node-group .html](https://docs.aws.amazon.com/eks/latest/userguide/update-managed-node-group.html)>` di Panduan Pengguna Amazon EKS.

- Untuk detail API, lihat [UpdateNodegroupVersion](#) di Referensi AWS CLI Perintah.

## Contoh Elastic Beanstalk menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with Elastic Beanstalk.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **abort-environment-update**

Contoh kode berikut menunjukkan cara menggunakan `abort-environment-update`.

AWS CLI

Untuk membatalkan penerapan

Perintah berikut membatalkan penerapan versi aplikasi yang sedang berjalan untuk lingkungan bernama: `my-env`

```
aws elasticbeanstalk abort-environment-update --environment-name my-env
```

- Untuk detail API, lihat [AbortEnvironmentUpdate](#) di Referensi AWS CLI Perintah.

### **check-dns-availability**

Contoh kode berikut menunjukkan cara menggunakan `check-dns-availability`.

## AWS CLI

Untuk memeriksa ketersediaan CNAME

Perintah berikut memeriksa ketersediaan subdomain `my-cname.elasticbeanstalk.com`:

```
aws elasticbeanstalk check-dns-availability --cname-prefix my-cname
```

Output:

```
{
  "Available": true,
  "FullyQualifiedCNAME": "my-cname.elasticbeanstalk.com"
}
```

- Untuk detail API, lihat [CheckDnsAvailability](#) di Referensi AWS CLI Perintah.

## create-application-version

Contoh kode berikut menunjukkan cara menggunakan `create-application-version`.

### AWS CLI

Untuk membuat versi aplikasi baru

Perintah berikut membuat versi baru, "v1" dari aplikasi bernama "MyApp":

```
aws elasticbeanstalk create-application-version --application-name MyApp
--version-label v1 --description MyAppv1 --source-bundle S3Bucket="my-
bucket",S3Key="sample.war" --auto-create-application
```

Aplikasi akan dibuat secara otomatis jika belum ada, karena `auto-create-application` opsi. Bundel sumber adalah `file.war` yang disimpan dalam bucket s3 bernama "my-bucket" yang berisi aplikasi sampel Apache Tomcat.

Output:

```
{
  "ApplicationVersion": {
    "ApplicationName": "MyApp",
    "VersionLabel": "v1",
```



```
"Description": "MyAppv1",
  "DateCreated": "2015-02-03T23:01:25.412Z",
  "DateUpdated": "2015-02-03T23:01:25.412Z",
  "SourceBundle": {
    "S3Bucket": "my-bucket",
    "S3Key": "sample.war"
  }
}
```

- Untuk detail API, lihat [CreateApplicationVersion](#) di Referensi AWS CLI Perintah.

## create-application

Contoh kode berikut menunjukkan cara menggunakan `create-application`.

### AWS CLI

Untuk membuat aplikasi baru

Perintah berikut membuat aplikasi baru bernama "MyApp":

```
aws elasticbeanstalk create-application --application-name MyApp --description "my
application"
```

`create-application` Perintah hanya mengkonfigurasi nama dan deskripsi aplikasi. Untuk mengunggah kode sumber untuk aplikasi, buat versi awal aplikasi menggunakan `create-application-version`. `create-application-version` juga memiliki `auto-create-application` opsi yang memungkinkan Anda membuat aplikasi dan versi aplikasi dalam satu langkah.

Output:

```
{
  "Application": {
    "ApplicationName": "MyApp",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-02-12T18:32:21.181Z",
    "Description": "my application",
    "DateCreated": "2015-02-12T18:32:21.181Z"
  }
}
```

```
}
```

- Untuk detail API, lihat [CreateApplication](#) di Referensi AWS CLI Perintah.

## **create-configuration-template**

Contoh kode berikut menunjukkan cara menggunakan `create-configuration-template`.

### AWS CLI

Untuk membuat template konfigurasi

Perintah berikut membuat template konfigurasi bernama `my-app-v1` dari pengaturan yang diterapkan ke lingkungan dengan ide `rpqsewtp2j`:

```
aws elasticbeanstalk create-configuration-template --application-name my-app --  
template-name my-app-v1 --environment-id e-rpqsewtp2j
```

Output:

```
{  
  "ApplicationName": "my-app",  
  "TemplateName": "my-app-v1",  
  "DateCreated": "2015-08-12T18:40:39Z",  
  "DateUpdated": "2015-08-12T18:40:39Z",  
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"  
}
```

- Untuk detail API, lihat [CreateConfigurationTemplate](#) di Referensi AWS CLI Perintah.

## **create-environment**

Contoh kode berikut menunjukkan cara menggunakan `create-environment`.

### AWS CLI

Untuk membuat lingkungan baru untuk aplikasi

Perintah berikut menciptakan lingkungan baru untuk versi “v1” dari aplikasi java bernama “my-app”:

```
aws elasticbeanstalk create-environment --application-name my-app --environment-name my-env --cname-prefix my-app --version-label v1 --solution-stack-name "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
```

### Output:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "v1",
  "Status": "Launching",
  "EnvironmentId": "e-izqpassy4h",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8",
  "CNAME": "my-app.elasticbeanstalk.com",
  "Health": "Grey",
  "Tier": {
    "Type": "Standard",
    "Name": "WebServer",
    "Version": " "
  },
  "DateUpdated": "2015-02-03T23:04:54.479Z",
  "DateCreated": "2015-02-03T23:04:54.479Z"
}
```

v1 adalah label dari versi aplikasi yang sebelumnya diunggah dengan `create-application-version`.

Untuk menentukan file JSON untuk menentukan opsi konfigurasi lingkungan

`create-environment` Perintah berikut menentukan bahwa file JSON dengan nama `myoptions.json` harus digunakan untuk mengganti nilai yang diperoleh dari tumpukan solusi atau template konfigurasi:

```
aws elasticbeanstalk create-environment --environment-name sample-env --application-name sampleapp --option-settings file://myoptions.json
```

`myoptions.json` adalah objek JSON yang mendefinisikan beberapa pengaturan:

```
[
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Interval",
    "Value": "15"
  }
]
```

```
},
{
  "Namespace": "aws:elb:healthcheck",
  "OptionName": "Timeout",
  "Value": "8"
},
{
  "Namespace": "aws:elb:healthcheck",
  "OptionName": "HealthyThreshold",
  "Value": "2"
},
{
  "Namespace": "aws:elb:healthcheck",
  "OptionName": "UnhealthyThreshold",
  "Value": "3"
}
]
```

Untuk informasi selengkapnya, lihat Nilai Opsi dalam Panduan Pengembang AWS Elastic Beanstalk.

- Untuk detail API, lihat [CreateEnvironment](#) di Referensi AWS CLI Perintah.

## create-storage-location

Contoh kode berikut menunjukkan cara menggunakan `create-storage-location`.

### AWS CLI

Untuk membuat lokasi penyimpanan

Perintah berikut membuat lokasi penyimpanan di Amazon S3:

```
aws elasticbeanstalk create-storage-location
```

Output:

```
{
  "S3Bucket": "elasticbeanstalk-us-west-2-0123456789012"
}
```

- Untuk detail API, lihat [CreateStorageLocation](#) di Referensi AWS CLI Perintah.

## **delete-application-version**

Contoh kode berikut menunjukkan cara menggunakan `delete-application-version`.

### AWS CLI

Untuk menghapus versi aplikasi

Perintah berikut menghapus versi aplikasi bernama `22a0-stage-150819_182129` untuk aplikasi bernama `my-app`:

```
aws elasticbeanstalk delete-application-version --version-label 22a0-stage-150819_182129 --application-name my-app
```

- Untuk detail API, lihat [DeleteApplicationVersion](#) di Referensi AWS CLI Perintah.

## **delete-application**

Contoh kode berikut menunjukkan cara menggunakan `delete-application`.

### AWS CLI

Untuk menghapus aplikasi

Perintah berikut menghapus aplikasi bernama `my-app`:

```
aws elasticbeanstalk delete-application --application-name my-app
```

- Untuk detail API, lihat [DeleteApplication](#) di Referensi AWS CLI Perintah.

## **delete-configuration-template**

Contoh kode berikut menunjukkan cara menggunakan `delete-configuration-template`.

### AWS CLI

Untuk menghapus template konfigurasi

Perintah berikut menghapus template konfigurasi bernama `my-template` untuk aplikasi bernama `my-app`:

```
aws elasticbeanstalk delete-configuration-template --template-name my-template --
application-name my-app
```

- Untuk detail API, lihat [DeleteConfigurationTemplate](#) di Referensi AWS CLI Perintah.

## **delete-environment-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-environment-configuration`.

### AWS CLI

Untuk menghapus konfigurasi draf

Perintah berikut menghapus konfigurasi draf untuk lingkungan bernama `my-env`:

```
aws elasticbeanstalk delete-environment-configuration --environment-name my-env --
application-name my-app
```

- Untuk detail API, lihat [DeleteEnvironmentConfiguration](#) di Referensi AWS CLI Perintah.

## **describe-application-versions**

Contoh kode berikut menunjukkan cara menggunakan `describe-application-versions`.

### AWS CLI

Untuk melihat informasi tentang versi aplikasi

Perintah berikut mengambil informasi tentang versi aplikasi berlabel `v2`:

```
aws elasticbeanstalk describe-application-versions --application-name my-app --
version-label "v2"
```

Output:

```
{
  "ApplicationVersions": [
    {
      "ApplicationName": "my-app",
```

```

    "VersionLabel": "v2",
    "Description": "update cover page",
    "DateCreated": "2015-07-23T01:32:26.079Z",
    "DateUpdated": "2015-07-23T01:32:26.079Z",
    "SourceBundle": {
      "S3Bucket": "elasticbeanstalk-us-west-2-015321684451",
      "S3Key": "my-app/5026-stage-150723_224258.war"
    }
  },
  {
    "ApplicationName": "my-app",
    "VersionLabel": "v1",
    "Description": "initial version",
    "DateCreated": "2015-07-23T22:26:10.816Z",
    "DateUpdated": "2015-07-23T22:26:10.816Z",
    "SourceBundle": {
      "S3Bucket": "elasticbeanstalk-us-west-2-015321684451",
      "S3Key": "my-app/5026-stage-150723_222618.war"
    }
  }
]
}

```

- Untuk detail API, lihat [DescribeApplicationVersions](#) di Referensi AWS CLI Perintah.

## describe-applications

Contoh kode berikut menunjukkan cara menggunakan `describe-applications`.

### AWS CLI

Untuk melihat daftar aplikasi

Perintah berikut mengambil informasi tentang aplikasi di wilayah saat ini:

```
aws elasticbeanstalk describe-applications
```

Output:

```
{
  "Applications": [
    {
```

```

    "ApplicationName": "ruby",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-08-13T21:05:44.376Z",
    "Versions": [
      "Sample Application"
    ],
    "DateCreated": "2015-08-13T21:05:44.376Z"
  },
  {
    "ApplicationName": "pythonsample",
    "Description": "Application created from the EB CLI using \"eb init\"",
    "Versions": [
      "Sample Application"
    ],
    "DateCreated": "2015-08-13T19:05:43.637Z",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-08-13T19:05:43.637Z"
  },
  {
    "ApplicationName": "nodejs-example",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-08-06T17:50:02.486Z",
    "Versions": [
      "add elasticache",
      "First Release"
    ],
    "DateCreated": "2015-08-06T17:50:02.486Z"
  }
]
}

```

- Untuk detail API, lihat [DescribeApplications](#) di Referensi AWS CLI Perintah.

## describe-configuration-options

Contoh kode berikut menunjukkan cara menggunakan `describe-configuration-options`.

### AWS CLI

Untuk melihat opsi konfigurasi untuk lingkungan

Perintah berikut mengambil deskripsi dari semua opsi konfigurasi yang tersedia untuk lingkungan bernama: `my-env`



```
aws elasticbeanstalk describe-configuration-options --environment-name my-env --
application-name my-app
```

### Output (disingkat):

```
{
  "Options": [
    {
      "Name": "JVMOptions",
      "UserDefined": false,
      "DefaultValue": "Xms=256m,Xmx=256m,XX:MaxPermSize=64m,JVM Options=",
      "ChangeSeverity": "RestartApplicationServer",
      "Namespace": "aws:cloudformation:template:parameter",
      "ValueType": "KeyValueList"
    },
    {
      "Name": "Interval",
      "UserDefined": false,
      "DefaultValue": "30",
      "ChangeSeverity": "NoInterruption",
      "Namespace": "aws:elb:healthcheck",
      "MaxValue": 300,
      "MinValue": 5,
      "ValueType": "Scalar"
    },
    ...
    {
      "Name": "LowerThreshold",
      "UserDefined": false,
      "DefaultValue": "2000000",
      "ChangeSeverity": "NoInterruption",
      "Namespace": "aws:autoscaling:trigger",
      "MinValue": 0,
      "ValueType": "Scalar"
    },
    {
      "Name": "ListenerEnabled",
      "UserDefined": false,
      "DefaultValue": "true",
      "ChangeSeverity": "Unknown",
      "Namespace": "aws:elb:listener",
      "ValueType": "Boolean"
    }
  ]
}
```

```
]
}
```

Opsi konfigurasi yang tersedia bervariasi per platform dan versi konfigurasi. Untuk informasi selengkapnya tentang ruang nama dan opsi yang didukung, lihat Nilai Opsi dalam Panduan Pengembang Elastic AWS Beanstalk.

- Untuk detail API, lihat [DescribeConfigurationOptions](#) di Referensi AWS CLI Perintah.

## describe-configuration-settings

Contoh kode berikut menunjukkan cara menggunakan `describe-configuration-settings`.

### AWS CLI

Untuk melihat pengaturan konfigurasi untuk lingkungan

Perintah berikut mengambil pengaturan konfigurasi untuk lingkungan bernama `my-env`:

```
aws elasticbeanstalk describe-configuration-settings --environment-name my-env --
application-name my-app
```

Output (disingkat):

```
{
  "ConfigurationSettings": [
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Description": "Environment created from the EB CLI using \"eb create
\\\"\",
      "DeploymentStatus": "deployed",
      "DateCreated": "2015-08-13T19:16:25Z",
      "OptionSettings": [
        {
          "OptionName": "Availability Zones",
          "ResourceName": "AWSEBAutoScalingGroup",
          "Namespace": "aws:autoscaling:asg",
          "Value": "Any"
        },
        {
          "OptionName": "Cooldown",
          "ResourceName": "AWSEBAutoScalingGroup",
```

```

        "Namespace": "aws:autoscaling:asg",
        "Value": "360"
    },
    ...
    {
        "OptionName": "ConnectionDrainingTimeout",
        "ResourceName": "AWSEBLoadBalancer",
        "Namespace": "aws:elb:policies",
        "Value": "20"
    },
    {
        "OptionName": "ConnectionSettingIdleTimeout",
        "ResourceName": "AWSEBLoadBalancer",
        "Namespace": "aws:elb:policies",
        "Value": "60"
    }
],
    "DateUpdated": "2015-08-13T23:30:07Z",
    "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8
Java 8"
    }
]
}

```

Untuk informasi selengkapnya tentang ruang nama dan opsi yang didukung, lihat Nilai Opsi dalam Panduan Pengembang Elastic AWS Beanstalk.

- Untuk detail API, lihat [DescribeConfigurationSettings](#) di Referensi AWS CLI Perintah.

## describe-environment-health

Contoh kode berikut menunjukkan cara menggunakan `describe-environment-health`.

### AWS CLI

Untuk melihat kesehatan lingkungan

Perintah berikut mengambil informasi kesehatan secara keseluruhan untuk lingkungan bernama `my-env`:

```
aws elasticbeanstalk describe-environment-health --environment-name my-env --
attribute-names All
```

## Output:

```
{
  "Status": "Ready",
  "EnvironmentName": "my-env",
  "Color": "Green",
  "ApplicationMetrics": {
    "Duration": 10,
    "Latency": {
      "P99": 0.004,
      "P75": 0.002,
      "P90": 0.003,
      "P95": 0.004,
      "P85": 0.003,
      "P10": 0.001,
      "P999": 0.004,
      "P50": 0.001
    },
    "RequestCount": 45,
    "StatusCodes": {
      "Status3xx": 0,
      "Status2xx": 45,
      "Status5xx": 0,
      "Status4xx": 0
    }
  },
  "RefreshedAt": "2015-08-20T21:09:18Z",
  "HealthStatus": "Ok",
  "InstancesHealth": {
    "Info": 0,
    "Ok": 1,
    "Unknown": 0,
    "Severe": 0,
    "Warning": 0,
    "Degraded": 0,
    "NoData": 0,
    "Pending": 0
  },
  "Causes": []
}
```

Informasi Kesehatan hanya tersedia untuk lingkungan dengan pelaporan kesehatan yang ditingkatkan diaktifkan. Untuk informasi lebih lanjut, lihat Pelaporan dan Pemantauan Kesehatan yang Ditingkatkan di Panduan Pengembang AWS Elastic Beanstalk.

- Untuk detail API, lihat [DescribeEnvironmentHealth](#) di Referensi AWS CLI Perintah.

## describe-environment-resources

Contoh kode berikut menunjukkan cara menggunakan `describe-environment-resources`.

### AWS CLI

Untuk melihat informasi tentang AWS sumber daya di lingkungan Anda

Perintah berikut mengambil informasi tentang sumber daya dalam lingkungan bernama `my-env`:

```
aws elasticbeanstalk describe-environment-resources --environment-name my-env
```

Output:

```
{
  "EnvironmentResources": {
    "EnvironmentName": "my-env",
    "AutoScalingGroups": [
      {
        "Name": "awseb-e-qu3fyyjyjs-stack-AWSEBAutoScalingGroup-
QSB2Z088SXZT"
      }
    ],
    "Triggers": [],
    "LoadBalancers": [
      {
        "Name": "awseb-e-q-AWSEBLoa-1EEPZ0K98BIF0"
      }
    ],
    "Queues": [],
    "Instances": [
      {
        "Id": "i-0c91c786"
      }
    ],
    "LaunchConfigurations": [
      {
```

```

        "Name": "awseb-e-qu3fyyjyjs-stack-
AWSEBAutoScalingLaunchConfiguration-1UUVQIBC96TQ2"
    }
]
}
}

```

- Untuk detail API, lihat [DescribeEnvironmentResources](#) di Referensi AWS CLI Perintah.

## describe-environments

Contoh kode berikut menunjukkan cara menggunakan `describe-environments`.

### AWS CLI

Untuk melihat informasi tentang lingkungan

Perintah berikut mengambil informasi tentang lingkungan bernama `my-env`:

```
aws elasticbeanstalk describe-environments --environment-names my-env
```

### Output:

```

{
  "Environments": [
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "VersionLabel": "7f58-stage-150812_025409",
      "Status": "Ready",
      "EnvironmentId": "e-rpqsewtp2j",
      "EndpointURL": "awseb-e-w-AWSEBLoa-1483140XB0Q4L-109QXY8121.us-
west-2.elb.amazonaws.com",
      "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8
Java 8",
      "CNAME": "my-env.elasticbeanstalk.com",
      "Health": "Green",
      "AbortableOperationInProgress": false,
      "Tier": {
        "Version": " ",
        "Type": "Standard",
        "Name": "WebServer"
      }
    },
  ],
}

```

```
        "DateUpdated": "2015-08-12T18:16:55.019Z",
        "DateCreated": "2015-08-07T20:48:49.599Z"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeEnvironments](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Untuk melihat acara untuk suatu lingkungan

Perintah berikut mengambil peristiwa untuk lingkungan bernama `my-env`:

```
aws elasticbeanstalk describe-events --environment-name my-env
```

Output (disingkat):

```
{
  "Events": [
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Message": "Environment health has transitioned from Info to Ok.",
      "EventDate": "2015-08-20T07:06:53.535Z",
      "Severity": "INFO"
    },
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Severity": "INFO",
      "RequestId": "b7f3960b-4709-11e5-ba1e-07e16200da41",
      "Message": "Environment update completed successfully.",
      "EventDate": "2015-08-20T07:06:02.049Z"
    },
    ...
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
```

```

    "Severity": "INFO",
    "RequestId": "ca8dfbf6-41ef-11e5-988b-651aa638f46b",
    "Message": "Using elasticbeanstalk-us-west-2-012445113685 as Amazon S3
storage bucket for environment data.",
    "EventDate": "2015-08-13T19:16:27.561Z"
  },
  {
    "ApplicationName": "my-app",
    "EnvironmentName": "my-env",
    "Severity": "INFO",
    "RequestId": "cdfba8f6-41ef-11e5-988b-65638f41aa6b",
    "Message": "createEnvironment is starting.",
    "EventDate": "2015-08-13T19:16:26.581Z"
  }
]
}

```

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## describe-instances-health

Contoh kode berikut menunjukkan cara menggunakan `describe-instances-health`.

### AWS CLI

Untuk melihat kesehatan lingkungan

Perintah berikut mengambil informasi kesehatan untuk instance di lingkungan bernama: `my-env`

```
aws elasticbeanstalk describe-instances-health --environment-name my-env --
attribute-names All
```

Output:

```

{
  "InstanceHealthList": [
    {
      "InstanceId": "i-08691cc7",
      "ApplicationMetrics": {
        "Duration": 10,
        "Latency": {
          "P99": 0.006,
          "P75": 0.002,

```



```
        "P90": 0.004,  
        "P95": 0.005,  
        "P85": 0.003,  
        "P10": 0.0,  
        "P999": 0.006,  
        "P50": 0.001  
    },  
    "RequestCount": 48,  
    "StatusCodes": {  
        "Status3xx": 0,  
        "Status2xx": 47,  
        "Status5xx": 0,  
        "Status4xx": 1  
    }  
},  
"System": {  
    "LoadAverage": [  
        0.0,  
        0.02,  
        0.05  
    ],  
    "CPUUtilization": {  
        "SoftIRQ": 0.1,  
        "IOWait": 0.2,  
        "System": 0.3,  
        "Idle": 97.8,  
        "User": 1.5,  
        "IRQ": 0.0,  
        "Nice": 0.1  
    }  
},  
"Color": "Green",  
"HealthStatus": "Ok",  
"LaunchedAt": "2015-08-13T19:17:09Z",  
"Causes": []  
}  
],  
"RefreshedAt": "2015-08-20T21:09:08Z"  
}
```

Informasi Kesehatan hanya tersedia untuk lingkungan dengan pelaporan kesehatan yang ditingkatkan diaktifkan. Untuk informasi lebih lanjut, lihat Pelaporan dan Pemantauan Kesehatan yang Ditingkatkan di Panduan Pengembang AWS Elastic Beanstalk.

- Untuk detail API, lihat [DescribeInstancesHealth](#) di Referensi AWS CLI Perintah.

## list-available-solution-stacks

Contoh kode berikut menunjukkan cara menggunakan `list-available-solution-stacks`.

### AWS CLI

Untuk melihat tumpukan solusi

Perintah berikut mencantumkan tumpukan solusi untuk semua konfigurasi platform yang tersedia saat ini dan semua yang telah Anda gunakan di masa lalu:

```
aws elasticbeanstalk list-available-solution-stacks
```

Output (disingkat):

```
{
  "SolutionStacks": [
    "64bit Amazon Linux 2015.03 v2.0.0 running Node.js",
    "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.6",
    "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.5",
    "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.4",
    "64bit Amazon Linux 2015.03 v2.0.0 running Python 3.4",
    "64bit Amazon Linux 2015.03 v2.0.0 running Python 2.7",
    "64bit Amazon Linux 2015.03 v2.0.0 running Python",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.2 (Puma)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.2 (Passenger Standalone)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.1 (Puma)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.1 (Passenger Standalone)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.0 (Puma)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.0 (Passenger Standalone)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 1.9.3",
    "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8",
    "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 7 Java 7",
    "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 7 Java 6",
    "64bit Windows Server Core 2012 R2 running IIS 8.5",
    "64bit Windows Server 2012 R2 running IIS 8.5",
    "64bit Windows Server 2012 running IIS 8",
    "64bit Windows Server 2008 R2 running IIS 7.5",
    "64bit Amazon Linux 2015.03 v2.0.0 running Docker 1.6.2",
    "64bit Amazon Linux 2015.03 v2.0.0 running Multi-container Docker 1.6.2
    (Generic)",
  ]
}
```

```

    "64bit Debian jessie v2.0.0 running GlassFish 4.1 Java 8 (Preconfigured -
    Docker)",
    "64bit Debian jessie v2.0.0 running GlassFish 4.0 Java 7 (Preconfigured -
    Docker)",
    "64bit Debian jessie v2.0.0 running Go 1.4 (Preconfigured - Docker)",
    "64bit Debian jessie v2.0.0 running Go 1.3 (Preconfigured - Docker)",
    "64bit Debian jessie v2.0.0 running Python 3.4 (Preconfigured - Docker)",
  ],
  "SolutionStackDetails": [
    {
      "PermittedFileTypes": [
        "zip"
      ],
      "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Node.js"
    },
    ...
  ]
}

```

- Untuk detail API, lihat [ListAvailableSolutionStacks](#) di Referensi AWS CLI Perintah.

## rebuild-environment

Contoh kode berikut menunjukkan cara menggunakan `rebuild-environment`.

### AWS CLI

Untuk membangun kembali lingkungan

Perintah berikut mengakhiri dan membuat ulang sumber daya dalam lingkungan bernama: `my-env`

```
aws elasticbeanstalk rebuild-environment --environment-name my-env
```

- Untuk detail API, lihat [RebuildEnvironment](#) di Referensi AWS CLI Perintah.

## request-environment-info

Contoh kode berikut menunjukkan cara menggunakan `request-environment-info`.

## AWS CLI

Untuk meminta log berekor

Perintah berikut meminta log dari lingkungan bernama `my-env`:

```
aws elasticbeanstalk request-environment-info --environment-name my-env --info-type tail
```

Setelah meminta log, ambil lokasinya dengan `retrieve-environment-info`

- Untuk detail API, lihat [RequestEnvironmentInfo](#) di Referensi AWS CLI Perintah.

## **restart-app-server**

Contoh kode berikut menunjukkan cara menggunakan `restart-app-server`.

## AWS CLI

Untuk me-restart server aplikasi

Perintah berikut me-restart server aplikasi pada semua instance dalam lingkungan bernama: `my-env`

```
aws elasticbeanstalk restart-app-server --environment-name my-env
```

- Untuk detail API, lihat [RestartAppServer](#) di Referensi AWS CLI Perintah.

## **retrieve-environment-info**

Contoh kode berikut menunjukkan cara menggunakan `retrieve-environment-info`.

## AWS CLI

Untuk mengambil log berekor

Perintah berikut mengambil link ke log dari lingkungan bernama `my-env`:

```
aws elasticbeanstalk retrieve-environment-info --environment-name my-env --info-type tail
```

Output:

```
{
  "EnvironmentInfo": [
    {
      "SampleTimestamp": "2015-08-20T22:23:17.703Z",
      "Message": "https://elasticbeanstalk-us-
west-2-0123456789012.s3.amazonaws.com/resources/environments/
logs/tail/e-fyqyju3yjs/i-09c1c867/TailLogs-1440109397703.out?
AWSAccessKeyId=AKGPT4J56IAJ2EUBL5CQ&Expires=1440195891&Signature=n
%2BEa10V6A2HI0x4Rcfb7LT16bBM%3D",
      "InfoType": "tail",
      "Ec2InstanceId": "i-09c1c867"
    }
  ]
}
```

Lihat tautan di browser. Sebelum pengambilan, log harus diminta dengan `request-environment-info`.

- Untuk detail API, lihat [RetrieveEnvironmentInfo](#) di Referensi AWS CLI Perintah.

## **swap-environment-cnames**

Contoh kode berikut menunjukkan cara menggunakan `swap-environment-cnames`.

### AWS CLI

Untuk menukar CNAMEs lingkungan

Perintah berikut menukar subdomain yang ditetapkan dari dua lingkungan:

```
aws elasticbeanstalk swap-environment-cnames --source-environment-name my-env-blue
--destination-environment-name my-env-green
```

- Untuk detail API, lihat [SwapEnvironmentCnames](#) di Referensi AWS CLI Perintah.

## **terminate-environment**

Contoh kode berikut menunjukkan cara menggunakan `terminate-environment`.

### AWS CLI

Untuk mengakhiri lingkungan

Perintah berikut mengakhiri lingkungan Elastic Beanstalk bernama: my-env

```
aws elasticbeanstalk terminate-environment --environment-name my-env
```

Output:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "Status": "Terminating",
  "EnvironmentId": "e-fh2eravpns",
  "EndpointURL": "awseb-e-f-AWSEBLoa-1I9XUMP4-8492WNUP202574.us-
west-2.elb.amazonaws.com",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java
8",
  "CNAME": "my-env.elasticbeanstalk.com",
  "Health": "Grey",
  "AbortableOperationInProgress": false,
  "Tier": {
    "Version": " ",
    "Type": "Standard",
    "Name": "WebServer"
  },
  "DateUpdated": "2015-08-12T19:05:54.744Z",
  "DateCreated": "2015-08-12T18:52:53.622Z"
}
```

- Untuk detail API, lihat [TerminateEnvironment](#) di Referensi AWS CLI Perintah.

## update-application-version

Contoh kode berikut menunjukkan cara menggunakan `update-application-version`.

### AWS CLI

Untuk mengubah deskripsi versi aplikasi

Perintah berikut memperbarui deskripsi versi aplikasi bernama `22a0-stage-150819_185942`:

```
aws elasticbeanstalk update-application-version --version-label 22a0-
stage-150819_185942 --application-name my-app --description "new description"
```

**Output:**

```
{
  "ApplicationVersion": {
    "ApplicationName": "my-app",
    "VersionLabel": "22a0-stage-150819_185942",
    "Description": "new description",
    "DateCreated": "2015-08-19T18:59:17.646Z",
    "DateUpdated": "2015-08-20T22:53:28.871Z",
    "SourceBundle": {
      "S3Bucket": "elasticbeanstalk-us-west-2-0123456789012",
      "S3Key": "my-app/22a0-stage-150819_185942.war"
    }
  }
}
```

- Untuk detail API, lihat [UpdateApplicationVersion](#) di Referensi AWS CLI Perintah.

**update-application**

Contoh kode berikut menunjukkan cara menggunakan `update-application`.

**AWS CLI**

Untuk mengubah deskripsi aplikasi

Perintah berikut memperbarui deskripsi aplikasi bernama `my-app`:

```
aws elasticbeanstalk update-application --application-name my-app --description "my
Elastic Beanstalk application"
```

**Output:**

```
{
  "Application": {
    "ApplicationName": "my-app",
    "Description": "my Elastic Beanstalk application",
    "Versions": [
      "2fba-stage-150819_234450",
      "bf07-stage-150820_214945",
      "93f8",
    ]
  }
}
```

```
        "fd7c-stage-150820_000431",
        "22a0-stage-150819_185942"
    ],
    "DateCreated": "2015-08-13T19:15:50.449Z",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-08-20T22:34:56.195Z"
}
}
```

- Untuk detail API, lihat [UpdateApplication](#) di Referensi AWS CLI Perintah.

## update-configuration-template

Contoh kode berikut menunjukkan cara menggunakan `update-configuration-template`.

### AWS CLI

Untuk memperbarui templat konfigurasi

Perintah berikut menghapus konfigurasi metrik kesehatan CloudWatch kustom yang dikonfigurasi `ConfigDocument` dari template konfigurasi yang disimpan bernama `my-template`:

```
aws elasticbeanstalk update-configuration-template --template-name my-template --application-name my-app --options-to-remove Namespace=aws:elasticbeanstalk:healthreporting:system,OptionName=ConfigDocument
```

Output:

```
{
  "ApplicationName": "my-app",
  "TemplateName": "my-template",
  "DateCreated": "2015-08-20T22:39:31Z",
  "DateUpdated": "2015-08-20T22:43:11Z",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
}
```

Untuk informasi selengkapnya tentang ruang nama dan opsi yang didukung, lihat Nilai Opsi dalam Panduan Pengembang Elastic AWS Beanstalk.

- Untuk detail API, lihat [UpdateConfigurationTemplate](#) di Referensi AWS CLI Perintah.



## update-environment

Contoh kode berikut menunjukkan cara menggunakan `update-environment`.

### AWS CLI

Untuk memperbarui lingkungan ke versi baru

Perintah berikut memperbarui lingkungan bernama “my-env” ke versi “v2” dari aplikasi yang dimilikinya:

```
aws elasticbeanstalk update-environment --environment-name my-env --version-label v2
```

Perintah ini mensyaratkan bahwa lingkungan “my-env” sudah ada dan milik aplikasi yang memiliki versi aplikasi yang valid dengan label “v2”.

Output:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "v2",
  "Status": "Updating",
  "EnvironmentId": "e-szqipays4h",
  "EndpointURL": "awseb-e-i-AWSEBLoa-1RD LX6TC9VUA0-0123456789.us-
west-2.elb.amazonaws.com",
  "SolutionStackName": "64bit Amazon Linux running Tomcat 7",
  "CNAME": "my-env.elasticbeanstalk.com",
  "Health": "Grey",
  "Tier": {
    "Version": " ",
    "Type": "Standard",
    "Name": "WebServer"
  },
  "DateUpdated": "2015-02-03T23:12:29.119Z",
  "DateCreated": "2015-02-03T23:04:54.453Z"
}
```

Untuk mengatur variabel lingkungan

Perintah berikut menetapkan nilai variabel “PARAM1” di lingkungan “my-env” ke “”: ParamValue

```
aws elasticbeanstalk update-environment --environment-name my-env --option-settings
Namespace=aws:elasticbeanstalk:application:environment,OptionName=PARAM1,Value=ParamValue
```

`option-settings` Parameter mengambil namespace selain nama dan nilai variabel. Elastic Beanstalk mendukung beberapa ruang nama untuk opsi selain variabel lingkungan.

Untuk mengkonfigurasi pengaturan opsi dari file

Perintah berikut mengkonfigurasi beberapa opsi di `aws:elb:loadbalancer` namespace dari file:

```
aws elasticbeanstalk update-environment --environment-name my-env --option-settings
file://options.json
```

`options.json` adalah objek JSON yang mendefinisikan beberapa pengaturan:

```
[
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Interval",
    "Value": "15"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Timeout",
    "Value": "8"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "HealthyThreshold",
    "Value": "2"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "UnhealthyThreshold",
    "Value": "3"
  }
]
```

Output:

```
{
```

```
"ApplicationName": "my-app",
"EnvironmentName": "my-env",
"VersionLabel": "7f58-stage-150812_025409",
"Status": "Updating",
"EnvironmentId": "e-wtp2rqpsej",
"EndpointURL": "awseb-e-w-AWSEBLoa-14XB83101Q4L-104QXY80921.sa-
east-1.elb.amazonaws.com",
"SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java
8",
"CNAME": "my-env.elasticbeanstalk.com",
"Health": "Grey",
"AbortableOperationInProgress": true,
"Tier": {
  "Version": " ",
  "Type": "Standard",
  "Name": "WebServer"
},
"DateUpdated": "2015-08-12T18:15:23.804Z",
"DateCreated": "2015-08-07T20:48:49.599Z"
}
```

Untuk informasi selengkapnya tentang ruang nama dan opsi yang didukung, lihat Nilai Opsi dalam Panduan Pengembang Elastic AWS Beanstalk.

- Untuk detail API, lihat [UpdateEnvironment](#) di Referensi AWS CLI Perintah.

## validate-configuration-settings

Contoh kode berikut menunjukkan cara menggunakan `validate-configuration-settings`.

### AWS CLI

Untuk memvalidasi pengaturan konfigurasi

Perintah berikut memvalidasi dokumen konfigurasi metrik CloudWatch kustom:

```
aws elasticbeanstalk validate-configuration-settings --application-name my-app --
environment-name my-env --option-settings file://options.json
```

`options.json` adalah dokumen JSON yang mencakup satu atau lebih pengaturan konfigurasi untuk memvalidasi:

```
[
```

```

{
  "Namespace": "aws:elasticbeanstalk:healthreporting:system",
  "OptionName": "ConfigDocument",
  "Value": "{\\"CloudWatchMetrics\\": {\\"Environment\\":
{\\"ApplicationLatencyP99.9\\": null,\\"InstancesSevere\\": 60,
\\"ApplicationLatencyP90\\": 60,\\"ApplicationLatencyP99\\": null,
\\"ApplicationLatencyP95\\": 60,\\"InstancesUnknown\\": 60,\\"ApplicationLatencyP85\\":
60,\\"InstancesInfo\\": null,\\"ApplicationRequests2xx\\": null,\\"InstancesDegraded
\\": null,\\"InstancesWarning\\": 60,\\"ApplicationLatencyP50\\": 60,
\\"ApplicationRequestsTotal\\": null,\\"InstancesNoData\\": null,\\"InstancesPending
\\": 60,\\"ApplicationLatencyP10\\": null,\\"ApplicationRequests5xx\\": null,
\\"ApplicationLatencyP75\\": null,\\"InstancesOk\\": 60,\\"ApplicationRequests3xx\\":
null,\\"ApplicationRequests4xx\\": null},\\"Instance\\": {\\"ApplicationLatencyP99.9\\":
null,\\"ApplicationLatencyP90\\": 60,\\"ApplicationLatencyP99\\": null,
\\"ApplicationLatencyP95\\": null,\\"ApplicationLatencyP85\\": null,\\"CPUUser\\": 60,
\\"ApplicationRequests2xx\\": null,\\"CPUIdle\\": null,\\"ApplicationLatencyP50\\":
null,\\"ApplicationRequestsTotal\\": 60,\\"RootFilesystemUtil\\": null,
\\"LoadAverage1min\\": null,\\"CPUIrq\\": null,\\"CPUNice\\": 60,\\"CPUiowait\\": 60,
\\"ApplicationLatencyP10\\": null,\\"LoadAverage5min\\": null,\\"ApplicationRequests5xx
\\": null,\\"ApplicationLatencyP75\\": 60,\\"CPUSystem\\": 60,\\"ApplicationRequests3xx\\":
60,\\"ApplicationRequests4xx\\": null,\\"InstanceHealth\\": null,\\"CPUSoftirq\\": 60}},
\\"Version\\": 1}"
}
]

```

Jika opsi yang Anda tentukan valid untuk lingkungan tertentu, Elastic Beanstalk mengembalikan array Pesan kosong:

```

{
  "Messages": []
}

```

Jika validasi gagal, respons akan mencakup informasi tentang kesalahan:

```

{
  "Messages": [
    {
      "OptionName": "ConfigDocumet",
      "Message": "Invalid option specification (Namespace:
'aws:elasticbeanstalk:healthreporting:system', OptionName: 'ConfigDocumet'):
Unknown configuration setting.",
      "Namespace": "aws:elasticbeanstalk:healthreporting:system",
      "Severity": "error"
    }
  ]
}

```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya tentang ruang nama dan opsi yang didukung, lihat Nilai Opsi dalam Panduan Pengembang Elastic AWS Beanstalk.

- Untuk detail API, lihat [ValidateConfigurationSettings](#) di Referensi AWS CLI Perintah.

## Elastic Load Balancing - Contoh versi 1 menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Elastic Load Balancing - Versi 1.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags**

Contoh kode berikut menunjukkan cara menggunakan `add-tags`.

AWS CLI

Untuk menambahkan tag ke penyeimbang beban

Contoh ini menambahkan tag ke penyeimbang beban yang ditentukan.

Perintah:

```
aws elb add-tags --load-balancer-name my-load-balancer --tags
"Key=project,Value=lima" "Key=department,Value=digital-media"
```

- Untuk detail API, lihat [AddTags](#) di Referensi AWS CLI Perintah.

## **apply-security-groups-to-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `apply-security-groups-to-load-balancer`.

### AWS CLI

Untuk mengaitkan grup keamanan dengan penyeimbang beban di VPC

Contoh ini mengaitkan grup keamanan dengan penyeimbang beban yang ditentukan dalam VPC.

Perintah:

```
aws elb apply-security-groups-to-load-balancer --load-balancer-name my-load-balancer
--security-groups sg-fc448899
```

Output:

```
{
  "SecurityGroups": [
    "sg-fc448899"
  ]
}
```

- Untuk detail API, lihat [ApplySecurityGroupsToLoadBalancer](#) di Referensi AWS CLI Perintah.

## **attach-load-balancer-to-subnets**

Contoh kode berikut menunjukkan cara menggunakan `attach-load-balancer-to-subnets`.

### AWS CLI

Untuk melampirkan subnet ke penyeimbang beban

Contoh ini menambahkan subnet yang ditentukan ke set subnet yang dikonfigurasi untuk penyeimbang beban yang ditentukan.

**Perintah:**

```
aws elb attach-load-balancer-to-subnets --load-balancer-name my-load-balancer --subnets subnet-0ecac448
```

**Output:**

```
{
  "Subnets": [
    "subnet-15aaab61",
    "subnet-0ecac448"
  ]
}
```

- Untuk detail API, lihat [AttachLoadBalancerToSubnets](#) di Referensi AWS CLI Perintah.

**configure-health-check**

Contoh kode berikut menunjukkan cara menggunakan `configure-health-check`.

**AWS CLI**

Untuk menentukan pengaturan pemeriksaan kesehatan untuk instans EC2 backend Anda

Contoh ini menentukan pengaturan pemeriksaan kesehatan yang digunakan untuk mengevaluasi kesehatan instans EC2 backend Anda.

**Perintah:**

```
aws elb configure-health-check --load-balancer-name my-load-balancer --health-check Target=HTTP:80/png,Interval=30,UnhealthyThreshold=2,HealthyThreshold=2,Timeout=3
```

**Output:**

```
{
  "HealthCheck": {
    "HealthyThreshold": 2,
    "Interval": 30,
    "Target": "HTTP:80/png",
    "Timeout": 3,
    "UnhealthyThreshold": 2
  }
}
```

```
}  
}
```

- Untuk detail API, lihat [ConfigureHealthCheck](#) di Referensi AWS CLI Perintah.

## **create-app-cookie-stickiness-policy**

Contoh kode berikut menunjukkan cara menggunakan `create-app-cookie-stickiness-policy`.

### AWS CLI

Untuk menghasilkan kebijakan lengket untuk penyeimbang beban HTTPS Anda

Contoh ini menghasilkan kebijakan lengket yang mengikuti masa pakai sesi lengket dari cookie yang dihasilkan aplikasi.

Perintah:

```
aws elb create-app-cookie-stickiness-policy --load-balancer-name my-load-balancer --  
policy-name my-app-cookie-policy --cookie-name my-app-cookie
```

- Untuk detail API, lihat [CreateAppCookieStickinessPolicy](#) di Referensi AWS CLI Perintah.

## **create-lb-cookie-stickiness-policy**

Contoh kode berikut menunjukkan cara menggunakan `create-lb-cookie-stickiness-policy`.

### AWS CLI

Untuk menghasilkan kebijakan lengket berbasis durasi untuk penyeimbang beban HTTPS Anda

Contoh ini menghasilkan kebijakan lengket dengan masa pakai sesi lengket yang dikendalikan oleh periode kedaluwarsa yang ditentukan.

Perintah:

```
aws elb create-lb-cookie-stickiness-policy --load-balancer-name my-load-balancer --  
policy-name my-duration-cookie-policy --cookie-expiration-period 60
```

- Untuk detail API, lihat [CreateLbCookieStickinessPolicy](#) di Referensi AWS CLI Perintah.



## **create-load-balancer-listeners**

Contoh kode berikut menunjukkan cara menggunakan `create-load-balancer-listeners`.

### AWS CLI

Untuk membuat pendengar HTTP untuk penyeimbang beban

Contoh ini membuat pendengar untuk penyeimbang beban Anda di port 80 menggunakan protokol HTTP.

Perintah:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --  
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
```

Untuk membuat pendengar HTTPS untuk penyeimbang beban

Contoh ini membuat listener untuk penyeimbang beban Anda di port 443 menggunakan protokol HTTPS.

Perintah:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --  
listeners  
"Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTP,InstancePort=80"
```

- Untuk detail API, lihat [CreateLoadBalancerListeners](#) di Referensi AWS CLI Perintah.

## **create-load-balancer-policy**

Contoh kode berikut menunjukkan cara menggunakan `create-load-balancer-policy`.

### AWS CLI

Untuk membuat kebijakan yang mengaktifkan Protokol Proxy pada penyeimbang beban

Contoh ini membuat kebijakan yang mengaktifkan Protokol Proxy pada penyeimbang beban yang ditentukan.

Perintah:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-ProxyProtocol-policy --policy-type-name ProxyProtocolPolicyType --policy-attributes AttributeName=ProxyProtocol,AttributeValue=true
```

Untuk membuat kebijakan negosiasi SSL menggunakan kebijakan keamanan yang direkomendasikan

Contoh ini membuat kebijakan negosiasi SSL untuk penyeimbang beban HTTPS yang ditentukan menggunakan kebijakan keamanan yang direkomendasikan.

Perintah:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-SSLNegotiation-policy --policy-type-name SSLNegotiationPolicyType --policy-attributes AttributeName=Reference-Security-Policy,AttributeValue=ELBSecurityPolicy-2015-03
```

Untuk membuat kebijakan negosiasi SSL menggunakan kebijakan keamanan khusus

Contoh ini membuat kebijakan negosiasi SSL untuk penyeimbang beban HTTPS Anda menggunakan kebijakan keamanan khusus dengan mengaktifkan protokol dan cipher.

Perintah:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-SSLNegotiation-policy --policy-type-name SSLNegotiationPolicyType --policy-attributes AttributeName=Protocol-SSLv3,AttributeValue=true AttributeName=Protocol-TLSv1.1,AttributeValue=true AttributeName=DHE-RSA-AES256-SHA256,AttributeValue=true AttributeName=Server-Defined-Cipher-Order,AttributeValue=true
```

Untuk membuat kebijakan kunci publik

Contoh ini menciptakan kebijakan kunci publik.

Perintah:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-PublicKey-policy --policy-type-name PublicKeyPolicyType --policy-attributes AttributeName=PublicKey,AttributeValue=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAwAYUjnfy+dS74kj//c6x7R0tusUaeQCTgIUkayttRDWchuqo1pHC1u+n5xxXnBB2ejbb2WRsKIQ5rXEeixsjFpFsojpsQKkzhVGI6mJVZBJDVKSHmswnwLBdofLhzv11povBPTHe+o4haAWvDBALJU0pkSI1FecPHcs2hwx14zHoXy1e2k36A64nXW43wtfx5qcVSIxtCE0jnYRg7RPvybaGfQ
```

```
+v6Iaxb/+7J5kEvZhTFQId+bSiJImF1FSUT1W1xwzBZPUbcUkkXDj45vC2s3Z8E  
+Lk7a3uZhvsQHLZnrFuWjBWGWvZ/MhZYgEXAMPLE
```

Untuk membuat kebijakan otentikasi server backend

Contoh ini membuat kebijakan otentikasi server backend yang memungkinkan otentikasi pada instance backend Anda menggunakan kebijakan kunci publik.

Perintah:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-  
balancer --policy-name my-authentication-policy --policy-type-  
name BackendServerAuthenticationPolicyType --policy-attributes  
AttributeName=PublicKeyPolicyName,AttributeValue=my-PublicKey-policy
```

- Untuk detail API, lihat [CreateLoadBalancerPolicy](#) di Referensi AWS CLI Perintah.

## create-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `create-load-balancer`.

### AWS CLI

Untuk membuat penyeimbang beban HTTP

Contoh ini membuat penyeimbang beban dengan pendengar HTTP di VPC.

Perintah:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --listeners  
"Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" --subnets  
subnet-15aaab61 --security-groups sg-a61988c3
```

Output:

```
{  
  "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com"  
}
```

Contoh ini membuat penyeimbang beban dengan pendengar HTTP di EC2-Classic.

Perintah:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --listeners
"Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" --
availability-zones us-west-2a us-west-2b
```

Output:

```
{
  "DNSName": "my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

Untuk membuat penyeimbang beban HTTPS

Contoh ini membuat penyeimbang beban dengan pendengar HTTPS di VPC.

Perintah:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --listeners
"Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
"Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTP,InstancePort=80,SSLCertificateId
certificate/my-server-cert" --subnets subnet-15aaab61 --security-groups sg-a61988c3
```

Output:

```
{
  "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com"
}
```

Contoh ini membuat penyeimbang beban dengan pendengar HTTPS di EC2-Classic.

Perintah:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --listeners
"Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
"Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTP,InstancePort=80,SSLCertificateId
certificate/my-server-cert" --availability-zones us-west-2a us-west-2b
```

Output:

```
{
  "DNSName": "my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

```
}
```

Untuk membuat penyeimbang beban internal

Contoh ini membuat penyeimbang beban internal dengan pendengar HTTP di VPC.

Perintah:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --listeners
"Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" --scheme
internal --subnets subnet-a85db0df --security-groups sg-a61988c3
```

Output:

```
{
  "DNSName": "internal-my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

- Untuk detail API, lihat [CreateLoadBalancer](#) di Referensi AWS CLI Perintah.

## **delete-load-balancer-listeners**

Contoh kode berikut menunjukkan cara menggunakan `delete-load-balancer-listeners`.

AWS CLI

Untuk menghapus listener dari load balancer

Contoh ini menghapus listener untuk port yang ditentukan dari penyeimbang beban yang ditentukan.

Perintah:

```
aws elb delete-load-balancer-listeners --load-balancer-name my-load-balancer --load-
balancer-ports 80
```

- Untuk detail API, lihat [DeleteLoadBalancerListeners](#) di Referensi AWS CLI Perintah.

## **delete-load-balancer-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-load-balancer-policy`.

## AWS CLI

Untuk menghapus kebijakan dari penyeimbang beban

Contoh ini menghapus kebijakan yang ditentukan dari penyeimbang beban yang ditentukan. Kebijakan tidak boleh diaktifkan pada pendengar mana pun.

Perintah:

```
aws elb delete-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-duration-cookie-policy
```

- Untuk detail API, lihat [DeleteLoadBalancerPolicy](#) di Referensi AWS CLI Perintah.

## **delete-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `delete-load-balancer`.

### AWS CLI

Untuk menghapus penyeimbang beban

Contoh ini menghapus penyeimbang beban yang ditentukan.

Perintah:

```
aws elb delete-load-balancer --load-balancer-name my-load-balancer
```

- Untuk detail API, lihat [DeleteLoadBalancer](#) di Referensi AWS CLI Perintah.

## **deregister-instances-from-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `deregister-instances-from-load-balancer`.

### AWS CLI

Untuk membatalkan pendaftaran instance dari penyeimbang beban

Contoh ini membatalkan pendaftaran instance tertentu dari penyeimbang beban yang ditentukan.

Perintah:

```
aws elb deregister-instances-from-load-balancer --load-balancer-name my-load-balancer --instances i-d6f6fae3
```

Output:

```
{
  "Instances": [
    {
      "InstanceId": "i-207d9717"
    },
    {
      "InstanceId": "i-afefb49b"
    }
  ]
}
```

- Untuk detail API, lihat [DeregisterInstancesFromLoadBalancer](#) di Referensi AWS CLI Perintah.

## **describe-account-limits**

Contoh kode berikut menunjukkan cara menggunakan `describe-account-limits`.

### AWS CLI

Untuk menjelaskan batas Classic Load Balancer

`describe-account-limits` Contoh berikut menampilkan detail tentang batas Classic Load Balancer untuk akun Anda AWS .

```
aws elb describe-account-limits
```

Output:

```
{
  "Limits": [
    {
      "Name": "classic-load-balancers",
      "Max": "20"
    },
    {
      "Name": "classic-listeners",
```

```
        "Max": "100"  
      },  
      {  
        "Name": "classic-registered-instances",  
        "Max": "1000"  
      }  
    ]  
  }  
}
```

- Untuk detail API, lihat [DescribeAccountLimits](#) di Referensi AWS CLI Perintah.

## describe-instance-health

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-health`.

### AWS CLI

Untuk menggambarkan kesehatan instans untuk penyeimbang beban

Contoh ini menjelaskan kesehatan instance untuk penyeimbang beban yang ditentukan.

Perintah:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer
```

Output:

```
{  
  "InstanceStates": [  
    {  
      "InstanceId": "i-207d9717",  
      "ReasonCode": "N/A",  
      "State": "InService",  
      "Description": "N/A"  
    },  
    {  
      "InstanceId": "i-afefb49b",  
      "ReasonCode": "N/A",  
      "State": "InService",  
      "Description": "N/A"  
    }  
  ]  
}
```



```
}
```

Untuk menggambarkan kesehatan sebuah instance untuk penyeimbang beban

Contoh ini menjelaskan kesehatan instance yang ditentukan untuk penyeimbang beban yang ditentukan.

Perintah:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer --instances  
i-7299c809
```

Berikut ini adalah contoh respons untuk contoh yang mendaftar.

Output:

```
{  
  "InstanceStates": [  
    {  
      "InstanceId": "i-7299c809",  
      "ReasonCode": "ELB",  
      "State": "OutOfService",  
      "Description": "Instance registration is still in progress."  
    }  
  ]  
}
```

Berikut ini adalah contoh respons untuk contoh yang tidak sehat.

Output:

```
{  
  "InstanceStates": [  
    {  
      "InstanceId": "i-7299c809",  
      "ReasonCode": "Instance",  
      "State": "OutOfService",  
      "Description": "Instance has failed at least the UnhealthyThreshold number  
of health checks consecutively."  
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeInstanceHealth](#) di Referensi AWS CLI Perintah.

## **describe-load-balancer-attributes**

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancer-attributes`.

### AWS CLI

Untuk menggambarkan atribut penyeimbang beban

Contoh ini menjelaskan atribut penyeimbang beban yang ditentukan.

Perintah:

```
aws elb describe-load-balancer-attributes --load-balancer-name my-load-balancer
```

Output:

```
{
  "LoadBalancerAttributes": {
    "ConnectionDraining": {
      "Enabled": false,
      "Timeout": 300
    },
    "CrossZoneLoadBalancing": {
      "Enabled": true
    },
    "ConnectionSettings": {
      "IdleTimeout": 30
    },
    "AccessLog": {
      "Enabled": false
    }
  }
}
```

- Untuk detail API, lihat [DescribeLoadBalancerAttributes](#) di Referensi AWS CLI Perintah.

## **describe-load-balancer-policies**

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancer-policies`.

## AWS CLI

Untuk menjelaskan semua kebijakan yang terkait dengan penyeimbang beban

Contoh ini menjelaskan semua kebijakan yang terkait dengan penyeimbang beban yang ditentukan.

Perintah:

```
aws elb describe-load-balancer-policies --load-balancer-name my-load-balancer
```

Output:

```
{
  "PolicyDescriptions": [
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "ProxyProtocol",
          "AttributeValue": "true"
        }
      ],
      "PolicyName": "my-ProxyProtocol-policy",
      "PolicyTypeName": "ProxyProtocolPolicyType"
    },
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "CookieName",
          "AttributeValue": "my-app-cookie"
        }
      ],
      "PolicyName": "my-app-cookie-policy",
      "PolicyTypeName": "AppCookieStickinessPolicyType"
    },
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "CookieExpirationPeriod",
          "AttributeValue": "60"
        }
      ],
      "PolicyName": "my-duration-cookie-policy",
```

```

    "PolicyTypeName": "LBCookieStickinessPolicyType"
  },
  .
  .
  .
]
}

```

Untuk menjelaskan kebijakan spesifik yang terkait dengan penyeimbang beban

Contoh ini menjelaskan kebijakan tertentu yang terkait dengan penyeimbang beban yang ditentukan.

Perintah:

```
aws elb describe-load-balancer-policies --load-balancer-name my-load-balancer --
policy-name my-authentication-policy
```

Output:

```

{
  "PolicyDescriptions": [
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "PublicKeyPolicyName",
          "AttributeValue": "my-PublicKey-policy"
        }
      ],
      "PolicyName": "my-authentication-policy",
      "PolicyTypeName": "BackendServerAuthenticationPolicyType"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeLoadBalancerPolicies](#) di Referensi AWS CLI Perintah.

## describe-load-balancer-policy-types

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancer-policy-types`.

## AWS CLI

Untuk menjelaskan jenis kebijakan penyeimbang beban yang ditentukan oleh Elastic Load Balancing

Contoh ini menjelaskan jenis kebijakan penyeimbang beban yang dapat Anda gunakan untuk membuat konfigurasi kebijakan untuk penyeimbang beban Anda.

Perintah:

```
aws elb describe-load-balancer-policy-types
```

Output:

```
{
  "PolicyTypeDescriptions": [
    {
      "PolicyAttributeTypeDescriptions": [
        {
          "Cardinality": "ONE",
          "AttributeName": "ProxyProtocol",
          "AttributeType": "Boolean"
        }
      ],
      "PolicyTypeName": "ProxyProtocolPolicyType",
      "Description": "Policy that controls whether to include the IP address and port of the originating request for TCP messages. This policy operates on TCP/SSL listeners only"
    },
    {
      "PolicyAttributeTypeDescriptions": [
        {
          "Cardinality": "ONE",
          "AttributeName": "PublicKey",
          "AttributeType": "String"
        }
      ],
      "PolicyTypeName": "PublicKeyPolicyType",
      "Description": "Policy containing a list of public keys to accept when authenticating the back-end server(s). This policy cannot be applied directly to back-end servers or listeners but must be part of a BackendServerAuthenticationPolicyType."
    }
  ],
}
```

```
{
  "PolicyAttributeTypeDescriptions": [
    {
      "Cardinality": "ONE",
      "AttributeName": "CookieName",
      "AttributeType": "String"
    }
  ],
  "PolicyTypeName": "AppCookieStickinessPolicyType",
  "Description": "Stickiness policy with session lifetimes controlled by the
lifetime of the application-generated cookie. This policy can be associated only
with HTTP/HTTPS listeners."
},
{
  "PolicyAttributeTypeDescriptions": [
    {
      "Cardinality": "ZERO_OR_ONE",
      "AttributeName": "CookieExpirationPeriod",
      "AttributeType": "Long"
    }
  ],
  "PolicyTypeName": "LBCookieStickinessPolicyType",
  "Description": "Stickiness policy with session lifetimes controlled by
the browser (user-agent) or a specified expiration period. This policy can be
associated only with HTTP/HTTPS listeners."
},
{
  "PolicyAttributeTypeDescriptions": [
    .
    .
    .
  ],
  "PolicyTypeName": "SSLNegotiationPolicyType",
  "Description": "Listener policy that defines the ciphers and protocols
that will be accepted by the load balancer. This policy can be associated only with
HTTPS/SSL listeners."
},
{
  "PolicyAttributeTypeDescriptions": [
    {
      "Cardinality": "ONE_OR_MORE",
      "AttributeName": "PublicKeyPolicyName",
      "AttributeType": "PolicyName"
    }
  ]
}
```

```

    ],
    "PolicyTypeName": "BackendServerAuthenticationPolicyType",
    "Description": "Policy that controls authentication to back-end server(s)
and contains one or more policies, such as an instance of a PublicKeyPolicyType.
This policy can be associated only with back-end servers that are using HTTPS/SSL."
  }
]
}

```

- Untuk detail API, lihat [DescribeLoadBalancerPolicyTypes](#) di Referensi AWS CLI Perintah.

## describe-load-balancers

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancers`.

### AWS CLI

Untuk menggambarkan penyeimbang beban Anda

Contoh ini menjelaskan semua penyeimbang beban Anda.

Perintah:

```
aws elb describe-load-balancers
```

Untuk menggambarkan salah satu penyeimbang beban Anda

Contoh ini menjelaskan penyeimbang beban yang ditentukan.

Perintah:

```
aws elb describe-load-balancers --load-balancer-name my-load-balancer
```

Contoh respons berikut adalah untuk penyeimbang beban HTTPS di VPC.

Output:

```

{
  "LoadBalancerDescriptions": [
    {
      "Subnets": [
        "subnet-15aaab61"
      ],

```

```
"CanonicalHostedZoneNameID": "Z3DZXE0EXAMPLE",
"CanonicalHostedZoneName": "my-load-balancer-1234567890.us-
west-2.elb.amazonaws.com",
"ListenerDescriptions": [
  {
    "Listener": {
      "InstancePort": 80,
      "LoadBalancerPort": 80,
      "Protocol": "HTTP",
      "InstanceProtocol": "HTTP"
    },
    "PolicyNames": []
  },
  {
    "Listener": {
      "InstancePort": 443,
      "SSLCertificateId": "arn:aws:iam::123456789012:server-certificate/
my-server-cert",
      "LoadBalancerPort": 443,
      "Protocol": "HTTPS",
      "InstanceProtocol": "HTTPS"
    },
    "PolicyNames": [
      "ELBSecurityPolicy-2015-03"
    ]
  }
],
"HealthCheck": {
  "HealthyThreshold": 2,
  "Interval": 30,
  "Target": "HTTP:80/png",
  "Timeout": 3,
  "UnhealthyThreshold": 2
},
"VPCId": "vpc-a01106c2",
"BackendServerDescriptions": [
  {
    "InstancePort": 80,
    "PolicyNames": [
      "my-ProxyProtocol-policy"
    ]
  }
],
"Instances": [
```



```
{
  "InstanceId": "i-207d9717"
},
{
  "InstanceId": "i-afefb49b"
}
],
"DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com",
"SecurityGroups": [
  "sg-a61988c3"
],
"Policies": {
  "LBCookieStickinessPolicies": [
    {
      "PolicyName": "my-duration-cookie-policy",
      "CookieExpirationPeriod": 60
    }
  ],
  "AppCookieStickinessPolicies": [],
  "OtherPolicies": [
    "my-PublicKey-policy",
    "my-authentication-policy",
    "my-SSLNegotiation-policy",
    "my-ProxyProtocol-policy",
    "ELBSecurityPolicy-2015-03"
  ]
},
"LoadBalancerName": "my-load-balancer",
"CreatedTime": "2015-03-19T03:24:02.650Z",
"AvailabilityZones": [
  "us-west-2a"
],
"Scheme": "internet-facing",
"SourceSecurityGroup": {
  "OwnerAlias": "123456789012",
  "GroupName": "my-elb-sg"
}
}
]
```

- Untuk detail API, lihat [DescribeLoadBalancers](#) di Referensi AWS CLI Perintah.

## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Untuk mendeskripsikan tag yang ditetapkan ke penyeimbang beban

Contoh ini menjelaskan tag yang ditetapkan untuk penyeimbang beban tertentu.

Perintah:

```
aws elb describe-tags --load-balancer-name my-load-balancer
```

Output:

```
{
  "TagDescriptions": [
    {
      "Tags": [
        {
          "Value": "lima",
          "Key": "project"
        },
        {
          "Value": "digital-media",
          "Key": "department"
        }
      ],
      "LoadBalancerName": "my-load-balancer"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## detach-load-balancer-from-subnets

Contoh kode berikut menunjukkan cara menggunakan `detach-load-balancer-from-subnets`.

### AWS CLI

Untuk melepaskan penyeimbang beban dari subnet

Contoh ini melepaskan penyeimbang beban yang ditentukan dari subnet yang ditentukan.

Perintah:

```
aws elb detach-load-balancer-from-subnets --load-balancer-name my-load-balancer --
subnets subnet-0ecac448
```

Output:

```
{
  "Subnets": [
    "subnet-15aaab61"
  ]
}
```

- Untuk detail API, lihat [DetachLoadBalancerFromSubnets](#) di Referensi AWS CLI Perintah.

## **disable-availability-zones-for-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `disable-availability-zones-for-load-balancer`.

AWS CLI

Untuk menonaktifkan Availability Zones untuk load balancer

Contoh ini menghapus Availability Zone yang ditentukan dari kumpulan Availability Zones untuk load balancer yang ditentukan.

Perintah:

```
aws elb disable-availability-zones-for-load-balancer --load-balancer-name my-load-
balancer --availability-zones us-west-2a
```

Output:

```
{
  "AvailabilityZones": [
    "us-west-2b"
  ]
}
```

- Untuk detail API, lihat [DisableAvailabilityZonesForLoadBalancer](#) di Referensi AWS CLI Perintah.

## **enable-availability-zones-for-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `enable-availability-zones-for-load-balancer`.

### AWS CLI

Untuk mengaktifkan Availability Zones untuk load balancer

Contoh ini menambahkan Availability Zone yang ditentukan ke load balancer yang ditentukan.

Perintah:

```
aws elb enable-availability-zones-for-load-balancer --load-balancer-name my-load-balancer --availability-zones us-west-2b
```

Output:

```
{
  "AvailabilityZones": [
    "us-west-2a",
    "us-west-2b"
  ]
}
```

- Untuk detail API, lihat [EnableAvailabilityZonesForLoadBalancer](#) di Referensi AWS CLI Perintah.

## **modify-load-balancer-attributes**

Contoh kode berikut menunjukkan cara menggunakan `modify-load-balancer-attributes`.

### AWS CLI

Untuk memodifikasi atribut penyeimbang beban

Contoh ini memodifikasi `CrossZoneLoadBalancing` atribut penyeimbang beban tertentu.

Perintah:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer --load-balancer-attributes "{\"CrossZoneLoadBalancing\":{\"Enabled\":true}}"
```

Output:

```
{
  "LoadBalancerAttributes": {
    "CrossZoneLoadBalancing": {
      "Enabled": true
    }
  },
  "LoadBalancerName": "my-load-balancer"
}
```

Contoh ini memodifikasi `ConnectionDraining` atribut penyeimbang beban tertentu.

Perintah:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer --load-balancer-attributes "{\"ConnectionDraining\":{\"Enabled\":true,\"Timeout\":300}}"
```

Output:

```
{
  "LoadBalancerAttributes": {
    "ConnectionDraining": {
      "Enabled": true,
      "Timeout": 300
    }
  },
  "LoadBalancerName": "my-load-balancer"
}
```

- Untuk detail API, lihat [ModifyLoadBalancerAttributes](#) di Referensi AWS CLI Perintah.

## register-instances-with-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `register-instances-with-load-balancer`.

## AWS CLI

Untuk mendaftarkan instance dengan penyeimbang beban

Contoh ini mendaftarkan instance tertentu dengan penyeimbang beban yang ditentukan.

Perintah:

```
aws elb register-instances-with-load-balancer --load-balancer-name my-load-balancer
--instances i-d6f6fae3
```

Output:

```
{
  "Instances": [
    {
      "InstanceId": "i-d6f6fae3"
    },
    {
      "InstanceId": "i-207d9717"
    },
    {
      "InstanceId": "i-afefb49b"
    }
  ]
}
```

- Untuk detail API, lihat [RegisterInstancesWithLoadBalancer](#) di Referensi AWS CLI Perintah.

### **remove-tags**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags`.

## AWS CLI

Untuk menghapus tag dari penyeimbang beban

Contoh ini menghapus tag dari penyeimbang beban yang ditentukan.

Perintah:

```
aws elb remove-tags --load-balancer-name my-load-balancer --tags project
```

- Untuk detail API, lihat [RemoveTags](#) di Referensi AWS CLI Perintah.

## set-load-balancer-listener-ssl-certificate

Contoh kode berikut menunjukkan cara menggunakan `set-load-balancer-listener-ssl-certificate`.

### AWS CLI

Untuk memperbarui sertifikat SSL untuk penyeimbang beban HTTPS

Contoh ini menggantikan sertifikat SSL yang ada untuk penyeimbang beban HTTPS yang ditentukan.

Perintah:

```
aws elb set-load-balancer-listener-ssl-certificate --load-balancer-name my-load-balancer --load-balancer-port 443 --ssl-certificate-id arn:aws:iam::123456789012:server-certificate/new-server-cert
```

- Untuk detail API, lihat [SetLoadBalancerListenerSslCertificate](#) di Referensi AWS CLI Perintah.

## set-load-balancer-policies-for-backend-server

Contoh kode berikut menunjukkan cara menggunakan `set-load-balancer-policies-for-backend-server`.

### AWS CLI

Untuk mengganti kebijakan yang terkait dengan port untuk instance backend

Contoh ini menggantikan kebijakan yang saat ini terkait dengan port yang ditentukan.

Perintah:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-load-balancer --instance-port 80 --policy-names my-ProxyProtocol-policy
```

Untuk menghapus semua kebijakan yang saat ini terkait dengan port pada instans backend Anda

Contoh ini menghapus semua kebijakan yang terkait dengan port yang ditentukan.

Perintah:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-load-balancer --instance-port 80 --policy-names []
```

Untuk mengonfirmasi bahwa kebijakan dihapus, gunakan `describe-load-balancer-policies` perintah.

- Untuk detail API, lihat [SetLoadBalancerPoliciesForBackendServer](#) di Referensi AWS CLI Perintah.

### **set-load-balancer-policies-of-listener**

Contoh kode berikut menunjukkan cara menggunakan `set-load-balancer-policies-of-listener`.

AWS CLI

Untuk mengganti kebijakan yang terkait dengan pendengar

Contoh ini menggantikan kebijakan yang saat ini terkait dengan listener yang ditentukan.

Perintah:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-load-balancer --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

Untuk menghapus semua kebijakan yang terkait dengan listener Anda

Contoh ini menghapus semua kebijakan yang saat ini terkait dengan listener tertentu.

Perintah:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-load-balancer --load-balancer-port 443 --policy-names []
```

Untuk mengonfirmasi bahwa kebijakan dihapus dari penyeimbang beban, gunakan `describe-load-balancer-policies` perintah.

- Untuk detail API, lihat [SetLoadBalancerPoliciesOfListener](#) di Referensi AWS CLI Perintah.



## Elastic Load Balancing - Contoh versi 2 menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Elastic Load Balancing - Versi 2.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-listener-certificates**

Contoh kode berikut menunjukkan cara menggunakan `add-listener-certificates`.

AWS CLI

Untuk menambahkan sertifikat ke pendengar yang aman

Contoh ini menambahkan sertifikat yang ditentukan ke pendengar aman yang ditentukan.

Perintah:

```
aws elbv2 add-listener-certificates --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705
```

Output:

```
{
  "Certificates": [
```

```
{
  "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705",
  "IsDefault": false
}
]
```

- Untuk detail API, lihat [AddListenerCertificates](#) di Referensi AWS CLI Perintah.

## add-tags

Contoh kode berikut menunjukkan cara menggunakan `add-tags`.

### AWS CLI

Untuk menambahkan tag ke penyeimbang beban

`add-tags` Contoh berikut menambahkan `project` dan `department` tag ke penyeimbang beban tertentu.

```
aws elbv2 add-tags \
  --resource-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
  --tags "Key=project,Value=lima" "Key=department,Value=digital-media"
```

- Untuk detail API, lihat [AddTags](#) di Referensi AWS CLI Perintah.

## create-listener

Contoh kode berikut menunjukkan cara menggunakan `create-listener`.

### AWS CLI

Contoh 1: Untuk membuat pendengar HTTP

`create-listener` Contoh berikut membuat listener HTTP untuk Application Load Balancer tertentu yang meneruskan permintaan ke grup target yang ditentukan.

```
aws elbv2 create-listener \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
```

```
--protocol HTTP \  
--port 80 \  
--default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Untuk informasi selengkapnya, lihat [Tutorial: Membuat Application Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Application Load Balancers.

Contoh 2: Untuk membuat pendengar HTTPS

`create-listener` Contoh berikut membuat listener HTTPS untuk Application Load Balancer tertentu yang meneruskan permintaan ke grup target yang ditentukan. Anda harus menentukan sertifikat SSL untuk pendengar HTTPS. Anda dapat membuat dan mengelola sertifikat menggunakan AWS Certificate Manager (ACM). Atau, Anda dapat membuat sertifikat menggunakan alat SSL/TLS, mendapatkan sertifikat yang ditandatangani oleh otoritas sertifikat (CA), dan mengunggah sertifikat ke AWS Identity and Access Management (IAM).

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \  
  --protocol HTTPS \  
  --port 443 \  
  --certificates CertificateArn=arn:aws:acm:us-  
west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557 \  
  --ssl-policy ELBSecurityPolicy-2016-08 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Untuk informasi selengkapnya, lihat [Menambahkan pendengar HTTPS](#) di Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

Contoh 3: Untuk membuat pendengar TCP

`create-listener` Contoh berikut membuat pendengar TCP untuk Network Load Balancer tertentu yang meneruskan permintaan ke grup target yang ditentukan.

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \  
  --protocol TCP \  
  --port 80 \  

```

```
--default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78
```

Untuk informasi selengkapnya, lihat [Tutorial: Membuat Network Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Network Load Balancers.

Contoh 4: Untuk membuat pendengar TLS

`create-listener` Contoh berikut membuat pendengar TLS untuk Network Load Balancer tertentu yang meneruskan permintaan ke grup target yang ditentukan. Anda harus menentukan sertifikat SSL untuk pendengar TLS.

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \  
  --protocol TLS \  
  --port 443 \  
  --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557 \  
  --ssl-policy ELBSecurityPolicy-2016-08 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Untuk informasi selengkapnya, lihat [pendengar TLS untuk Network Load Balancer Anda di Panduan Pengguna untuk Network Load Balancers](#).

Contoh 5: Untuk membuat pendengar UDP

`create-listener` Contoh berikut membuat pendengar UDP untuk Network Load Balancer tertentu yang meneruskan permintaan ke grup target yang ditentukan.

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \  
  --protocol UDP \  
  --port 53 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78
```

Untuk informasi selengkapnya, lihat [Tutorial: Membuat Network Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Network Load Balancers.

Contoh 6: Untuk membuat pendengar untuk gateway dan penerusan yang ditentukan

`create-listener` Contoh berikut membuat listener untuk Load Balancer Gateway tertentu yang meneruskan permintaan ke grup target yang ditentukan.

```
aws elbv2 create-listener \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-gateway-load-balancer/e0f9b3d5c7f7d3d6 \
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/my-glb-targets/007ca469fae3bb1615
```

Output:

```
{
  "Listeners": [
    {
      "ListenerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:listener/gwy/my-agw-lb-example2/e0f9b3d5c7f7d3d6/
afc127db15f925de",
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-agw-lb-example2/e0f9b3d5c7f7d3d6",
      "DefaultActions": [
        {
          "Type": "forward",
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/test-tg-agw-2/007ca469fae3bb1615",
          "ForwardConfig": {
            "TargetGroups": [
              {
                "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/test-tg-agw-2/007ca469fae3bb1615"
              }
            ]
          }
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Gateway Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Penyeimbang Beban Gateway.

- Untuk detail API, lihat [CreateListener](#) di Referensi AWS CLI Perintah.

## create-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `create-load-balancer`.

### AWS CLI

Contoh 1: Untuk membuat penyeimbang beban yang menghadap ke Internet

`create-load-balancer` Contoh berikut membuat Application Load Balancer yang menghadap Internet dan mengaktifkan Availability Zones untuk subnet yang ditentukan.

```
aws elbv2 create-load-balancer \  
  --name my-load-balancer \  
  --subnets subnet-b7d581c0 subnet-8360a9e7
```

Output:

```
{  
  "LoadBalancers": [  
    {  
      "Type": "application",  
      "Scheme": "internet-facing",  
      "IpAddressType": "ipv4",  
      "VpcId": "vpc-3ac0fb5f",  
      "AvailabilityZones": [  
        {  
          "ZoneName": "us-west-2a",  
          "SubnetId": "subnet-8360a9e7"  
        },  
        {  
          "ZoneName": "us-west-2b",  
          "SubnetId": "subnet-b7d581c0"  
        }  
      ],  
      "CreatedTime": "2017-08-25T21:26:12.920Z",  
      "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",  
      "DNSName": "my-load-balancer-424835706.us-west-2.elb.amazonaws.com",  
      "SecurityGroups": [  
        "sg-5943793c"  
      ],  
    }  
  ],  
}
```

```

    "LoadBalancerName": "my-load-balancer",
    "State": {
      "Code": "provisioning"
    },
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Tutorial: Membuat Application Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Application Load Balancers.

Contoh 2: Untuk membuat penyeimbang beban internal

`create-load-balancer` Contoh berikut membuat Application Load Balancer internal dan mengaktifkan Availability Zones untuk subnet yang ditentukan.

```

aws elbv2 create-load-balancer \
  --name my-internal-load-balancer \
  --scheme internal \
  --subnets subnet-b7d581c0 subnet-8360a9e7

```

Output:

```

{
  "LoadBalancers": [
    {
      "Type": "application",
      "Scheme": "internal",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
        {
          "ZoneName": "us-west-2a",
          "SubnetId": "subnet-8360a9e7"
        },
        {
          "ZoneName": "us-west-2b",
          "SubnetId": "subnet-b7d581c0"
        }
      ],
      "CreatedTime": "2016-03-25T21:29:48.850Z",

```

```

        "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
        "DNSName": "internal-my-internal-load-balancer-1529930873.us-
west-2.elb.amazonaws.com",
        "SecurityGroups": [
            "sg-5943793c"
        ],
        "LoadBalancerName": "my-internal-load-balancer",
        "State": {
            "Code": "provisioning"
        },
        "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-internal-load-balancer/5b49b8d4303115c2"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Tutorial: Membuat Application Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Application Load Balancers.

### Contoh 3: Untuk membuat Network Load Balancer

`create-load-balancer` Contoh berikut membuat Network Load Balancer yang menghadap Internet dan mengaktifkan Availability Zone untuk subnet yang ditentukan. Ini menggunakan pemetaan subnet untuk mengaitkan alamat IP Elastis yang ditentukan dengan antarmuka jaringan yang digunakan oleh node penyeimbang beban untuk Availability Zone.

```

aws elbv2 create-load-balancer \
  --name my-network-load-balancer \
  --type network \
  --subnet-mappings SubnetId=subnet-b7d581c0,AllocationId=eipalloc-64d5890a

```

### Output:

```

{
  "LoadBalancers": [
    {
      "Type": "network",
      "Scheme": "internet-facing",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
        {

```



```

        "LoadBalancerAddresses": [
            {
                "IpAddress": "35.161.207.171",
                "AllocationId": "eipalloc-64d5890a"
            }
        ],
        "ZoneName": "us-west-2b",
        "SubnetId": "subnet-5264e837"
    }
],
"CreatedTime": "2017-10-15T22:41:25.657Z",
"CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
"DNSName": "my-network-load-balancer-5d1b75f4f1cee11e.elb.us-
west-2.amazonaws.com",
"LoadBalancerName": "my-network-load-balancer",
"State": {
    "Code": "provisioning"
},
"LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e"
}
]
}

```

Untuk informasi selengkapnya, lihat [Tutorial: Membuat Network Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Network Load Balancers.

Contoh 4: Untuk membuat Load Balancer Gateway

`create-load-balancer` Contoh berikut membuat Load Balancer Gateway dan mengaktifkan Availability Zones untuk subnet yang ditentukan.

```

aws elbv2 create-load-balancer \
  --name my-gateway-load-balancer \
  --type gateway \
  --subnets subnet-dc83f691 subnet-a62583f9

```

Output:

```

{
  "LoadBalancers": [
    {
      "Type": "gateway",

```

```
"VpcId": "vpc-838475fe",
"AvailabilityZones": [
  {
    "ZoneName": "us-east-1b",
    "SubnetId": "subnet-a62583f9"
  },
  {
    "ZoneName": "us-east-1a",
    "SubnetId": "subnet-dc83f691"
  }
],
"CreatedTime": "2021-07-14T19:33:43.324000+00:00",
"LoadBalancerName": "my-gateway-load-balancer",
"State": {
  "Code": "provisioning"
},
"LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-gateway-load-balancer/dfbb5a7d32cdee79"
}
]
```

Untuk informasi selengkapnya, lihat [Memulai Gateway Load Balancer menggunakan AWS CLI](#) di Panduan Pengguna untuk Penyeimbang Beban Gateway.

- Untuk detail API, lihat [CreateLoadBalancer](#) di Referensi AWS CLI Perintah.

## create-rule

Contoh kode berikut menunjukkan cara menggunakan `create-rule`.

### AWS CLI

Contoh 1: Untuk membuat aturan menggunakan kondisi jalur dan tindakan maju

`create-rule` Contoh berikut membuat aturan yang meneruskan permintaan ke grup target tertentu jika URL berisi pola yang ditentukan.

```
aws elbv2 create-rule \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --priority 5 \
  --conditions file://conditions-pattern.json
```

```
--actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Isi dari `conditions-pattern.json`:

```
[
  {
    "Field": "path-pattern",
    "PathPatternConfig": {
      "Values": ["/images/*"]
    }
  }
]
```

Contoh 2: Untuk membuat aturan menggunakan kondisi host dan respons tetap

`create-rule` Contoh berikut membuat aturan yang memberikan respons tetap jika nama host di header host cocok dengan nama host yang ditentukan.

```
aws elbv2 create-rule \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --priority 10 \
  --conditions file://conditions-host.json \
  --actions file://actions-fixed-response.json
```

Isi dari `conditions-host.json`

```
[
  {
    "Field": "host-header",
    "HostHeaderConfig": {
      "Values": ["*.example.com"]
    }
  }
]
```

Isi dari `actions-fixed-response.json`

```
[
  {
    "Type": "fixed-response",
```

```

    "FixedResponseConfig": {
      "MessageBody": "Hello world",
      "StatusCode": "200",
      "ContentType": "text/plain"
    }
  }
]

```

Contoh 3: Untuk membuat aturan menggunakan kondisi alamat IP sumber, tindakan otentikasi, dan tindakan maju

`create-rule` Contoh berikut membuat aturan yang mengautentikasi pengguna jika alamat IP sumber cocok dengan alamat IP yang ditentukan, dan meneruskan permintaan ke grup target yang ditentukan jika otentikasi berhasil.

```

aws elbv2 create-rule \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --priority 20 \
  --conditions file://conditions-source-ip.json \
  --actions file://actions-authenticate.json

```

Isi dari `conditions-source-ip.json`

```

[
  {
    "Field": "source-ip",
    "SourceIpConfig": {
      "Values": ["192.0.2.0/24", "198.51.100.10/32"]
    }
  }
]

```

Isi dari `actions-authenticate.json`

```

[
  {
    "Type": "authenticate-oidc",
    "AuthenticateOidcConfig": {
      "Issuer": "https://idp-issuer.com",
      "AuthorizationEndpoint": "https://authorization-endpoint.com",
      "TokenEndpoint": "https://token-endpoint.com",

```

```

    "UserInfoEndpoint": "https://user-info-endpoint.com",
    "ClientId": "abcdefghijklmnopqrstuvwxy123456789",
    "ClientSecret": "123456789012345678901234567890",
    "SessionCookieName": "my-cookie",
    "SessionTimeout": 3600,
    "Scope": "email",
    "AuthenticationRequestExtraParams": {
        "display": "page",
        "prompt": "login"
    },
    "OnUnauthenticatedRequest": "deny"
  },
  "Order": 1
},
{
  "Type": "forward",
  "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:880185128111:targetgroup/cli-test/642a97ecb0e0f26b",
  "Order": 2
}
]

```

- Untuk detail API, lihat [CreateRule](#) di Referensi AWS CLI Perintah.

## create-target-group

Contoh kode berikut menunjukkan cara menggunakan `create-target-group`.

### AWS CLI

Contoh 1: Untuk membuat grup target untuk Application Load Balancer

`create-target-group` Contoh berikut membuat grup target untuk Application Load Balancer tempat Anda mendaftarkan target berdasarkan ID instance (tipe targetnya adalah `instance`). Grup target ini menggunakan protokol HTTP, port 80, dan pengaturan pemeriksaan kesehatan default untuk grup target HTTP.

```

aws elbv2 create-target-group \
  --name my-targets \
  --protocol HTTP \
  --port 80 \
  --target-type instance \

```

```
--vpc-id vpc-3ac0fb5f
```

### Output:

```
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
      "TargetGroupName": "my-targets",
      "Protocol": "HTTP",
      "Port": 80,
      "VpcId": "vpc-3ac0fb5f",
      "HealthCheckProtocol": "HTTP",
      "HealthCheckPort": "traffic-port",
      "HealthCheckEnabled": true,
      "HealthCheckIntervalSeconds": 30,
      "HealthCheckTimeoutSeconds": 5,
      "HealthyThresholdCount": 5,
      "UnhealthyThresholdCount": 2,
      "HealthCheckPath": "/",
      "Matcher": {
        "HttpCode": "200"
      },
      "TargetType": "instance",
      "ProtocolVersion": "HTTP1",
      "IpAddressType": "ipv4"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat grup target](#) di Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

Contoh 2: Untuk membuat grup target untuk merutekan lalu lintas dari Application Load Balancer ke fungsi Lambda

`create-target-group` Contoh berikut membuat grup target untuk Application Load Balancer di mana targetnya adalah fungsi Lambda (tipe targetnya adalah). Pemeriksaan kesehatan dinonaktifkan untuk grup target ini secara default.

```
aws elbv2 create-target-group \
```

```
--name my-lambda-target \  
--target-type lambda
```

**Output:**

```
{  
  "TargetGroups": [  
    {  
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-lambda-target/a3003e085dbb8ddc",  
      "TargetGroupName": "my-lambda-target",  
      "HealthCheckEnabled": false,  
      "HealthCheckIntervalSeconds": 35,  
      "HealthCheckTimeoutSeconds": 30,  
      "HealthyThresholdCount": 5,  
      "UnhealthyThresholdCount": 2,  
      "HealthCheckPath": "/",  
      "Matcher": {  
        "HttpCode": "200"  
      },  
      "TargetType": "lambda",  
      "IpAddressType": "ipv4"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat: [Fungsi Lambda sebagai target](#) di Panduan pengguna untuk Application Load Balancers.

**Contoh 3: Untuk membuat grup target untuk Network Load Balancer**

`create-target-group` Contoh berikut membuat grup target untuk Network Load Balancer tempat Anda mendaftarkan target berdasarkan alamat IP (tipe targetnya adalah `ip`). Grup target ini menggunakan protokol TCP, port 80, dan pengaturan pemeriksaan kesehatan default untuk grup target TCP.

```
aws elbv2 create-target-group \  
  --name my-ip-targets \  
  --protocol TCP \  
  --port 80 \  
  --target-type ip \  
  --vpc-id vpc-3ac0fb5f
```

**Output:**

```
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-ip-targets/b6bba954d1361c78",
      "TargetGroupName": "my-ip-targets",
      "Protocol": "TCP",
      "Port": 80,
      "VpcId": "vpc-3ac0fb5f",
      "HealthCheckEnabled": true,
      "HealthCheckProtocol": "TCP",
      "HealthCheckPort": "traffic-port",
      "HealthCheckIntervalSeconds": 30,
      "HealthCheckTimeoutSeconds": 10,
      "HealthyThresholdCount": 5,
      "UnhealthyThresholdCount": 2,
      "TargetType": "ip",
      "IpAddressType": "ipv4"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat grup target](#) di Panduan Pengguna untuk Network Load Balancers.

Contoh 4: Untuk membuat grup target untuk merutekan lalu lintas dari Network Load Balancer ke Application Load Balancer

`create-target-group` Contoh berikut membuat grup target untuk Network Load Balancer tempat Anda mendaftarkan Application Load Balancer sebagai target (tipe targetnya adalah). `alb`

```
aws elbv2 create-target-group --name my-alb-target --protocol TCP --port 80 --target-type alb --
vpc-id vpc-id vpc-3ac0fb5f
```

**Output:**

```
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-alb-target/a3003e085dbb8ddc",
```



```

    "TargetGroupName": "my-alb-target",
    "Protocol": "TCP",
    "Port": 80,
    "VpcId": "vpc-838475fe",
    "HealthCheckProtocol": "HTTP",
    "HealthCheckPort": "traffic-port",
    "HealthCheckEnabled": true,
    "HealthCheckIntervalSeconds": 30,
    "HealthCheckTimeoutSeconds": 6,
    "HealthyThresholdCount": 5,
    "UnhealthyThresholdCount": 2,
    "HealthCheckPath": "/",
    "Matcher": {
        "HttpCode": "200-399"
    },
    "TargetType": "alb",
    "IpAddressType": "ipv4"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat grup target dengan Application Load Balancer sebagai target](#) dalam Panduan Pengguna untuk Network Load Balancers.

Contoh 5: Untuk membuat grup target untuk Load Balancer Gateway

`create-target-group` Contoh berikut membuat grup target untuk Load Balancer Gateway di mana target adalah sebuah instance, dan protokol grup target adalah GENEVE

```

aws elbv2 create-target-group \
  --name my-glb-targetgroup \
  --protocol GENEVE \
  --port 6081 \
  --target-type instance \
  --vpc-id vpc-838475fe

```

Output:

```

{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-glb-targetgroup/00c3d57eacd6f40b6f",

```

```

    "TargetGroupName": "my-glb-targetgroup",
    "Protocol": "GENEVE",
    "Port": 6081,
    "VpcId": "vpc-838475fe",
    "HealthCheckProtocol": "TCP",
    "HealthCheckPort": "80",
    "HealthCheckEnabled": true,
    "HealthCheckIntervalSeconds": 10,
    "HealthCheckTimeoutSeconds": 5,
    "HealthyThresholdCount": 5,
    "UnhealthyThresholdCount": 2,
    "TargetType": "instance"
  }
]
}

```

Untuk informasi selengkapnya, lihat Membuat grup target < <https://docs.aws.amazon.com/elasticloadbalancing/latest/gateway/create-target-group.html>> di Panduan Pengguna Load Balancer Gateway.

- Untuk detail API, lihat [CreateTargetGroup](#) di Referensi AWS CLI Perintah.

## delete-listener

Contoh kode berikut menunjukkan cara menggunakan `delete-listener`.

### AWS CLI

Untuk menghapus pendengar

`delete-listener` Contoh berikut menghapus pendengar yang ditentukan.

```

aws elbv2 delete-listener \
  --listener-arn arn:aws:elasticloadbalancing:ua-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2

```

- Untuk detail API, lihat [DeleteListener](#) di Referensi AWS CLI Perintah.

## delete-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `delete-load-balancer`.

## AWS CLI

Untuk menghapus penyeimbang beban

`delete-load-balancer` Contoh berikut menghapus penyeimbang beban yang ditentukan.

```
aws elbv2 delete-load-balancer \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

- Untuk detail API, lihat [DeleteLoadBalancer](#) di Referensi AWS CLI Perintah.

## `delete-rule`

Contoh kode berikut menunjukkan cara menggunakan `delete-rule`.

## AWS CLI

Untuk menghapus aturan

`delete-rule` Contoh berikut menghapus aturan yang ditentukan.

```
aws elbv2 delete-rule \
  --rule-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3
```

- Untuk detail API, lihat [DeleteRule](#) di Referensi AWS CLI Perintah.

## `delete-target-group`

Contoh kode berikut menunjukkan cara menggunakan `delete-target-group`.

## AWS CLI

Untuk menghapus grup target

`delete-target-group` Contoh berikut menghapus kelompok target yang ditentukan.

```
aws elbv2 delete-target-group \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus penyeimbang beban di Panduan Application Load Balancer](#).

- Untuk detail API, lihat [DeleteTargetGroup](#) di Referensi AWS CLI Perintah.

## deregister-targets

Contoh kode berikut menunjukkan cara menggunakan `deregister-targets`.

### AWS CLI

Contoh 1: Untuk membatalkan pendaftaran target dari grup target

`deregister-targets` Contoh berikut menghapus contoh tertentu dari kelompok target yang ditentukan.

```
aws elbv2 deregister-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \  
  --targets Id=i-1234567890abcdef0
```

Contoh 2: Untuk membatalkan pendaftaran target yang terdaftar menggunakan penggantian port

`deregister-targets` Contoh berikut menghapus instance dari grup target yang terdaftar menggunakan port overrides.

```
aws elbv2 deregister-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-internal-targets/3bb63f11dfb0faf9 \  
  --targets Id=i-1234567890abcdef0,Port=80 Id=i-1234567890abcdef0,Port=766
```

- Untuk detail API, lihat [DeregisterTargets](#) di Referensi AWS CLI Perintah.

## describe-account-limits

Contoh kode berikut menunjukkan cara menggunakan `describe-account-limits`.

### AWS CLI

Untuk menjelaskan batas Elastic Load Balancing

`describe-account-limits` Contoh berikut menampilkan batas Elastic Load Balancing untuk AWS akun Anda di Wilayah saat ini.

```
aws elbv2 describe-account-limits
```

Output:

```
{
  "Limits": [
    {
      "Name": "target-groups",
      "Max": "3000"
    },
    {
      "Name": "targets-per-application-load-balancer",
      "Max": "1000"
    },
    {
      "Name": "listeners-per-application-load-balancer",
      "Max": "50"
    },
    {
      "Name": "rules-per-application-load-balancer",
      "Max": "100"
    },
    {
      "Name": "network-load-balancers",
      "Max": "50"
    },
    {
      "Name": "targets-per-network-load-balancer",
      "Max": "3000"
    },
    {
      "Name": "targets-per-availability-zone-per-network-load-balancer",
      "Max": "500"
    },
    {
      "Name": "listeners-per-network-load-balancer",
      "Max": "50"
    },
    {
      "Name": "condition-values-per-alb-rule",
```

```
    "Max": "5"
  },
  {
    "Name": "condition-wildcards-per-alb-rule",
    "Max": "5"
  },
  {
    "Name": "target-groups-per-application-load-balancer",
    "Max": "100"
  },
  {
    "Name": "target-groups-per-action-on-application-load-balancer",
    "Max": "5"
  },
  {
    "Name": "target-groups-per-action-on-network-load-balancer",
    "Max": "1"
  },
  {
    "Name": "certificates-per-application-load-balancer",
    "Max": "25"
  },
  {
    "Name": "certificates-per-network-load-balancer",
    "Max": "25"
  },
  {
    "Name": "targets-per-target-group",
    "Max": "1000"
  },
  {
    "Name": "target-id-registrations-per-application-load-balancer",
    "Max": "1000"
  },
  {
    "Name": "network-load-balancer-enis-per-vpc",
    "Max": "1200"
  },
  {
    "Name": "application-load-balancers",
    "Max": "50"
  },
  {
    "Name": "gateway-load-balancers",
```

```

        "Max": "100"
      },
      {
        "Name": "gateway-load-balancers-per-vpc",
        "Max": "100"
      },
      {
        "Name": "geneve-target-groups",
        "Max": "100"
      },
      {
        "Name": "targets-per-availability-zone-per-gateway-load-balancer",
        "Max": "300"
      }
    ]
  }
}

```

Untuk informasi lebih lanjut, lihat [Kuota](#) di Referensi AWS Umum.

- Untuk detail API, lihat [DescribeAccountLimits](#) di Referensi AWS CLI Perintah.

## describe-listener-certificates

Contoh kode berikut menunjukkan cara menggunakan `describe-listener-certificates`.

### AWS CLI

Untuk mendeskripsikan sertifikat untuk pendengar yang aman

Contoh ini menjelaskan sertifikat untuk pendengar aman yang ditentukan.

Perintah:

```
aws elbv2 describe-listener-certificates --listener-arn
arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

Output:

```
{
  "Certificates": [
    {
```

```

    "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705",
    "IsDefault": false
  },
  {
    "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557",
    "IsDefault": false
  },
  {
    "CertificateArn": "arn:aws:acm:us-west-2:123456789012:certificate/
fe59da96-6f58-4a22-8eed-6d0d50477e1d",
    "IsDefault": true
  }
]
}

```

- Untuk detail API, lihat [DescribeListenersCertificates](#) di Referensi AWS CLI Perintah.

## describe-listeners

Contoh kode berikut menunjukkan cara menggunakan `describe-listeners`.

### AWS CLI

Untuk menggambarkan pendengar

Contoh ini menjelaskan pendengar yang ditentukan.

Perintah:

```
aws elbv2 describe-listeners --listener-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

Output:

```

{
  "Listeners": [
    {
      "Port": 80,
      "Protocol": "HTTP",
      "DefaultActions": [

```



```

        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
            "Type": "forward"
        }
    ],
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
    "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
}
]
}

```

Untuk menggambarkan pendengar untuk penyeimbang beban

Contoh ini menjelaskan pendengar untuk penyeimbang beban yang ditentukan.

Perintah:

```
aws elbv2 describe-listeners --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

Output:

```

{
  "Listeners": [
    {
      "Port": 443,
      "Protocol": "HTTPS",
      "DefaultActions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
          "Type": "forward"
        }
      ],
      "SslPolicy": "ELBSecurityPolicy-2015-05",
      "Certificates": [
        {
          "CertificateArn": "arn:aws:iam::123456789012:server-certificate/
my-server-cert"
        }
      ]
    }
  ]
}

```

```

    ],
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
    "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65"
  },
  {
    "Port": 80,
    "Protocol": "HTTP",
    "DefaultActions": [
      {
        "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
        "Type": "forward"
      }
    ]
  },
  "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
  "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
}
]
}

```

- Untuk detail API, lihat [DescribeListeners](#) di Referensi AWS CLI Perintah.

## describe-load-balancer-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancer-attributes`.

### AWS CLI

Untuk menggambarkan atribut penyeimbang beban

`describe-load-balancer-attributes` Contoh berikut menampilkan atribut penyeimbang beban tertentu.

```

aws elbv2 describe-load-balancer-attributes \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188

```

Contoh output berikut menunjukkan atribut untuk Application Load Balancer.

```
{
  "Attributes": [
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.bucket"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "60",
      "Key": "idle_timeout.timeout_seconds"
    },
    {
      "Value": "false",
      "Key": "deletion_protection.enabled"
    },
    {
      "Value": "true",
      "Key": "routing.http2.enabled"
    }
  ]
}
```

Contoh output berikut mencakup atribut untuk Network Load Balancer.

```
{
  "Attributes": [
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.bucket"
    },
    {
      "Value": "",
```

```

        "Key": "access_logs.s3.prefix"
    },
    {
        "Value": "false",
        "Key": "deletion_protection.enabled"
    },
    {
        "Value": "false",
        "Key": "load_balancing.cross_zone.enabled"
    }
]
}

```

- Untuk detail API, lihat [DescribeLoadBalancerAttributes](#) di Referensi AWS CLI Perintah.

## describe-load-balancers

Contoh kode berikut menunjukkan cara menggunakan `describe-load-balancers`.

### AWS CLI

Untuk menggambarkan penyeimbang beban

Contoh ini menjelaskan penyeimbang beban yang ditentukan.

Perintah:

```

aws elbv2 describe-load-balancers --load-balancer-arns
arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188

```

Output:

```

{
  "LoadBalancers": [
    {
      "Type": "application",
      "Scheme": "internet-facing",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
        {

```

```

        "ZoneName": "us-west-2a",
        "SubnetId": "subnet-8360a9e7"
    },
    {
        "ZoneName": "us-west-2b",
        "SubnetId": "subnet-b7d581c0"
    }
],
"CreatedTime": "2016-03-25T21:26:12.920Z",
"CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
"DNSName": "my-load-balancer-424835706.us-west-2.elb.amazonaws.com",
"SecurityGroups": [
    "sg-5943793c"
],
"LoadBalancerName": "my-load-balancer",
"State": {
    "Code": "active"
},
"LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188"
}
]
}

```

Untuk menggambarkan semua penyeimbang beban

Contoh ini menjelaskan semua penyeimbang beban Anda.

Perintah:

```
aws elbv2 describe-load-balancers
```

- Untuk detail API, lihat [DescribeLoadBalancers](#) di Referensi AWS CLI Perintah.

## describe-rules

Contoh kode berikut menunjukkan cara menggunakan `describe-rules`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan aturan

`describe-rules` Contoh berikut menampilkan rincian untuk aturan yang ditentukan.

```
aws elbv2 describe-rules \  
  --rule-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/  
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/9683b2d02a6cabee
```

Contoh 2: Untuk mendeskripsikan aturan untuk pendengar

`describe-rules` Contoh berikut menampilkan rincian untuk aturan untuk listener tertentu. Outputnya mencakup aturan default dan aturan lain yang telah Anda tambahkan.

```
aws elbv2 describe-rules \  
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/  
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

- Untuk detail API, lihat [DescribeRules](#) di Referensi AWS CLI Perintah.

## **describe-ssl-policies**

Contoh kode berikut menunjukkan cara menggunakan `describe-ssl-policies`.

### AWS CLI

Contoh 1: Untuk membuat daftar kebijakan yang digunakan untuk negosiasi SSL berdasarkan jenis penyeimbang beban

`describe-ssl-policies` Contoh berikut menampilkan nama-nama kebijakan yang dapat Anda gunakan untuk negosiasi SSL dengan Application Load Balancer. Contoh menggunakan `--query` parameter untuk menampilkan hanya nama-nama kebijakan.

```
aws elbv2 describe-ssl-policies \  
  --load-balancer-type application \  
  --query SslPolicies[*].Name
```

Output:

```
[  
  "ELBSecurityPolicy-2016-08",  
  "ELBSecurityPolicy-TLS13-1-2-2021-06",  
  "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",  
  "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",  
  "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
```

```
"ELBSecurityPolicy-TLS13-1-1-2021-06",
"ELBSecurityPolicy-TLS13-1-0-2021-06",
"ELBSecurityPolicy-TLS13-1-3-2021-06",
"ELBSecurityPolicy-TLS-1-2-2017-01",
"ELBSecurityPolicy-TLS-1-1-2017-01",
"ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
"ELBSecurityPolicy-FS-2018-06",
"ELBSecurityPolicy-2015-05",
"ELBSecurityPolicy-TLS-1-0-2015-04",
"ELBSecurityPolicy-FS-1-2-Res-2019-08",
"ELBSecurityPolicy-FS-1-1-2019-08",
"ELBSecurityPolicy-FS-1-2-2019-08",
"ELBSecurityPolicy-FS-1-2-Res-2020-10"
```

```
]
```

Contoh 2: Untuk membuat daftar kebijakan yang mendukung protokol tertentu

`describe-ssl-policies` Contoh berikut menampilkan nama-nama kebijakan yang mendukung protokol TLS 1.3. Contoh menggunakan `--query` parameter untuk menampilkan hanya nama-nama kebijakan.

```
aws elbv2 describe-ssl-policies \
  --load-balancer-type application \
  --query SslPolicies[?contains(SslProtocols,'TLSv1.3')].Name
```

Output:

```
[
  "ELBSecurityPolicy-TLS13-1-2-2021-06",
  "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
  "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
  "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
  "ELBSecurityPolicy-TLS13-1-1-2021-06",
  "ELBSecurityPolicy-TLS13-1-0-2021-06",
  "ELBSecurityPolicy-TLS13-1-3-2021-06"
```

```
]
```

Contoh 3: Untuk menampilkan cipher untuk kebijakan

`describe-ssl-policies` Contoh berikut menampilkan nama-nama cipher untuk kebijakan yang ditentukan. Contoh menggunakan `--query` parameter untuk menampilkan hanya nama

cipher. Cipher pertama dalam daftar memiliki prioritas 1, dan cipher yang tersisa berada dalam urutan prioritas.

```
aws elbv2 describe-ssl-policies \  
  --names ELBSecurityPolicy-TLS13-1-2-2021-06 \  
  --query SslPolicies[*].Ciphers[*].Name
```

Output:

```
[  
  "TLS_AES_128_GCM_SHA256",  
  "TLS_AES_256_GCM_SHA384",  
  "TLS_CHACHA20_POLY1305_SHA256",  
  "ECDHE-ECDSA-AES128-GCM-SHA256",  
  "ECDHE-RSA-AES128-GCM-SHA256",  
  "ECDHE-ECDSA-AES128-SHA256",  
  "ECDHE-RSA-AES128-SHA256",  
  "ECDHE-ECDSA-AES256-GCM-SHA384",  
  "ECDHE-RSA-AES256-GCM-SHA384",  
  "ECDHE-ECDSA-AES256-SHA384",  
  "ECDHE-RSA-AES256-SHA384"  
]
```

Untuk informasi selengkapnya, lihat [Kebijakan keamanan](#) di Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

- Untuk detail API, lihat [DescribeSslPolicies](#) di Referensi AWS CLI Perintah.

## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Untuk mendeskripsikan tag yang ditetapkan ke penyeimbang beban

Contoh ini menjelaskan tag yang ditetapkan untuk penyeimbang beban yang ditentukan.

Perintah:

```
aws elbv2 describe-tags --resource-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```



**Output:**

```
{
  "TagDescriptions": [
    {
      "ResourceArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
      "Tags": [
        {
          "Value": "lima",
          "Key": "project"
        },
        {
          "Value": "digital-media",
          "Key": "department"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

**describe-target-group-attributes**

Contoh kode berikut menunjukkan cara menggunakan `describe-target-group-attributes`.

**AWS CLI**

Untuk menggambarkan atribut kelompok sasaran

`describe-target-group-attributes` Contoh berikut menampilkan atribut dari kelompok target yang ditentukan.

```
aws elbv2 describe-target-group-attributes \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Output mencakup atribut jika protokolnya HTTP atau HTTPS dan tipe targetnya adalah instance atau ip.

```
{
```

```
"Attributes": [  
  {  
    "Value": "false",  
    "Key": "stickiness.enabled"  
  },  
  {  
    "Value": "300",  
    "Key": "deregistration_delay.timeout_seconds"  
  },  
  {  
    "Value": "lb_cookie",  
    "Key": "stickiness.type"  
  },  
  {  
    "Value": "86400",  
    "Key": "stickiness.lb_cookie.duration_seconds"  
  },  
  {  
    "Value": "0",  
    "Key": "slow_start.duration_seconds"  
  }  
]  
}
```

Output berikut mencakup atribut jika protokolnya HTTP atau HTTPS dan tipe targetnya adalah `lambda`.

```
{  
  "Attributes": [  
    {  
      "Value": "false",  
      "Key": "lambda.multi_value_headers.enabled"  
    }  
  ]  
}
```

Output berikut mencakup atribut jika protokolnya adalah TCP, TLS, UDP, atau TCP\_UDP.

```
{  
  "Attributes": [  
    {  
      "Value": "false",  
      "Key": "proxy_protocol_v2.enabled"  
    }  
  ]  
}
```

```
    },  
    {  
      "Value": "300",  
      "Key": "deregistration_delay.timeout_seconds"  
    }  
  ]  
}
```

- Untuk detail API, lihat [DescribeTargetGroupAttributes](#) di Referensi AWS CLI Perintah.

## describe-target-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-target-groups`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan kelompok sasaran

`describe-target-groups` Contoh berikut menampilkan rincian untuk kelompok target yang ditentukan.

```
aws elbv2 describe-target-groups \  
  --target-group-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Output:

```
{  
  "TargetGroups": [  
    {  
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",  
      "TargetGroupName": "my-targets",  
      "Protocol": "HTTP",  
      "Port": 80,  
      "VpcId": "vpc-3ac0fb5f",  
      "HealthCheckProtocol": "HTTP",  
      "HealthCheckPort": "traffic-port",  
      "HealthCheckEnabled": true,  
      "HealthCheckIntervalSeconds": 30,  
      "HealthCheckTimeoutSeconds": 5,  
      "HealthyThresholdCount": 5,  
    }  
  ]  
}
```

```

    "UnhealthyThresholdCount": 2,
    "HealthCheckPath": "/",
    "Matcher": {
      "HttpCode": "200"
    },
    "LoadBalancerArns": [
      "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/my-load-balancer/50dc6c495c0c9188"
    ],
    "TargetType": "instance",
    "ProtocolVersion": "HTTP1",
    "IpAddressType": "ipv4"
  }
]
}

```

Contoh 2: Untuk mendeskripsikan semua kelompok target untuk penyeimbang beban

`describe-target-groups` Contoh berikut menampilkan rincian untuk semua kelompok target untuk penyeimbang beban tertentu. Contoh menggunakan `--query` parameter untuk menampilkan hanya nama grup target.

```

aws elbv2 describe-target-groups \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
  --query TargetGroups[*].TargetGroupName

```

Output:

```

[
  "my-instance-targets",
  "my-ip-targets",
  "my-lambda-target"
]

```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Penyeimbang Beban Aplikasi.

- Untuk detail API, lihat [DescribeTargetGroups](#) di Referensi AWS CLI Perintah.

## describe-target-health

Contoh kode berikut menunjukkan cara menggunakan `describe-target-health`.

## AWS CLI

Contoh 1: Untuk menggambarkan kesehatan target untuk kelompok sasaran

`describe-target-health` Contoh berikut menampilkan rincian kesehatan untuk target kelompok target yang ditentukan. Target ini sehat.

```
aws elbv2 describe-target-health \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Output:

```
{  
  "TargetHealthDescriptions": [  
    {  
      "HealthCheckPort": "80",  
      "Target": {  
        "Id": "i-ceddc4d",  
        "Port": 80  
      },  
      "TargetHealth": {  
        "State": "healthy"  
      }  
    },  
    {  
      "HealthCheckPort": "80",  
      "Target": {  
        "Id": "i-0f76fade",  
        "Port": 80  
      },  
      "TargetHealth": {  
        "State": "healthy"  
      }  
    }  
  ]  
}
```

Contoh 2: Untuk menggambarkan kesehatan target

`describe-target-health` Contoh berikut menampilkan rincian kesehatan untuk target yang ditentukan. Target ini sehat.

```
aws elbv2 describe-target-health \
  --targets Id=i-0f76fade,Port=80 \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

**Output:**

```
{
  "TargetHealthDescriptions": [
    {
      "HealthCheckPort": "80",
      "Target": {
        "Id": "i-0f76fade",
        "Port": 80
      },
      "TargetHealth": {
        "State": "healthy"
      }
    }
  ]
}
```

Contoh output berikut adalah untuk target yang kelompok targetnya tidak ditentukan dalam tindakan untuk pendengar. Target ini tidak dapat menerima lalu lintas dari penyeimbang beban.

```
{
  "TargetHealthDescriptions": [
    {
      "HealthCheckPort": "80",
      "Target": {
        "Id": "i-0f76fade",
        "Port": 80
      },
      "TargetHealth": {
        "State": "unused",
        "Reason": "Target.NotInUse",
        "Description": "Target group is not configured to receive traffic
from the load balancer"
      }
    }
  ]
}
```

Contoh output berikut adalah untuk target yang kelompok targetnya hanya ditentukan dalam tindakan untuk pendengar. Targetnya masih didaftarkan.

```
{
  "TargetHealthDescriptions": [
    {
      "HealthCheckPort": "80",
      "Target": {
        "Id": "i-0f76fade",
        "Port": 80
      },
      "TargetHealth": {
        "State": "initial",
        "Reason": "Elb.RegistrationInProgress",
        "Description": "Target registration is in progress"
      }
    }
  ]
}
```

Contoh output berikut adalah untuk target yang tidak sehat.

```
{
  "TargetHealthDescriptions": [
    {
      "HealthCheckPort": "80",
      "Target": {
        "Id": "i-0f76fade",
        "Port": 80
      },
      "TargetHealth": {
        "State": "unhealthy",
        "Reason": "Target.Timeout",
        "Description": "Connection to target timed out"
      }
    }
  ]
}
```

Contoh output berikut adalah untuk target yang merupakan fungsi Lambda dan pemeriksaan kesehatan dinonaktifkan.

```
{
  "TargetHealthDescriptions": [
    {
      "Target": {
        "Id": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
        "AvailabilityZone": "all",
      },
      "TargetHealth": {
        "State": "unavailable",
        "Reason": "Target.HealthCheckDisabled",
        "Description": "Health checks are not enabled for this target"
      }
    }
  ]
}
```

- Untuk detail API, lihat [DescribeTargetHealth](#) di Referensi AWS CLI Perintah.

## modify-listener

Contoh kode berikut menunjukkan cara menggunakan `modify-listener`.

### AWS CLI

Contoh 1: Untuk mengubah tindakan default menjadi tindakan maju

`modify-listener` Contoh berikut mengubah tindakan default (ke tindakan penerusan) untuk pendengar yang ditentukan.

```
aws elbv2 modify-listener \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f
```

Output:

```
{
  "Listeners": [
    {
      "Protocol": "HTTP",
```



```

    "DefaultActions": [
      {
        "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f",
        "Type": "forward"
      }
    ],
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
    "Port": 80,
    "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
  }
]
}

```

Contoh 2: Untuk mengubah tindakan default menjadi tindakan pengalihan

`modify-listener` Contoh berikut mengubah tindakan default menjadi tindakan pengalihan untuk listener tertentu.

```

aws elbv2 modify-listener \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --default-actions Type=redirect,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f

```

Output:

```

{
  "Listeners": [
    {
      "Protocol": "HTTP",
      "DefaultActions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f",
          "Type": "redirect"
        }
      ],
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
      "Port": 80,

```

```
    "ListenerArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"  
  }  
]  
}
```

### Contoh 3: Untuk mengubah sertifikat server

Contoh ini mengubah sertifikat server untuk listener HTTPS yang ditentukan.

```
aws elbv2 modify-listener \  
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/  
my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65 \  
  --certificates CertificateArn=arn:aws:iam::123456789012:server-certificate/my-  
new-server-cert
```

### Output:

```
{  
  "Listeners": [  
    {  
      "Protocol": "HTTPS",  
      "DefaultActions": [  
        {  
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",  
          "Type": "forward"  
        }  
      ],  
      "SslPolicy": "ELBSecurityPolicy-2015-05",  
      "Certificates": [  
        {  
          "CertificateArn": "arn:aws:iam::123456789012:server-certificate/  
my-new-server-cert"  
        }  
      ],  
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",  
      "Port": 443,  
      "ListenerArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65"  
    }  
  ]  
}
```

```
}
```

- Untuk detail API, lihat [ModifyListener](#) di Referensi AWS CLI Perintah.

## modify-load-balancer-attributes

Contoh kode berikut menunjukkan cara menggunakan `modify-load-balancer-attributes`.

### AWS CLI

Untuk mengaktifkan perlindungan penghapusan

Contoh ini memungkinkan perlindungan penghapusan untuk penyeimbang beban yang ditentukan.

Perintah:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-arn
arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --attributes Key=deletion_protection.enabled,Value=true
```

Output:

```
{
  "Attributes": [
    {
      "Value": "true",
      "Key": "deletion_protection.enabled"
    },
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "60",
      "Key": "idle_timeout.timeout_seconds"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.prefix"
    },
    {
```

```
        "Value": "",
        "Key": "access_logs.s3.bucket"
    }
]
}
```

Untuk mengubah batas waktu idle

Contoh ini mengubah nilai batas waktu idle untuk penyeimbang beban yang ditentukan.

Perintah:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-arn
arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --attributes Key=idle_timeout.timeout_seconds,Value=30
```

Output:

```
{
  "Attributes": [
    {
      "Value": "30",
      "Key": "idle_timeout.timeout_seconds"
    },
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "true",
      "Key": "deletion_protection.enabled"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.bucket"
    }
  ]
}
```

## Untuk mengaktifkan log akses

Contoh ini memungkinkan log akses untuk penyeimbang beban yang ditentukan. Perhatikan bahwa bucket S3 harus berada di wilayah yang sama dengan penyeimbang beban dan harus memiliki kebijakan yang dilampirkan yang memberikan akses ke layanan Elastic Load Balancing.

### Perintah:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-arn
arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --attributes Key=access_logs.s3.enabled,Value=true
Key=access_logs.s3.bucket,Value=my-loadbalancer-logs
Key=access_logs.s3.prefix,Value=myapp
```

### Output:

```
{
  "Attributes": [
    {
      "Value": "true",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "my-load-balancer-logs",
      "Key": "access_logs.s3.bucket"
    },
    {
      "Value": "myapp",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "60",
      "Key": "idle_timeout.timeout_seconds"
    },
    {
      "Value": "false",
      "Key": "deletion_protection.enabled"
    }
  ]
}
```

- Untuk detail API, lihat [ModifyLoadBalancerAttributes](#) di Referensi AWS CLI Perintah.

## modify-rule

Contoh kode berikut menunjukkan cara menggunakan `modify-rule`.

### AWS CLI

Untuk memodifikasi aturan

`modify-rule` Contoh berikut memperbarui tindakan dan kondisi untuk aturan yang ditentukan.

```
aws elbv2 modify-rule \  
  --actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \  
  --conditions Field=path-pattern,Values='/images/*' \  
  --rule-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/app/  
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/9683b2d02a6cabee
```

Output:

```
{  
  "Rules": [  
    {  
      "Priority": "10",  
      "Conditions": [  
        {  
          "Field": "path-pattern",  
          "Values": [  
            "/images/*"  
          ]  
        }  
      ],  
      "RuleArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:listener-rule/app/my-load-balancer/50dc6c495c0c9188/  
f2f7dc8efc522ab2/9683b2d02a6cabee",  
      "IsDefault": false,  
      "Actions": [  
        {  
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",  
          "Type": "forward"  
        }  
      ]  
    }  
  ]  
}
```

```
]
}
```

- Untuk detail API, lihat [ModifyRule](#) di Referensi AWS CLI Perintah.

## modify-target-group-attributes

Contoh kode berikut menunjukkan cara menggunakan `modify-target-group-attributes`.

### AWS CLI

Untuk mengubah batas waktu tunda deregistrasi

Contoh ini menetapkan batas waktu tunda deregistrasi ke nilai yang ditentukan untuk kelompok target yang ditentukan.

Perintah:

```
aws elbv2 modify-target-group-attributes --target-group-arn
arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067 --attributes
Key=deregistration_delay.timeout_seconds,Value=600
```

Output:

```
{
  "Attributes": [
    {
      "Value": "false",
      "Key": "stickiness.enabled"
    },
    {
      "Value": "600",
      "Key": "deregistration_delay.timeout_seconds"
    },
    {
      "Value": "lb_cookie",
      "Key": "stickiness.type"
    },
    {
      "Value": "86400",
      "Key": "stickiness.lb_cookie.duration_seconds"
    }
  ]
}
```

```

    }
  ]
}

```

- Untuk detail API, lihat [ModifyTargetGroupAttributes](#) di Referensi AWS CLI Perintah.

## modify-target-group

Contoh kode berikut menunjukkan cara menggunakan `modify-target-group`.

### AWS CLI

Untuk memodifikasi konfigurasi pemeriksaan kesehatan untuk grup target

`modify-target-group` Contoh berikut mengubah konfigurasi pemeriksaan kesehatan yang digunakan untuk mengevaluasi kesehatan target untuk kelompok sasaran yang ditentukan. Perhatikan bahwa karena cara CLI mem-parsing koma, Anda harus mengelilingi rentang `--matcher` opsi dengan tanda kutip tunggal alih-alih tanda kutip ganda.

```

aws elbv2 modify-target-group \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-https-targets/2453ed029918f21f \
  --health-check-protocol HTTPS \
  --health-check-port 443 \
  --matcher HttpCode='200,299'

```

### Output:

```

{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-https-targets/2453ed029918f21f",
      "TargetGroupName": "my-https-targets",
      "Protocol": "HTTPS",
      "Port": 443,
      "VpcId": "vpc-3ac0fb5f",
      "HealthCheckProtocol": "HTTPS",
      "HealthCheckPort": "443",
      "HealthCheckEnabled": true,
      "HealthCheckIntervalSeconds": 30,
      "HealthCheckTimeoutSeconds": 5,
    }
  ]
}

```



```

    "HealthyThresholdCount": 5,
    "UnhealthyThresholdCount": 2,
    "Matcher": {
      "HttpCode": "200,299"
    },
    "LoadBalancerArns": [
      "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/my-load-balancer/50dc6c495c0c9188"
    ],
    "TargetType": "instance",
    "ProtocolVersion": "HTTP1",
    "IpAddressType": "ipv4"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Penyeimbang Beban Aplikasi.

- Untuk detail API, lihat [ModifyTargetGroup](#) di Referensi AWS CLI Perintah.

## register-targets

Contoh kode berikut menunjukkan cara menggunakan `register-targets`.

### AWS CLI

Contoh 1: Untuk mendaftarkan target dengan grup target dengan ID instance

`register-targets` Contoh berikut mendaftarkan instance tertentu dengan kelompok target. Kelompok sasaran harus memiliki jenis target `instance`.

```

aws elbv2 register-targets \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \
  --targets Id=i-1234567890abcdef0 Id=i-0abcdef1234567890

```

Contoh 2: Untuk mendaftarkan target dengan grup target menggunakan penggantian port

`register-targets` Contoh berikut mendaftarkan instance tertentu dengan kelompok target menggunakan beberapa port. Ini memungkinkan Anda untuk mendaftarkan kontainer pada instance yang sama dengan target dalam grup target.

```

aws elbv2 register-targets \

```

```
--target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-internal-targets/3bb63f11dfb0faf9 \
--targets Id=i-0598c7d356eba48d7,Port=80 Id=i-0598c7d356eba48d7,Port=766
```

Contoh 3: Untuk mendaftarkan target dengan grup target berdasarkan alamat IP

`register-targets` Contoh berikut mendaftarkan alamat IP yang ditentukan dengan kelompok target. Kelompok sasaran harus memiliki jenis target `ip`.

```
aws elbv2 register-targets \
--target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-ip-targets/8518e899d173178f \
--targets Id=10.0.1.15 Id=10.0.1.23
```

Contoh 4: Untuk mendaftarkan fungsi Lambda sebagai target

`register-targets` Contoh berikut mendaftarkan alamat IP yang ditentukan dengan kelompok target. Kelompok sasaran harus memiliki jenis target `lambda`. Anda harus memberikan izin Elastic Load Balancing untuk menjalankan fungsi Lambda.

```
aws elbv2 register-targets \
--target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-ip-targets/8518e899d173178f \
--targets Id=arn:aws:lambda:us-west-2:123456789012:function:my-function
```

- Untuk detail API, lihat [RegisterTargets](#) di Referensi AWS CLI Perintah.

## **remove-listener-certificates**

Contoh kode berikut menunjukkan cara menggunakan `remove-listener-certificates`.

### AWS CLI

Untuk menghapus sertifikat dari pendengar yang aman

Contoh ini menghapus sertifikat yang ditentukan dari pendengar aman yang ditentukan.

Perintah:

```
aws elbv2 remove-listener-certificates --listener-arn
arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 --certificates
```

```
CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-  
f4a3-4072-80be-05b9ba72f705
```

- Untuk detail API, lihat [RemoveListenerCertificates](#) di Referensi AWS CLI Perintah.

## remove-tags

Contoh kode berikut menunjukkan cara menggunakan `remove-tags`.

### AWS CLI

Untuk menghapus tag dari penyeimbang beban

`remove-tags` Contoh berikut menghapus `project` dan `department` tag dari penyeimbang beban yang ditentukan.

```
aws elbv2 remove-tags \  
  --resource-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \  
  --tag-keys project department
```

- Untuk detail API, lihat [RemoveTags](#) di Referensi AWS CLI Perintah.

## set-ip-address-type

Contoh kode berikut menunjukkan cara menggunakan `set-ip-address-type`.

### AWS CLI

Untuk mengatur jenis alamat penyeimbang beban

Contoh ini menetapkan jenis alamat penyeimbang beban yang ditentukan. `dualstack` Subnet penyeimbang beban harus memiliki blok CIDR IPv6 yang terkait.

Perintah:

```
aws elbv2 set-ip-address-type --load-balancer-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --ip-address-  
type dualstack
```

Output:

```
{
  "IpAddressType": "dualstack"
}
```

- Untuk detail API, lihat [SetIpAddressType](#) di Referensi AWS CLI Perintah.

## set-rule-priorities

Contoh kode berikut menunjukkan cara menggunakan `set-rule-priorities`.

### AWS CLI

Untuk menetapkan prioritas aturan

Contoh ini menetapkan prioritas aturan yang ditentukan.

Perintah:

```
aws elbv2 set-rule-priorities --rule-priorities
RuleArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/app/my-
load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3,Priority=5
```

Output:

```
{
  "Rules": [
    {
      "Priority": "5",
      "Conditions": [
        {
          "Field": "path-pattern",
          "Values": [
            "/img/*"
          ]
        }
      ],
      "RuleArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-
rule/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3",
      "IsDefault": false,
      "Actions": [
        {
```

```

        "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
        "Type": "forward"
    }
]
}
}

```

- Untuk detail API, lihat [SetRulePriorities](#) di Referensi AWS CLI Perintah.

## set-security-groups

Contoh kode berikut menunjukkan cara menggunakan `set-security-groups`.

### AWS CLI

Untuk mengaitkan grup keamanan dengan penyeimbang beban

Contoh ini mengaitkan grup keamanan yang ditentukan dengan penyeimbang beban yang ditentukan.

Perintah:

```

aws elbv2 set-security-groups --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --security-
groups sg-5943793c

```

Output:

```

{
  "SecurityGroupIds": [
    "sg-5943793c"
  ]
}

```

- Untuk detail API, lihat [SetSecurityGroups](#) di Referensi AWS CLI Perintah.

## set-subnets

Contoh kode berikut menunjukkan cara menggunakan `set-subnets`.

## AWS CLI

Untuk mengaktifkan Availability Zones untuk load balancer

Contoh ini memungkinkan Availability Zone untuk subnet yang ditentukan untuk load balancer yang ditentukan.

Perintah:

```
aws elbv2 set-subnets --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --subnets subnet-8360a9e7 subnet-b7d581c0
```

Output:

```
{
  "AvailabilityZones": [
    {
      "SubnetId": "subnet-8360a9e7",
      "ZoneName": "us-west-2a"
    },
    {
      "SubnetId": "subnet-b7d581c0",
      "ZoneName": "us-west-2b"
    }
  ]
}
```

- Untuk detail API, lihat [SetSubnets](#) di Referensi AWS CLI Perintah.

## Contoh Elastic Transcoder menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Elastic Transcoder.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **cancel-job**

Contoh kode berikut menunjukkan cara menggunakan `cancel-job`.

AWS CLI

Untuk membatalkan pekerjaan untuk ElasticTranscoder

Ini membatalkan pekerjaan yang ditentukan untuk ElasticTranscoder.

Perintah:

```
aws elastictranscoder cancel-job --id 333333333333-abcde3
```

- Untuk detail API, lihat [CancelJob](#) di Referensi AWS CLI Perintah.

## **create-job**

Contoh kode berikut menunjukkan cara menggunakan `create-job`.

AWS CLI

Untuk membuat pekerjaan untuk ElasticTranscoder

`create-job` Contoh berikut menciptakan pekerjaan untuk ElasticTranscoder.

```
aws elastictranscoder create-job \  
  --pipeline-id 111111111111-abcde1 \  
  --inputs file://inputs.json \  
  --outputs file://outputs.json \  
  --output-key-prefix "recipes/" \  
  --user-metadata file://user-metadata.json
```

### Isi dari `inputs.json`:

```
[{
  "Key": "ETS_example_file.mp4",
  "FrameRate": "auto",
  "Resolution": "auto",
  "AspectRatio": "auto",
  "Interlaced": "auto",
  "Container": "mp4"
}]
```

### Isi `outputs.json`:

```
[
  {
    "Key": "webm/ETS_example_file-kindlefirehd.webm",
    "Rotate": "0",
    "PresetId": "1351620000001-100250"
  }
]
```

### Isi dari `user-metadata.json`:

```
{
  "Food type": "Italian",
  "Cook book": "recipe notebook"
}
```

### Output:

```
{
  "Job": {
    "Status": "Submitted",
    "Inputs": [
      {
        "Container": "mp4",
        "FrameRate": "auto",
        "Key": "ETS_example_file.mp4",
        "AspectRatio": "auto",
        "Resolution": "auto",
        "Interlaced": "auto"
      }
    ]
  }
}
```



```

    ],
    "Playlists": [],
    "Outputs": [
      {
        "Status": "Submitted",
        "Rotate": "0",
        "PresetId": "1351620000001-100250",
        "Watermarks": [],
        "Key": "webm/ETS_example_file-kindlefirehd.webm",
        "Id": "1"
      }
    ],
    "PipelineId": "3333333333333-abcde3",
    "OutputKeyPrefix": "recipes/",
    "UserMetadata": {
      "Cook book": "recipe notebook",
      "Food type": "Italian"
    },
    "Output": {
      "Status": "Submitted",
      "Rotate": "0",
      "PresetId": "1351620000001-100250",
      "Watermarks": [],
      "Key": "webm/ETS_example_file-kindlefirehd.webm",
      "Id": "1"
    },
    "Timing": {
      "SubmitTimeMillis": 1533838012298
    },
    "Input": {
      "Container": "mp4",
      "FrameRate": "auto",
      "Key": "ETS_example_file.mp4",
      "AspectRatio": "auto",
      "Resolution": "auto",
      "Interlaced": "auto"
    },
    "Id": "1533838012294-example",
    "Arn": "arn:aws:elastictranscoder:us-west-2:123456789012:job/1533838012294-
example"
  }
}

```

- Untuk detail API, lihat [CreateJob](#) di Referensi AWS CLI Perintah.

## create-pipeline

Contoh kode berikut menunjukkan cara menggunakan `create-pipeline`.

### AWS CLI

Untuk membuat pipa untuk ElasticTranscoder

`create-pipeline` Contoh berikut membuat pipeline untuk ElasticTranscoder.

```
aws elastictranscoder create-pipeline \  
  --name Default \  
  --input-bucket salesoffice.example.com-source \  
  --role arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role \  
  --notifications Progressing="",Completed="",Warning="",Error=arn:aws:sns:us-  
east-1:111222333444:ETS_Errors \  
  --content-config file://content-config.json \  
  --thumbnail-config file://thumbnail-config.json
```

Isi dari `content-config.json`:

```
{  
  "Bucket": "salesoffice.example.com-public-promos",  
  "Permissions": [  
    {  
      "GranteeType": "Email",  
      "Grantee": "marketing-promos@example.com",  
      "Access": [  
        "FullControl"  
      ]  
    }  
  ],  
  "StorageClass": "Standard"  
}
```

Isi dari `thumbnail-config.json`:

```
{  
  "Bucket": "salesoffice.example.com-public-promos-thumbnails",  
  "Permissions": [  
    {  
      "GranteeType": "Email",
```

```
        "Grantee": "marketing-promos@example.com",
        "Access": [
            "FullControl"
        ]
    },
],
"StorageClass": "ReducedRedundancy"
}
```

## Output:

```
{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "salesoffice.example.com-public-promos",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    },
    "Name": "Default",
    "ThumbnailConfig": {
      "Bucket": "salesoffice.example.com-public-promos-thumbnails",
      "StorageClass": "ReducedRedundancy",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    },
    "Notifications": {
      "Completed": "",

```

```

        "Warning": "",
        "Progressing": "",
        "Error": "arn:aws:sns:us-east-1:123456789012:ETS_Errors"
    },
    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "salesoffice.example.com-source",
    "Id": "1533765810590-example",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/1533765810590-example"
    },
    "Warnings": [
        {
            "Message": "The SNS notification topic for Error events and the pipeline
are in different regions, which increases processing time for jobs in the pipeline
and can incur additional charges. To decrease processing time and prevent cross-
regional charges, use the same region for the SNS notification topic and the
pipeline.",
            "Code": "6006"
        }
    ]
}

```

- Untuk detail API, lihat [CreatePipeline](#) di Referensi AWS CLI Perintah.

## create-preset

Contoh kode berikut menunjukkan cara menggunakan `create-preset`.

### AWS CLI

Untuk membuat preset untuk ElasticTranscoder

`create-preset` Contoh berikut membuat preset untuk ElasticTranscoder.

```

aws elastictranscoder create-preset \
  --name DefaultPreset \
  --description "Use for published videos" \
  --container mp4 \
  --video file://video.json \
  --audio file://audio.json \
  --thumbnails file://thumbnails.json

```

Isi dari `video.json`:

```
{
  "Codec": "H.264",
  "CodecOptions": {
    "Profile": "main",
    "Level": "2.2",
    "MaxReferenceFrames": "3",
    "MaxBitRate": "",
    "BufferSize": "",
    "InterlacedMode": "Progressive",
    "ColorSpaceConversionMode": "None"
  },
  "KeyframesMaxDist": "240",
  "FixedGOP": "false",
  "BitRate": "1600",
  "FrameRate": "auto",
  "MaxFrameRate": "30",
  "MaxWidth": "auto",
  "MaxHeight": "auto",
  "SizingPolicy": "Fit",
  "PaddingPolicy": "Pad",
  "DisplayAspectRatio": "auto",
  "Watermarks": [
    {
      "Id": "company logo",
      "MaxWidth": "20%",
      "MaxHeight": "20%",
      "SizingPolicy": "ShrinkToFit",
      "HorizontalAlign": "Right",
      "HorizontalOffset": "10px",
      "VerticalAlign": "Bottom",
      "VerticalOffset": "10px",
      "Opacity": "55.5",
      "Target": "Content"
    }
  ]
}
```

Isi dari `audio.json`:

```
{
  "Codec": "AAC",
  "CodecOptions": {
    "Profile": "AAC-LC"
  }
}
```

```
},  
  "SampleRate": "44100",  
  "BitRate": "96",  
  "Channels": "2"  
}
```

Isi dari `thumbnails.json`:

```
{  
  "Format": "png",  
  "Interval": "120",  
  "MaxWidth": "auto",  
  "MaxHeight": "auto",  
  "SizingPolicy": "Fit",  
  "PaddingPolicy": "Pad"  
}
```

Output:

```
{  
  "Preset": {  
    "Thumbnails": {  
      "SizingPolicy": "Fit",  
      "MaxWidth": "auto",  
      "Format": "png",  
      "PaddingPolicy": "Pad",  
      "Interval": "120",  
      "MaxHeight": "auto"  
    },  
    "Container": "mp4",  
    "Description": "Use for published videos",  
    "Video": {  
      "SizingPolicy": "Fit",  
      "MaxWidth": "auto",  
      "PaddingPolicy": "Pad",  
      "MaxFrameRate": "30",  
      "FrameRate": "auto",  
      "MaxHeight": "auto",  
      "KeyframesMaxDist": "240",  
      "FixedGOP": "false",  
      "Codec": "H.264",  
      "Watermarks": [  
        {  
          "SizingPolicy": "Fit",  
          "MaxWidth": "auto",  
          "MaxHeight": "auto",  
          "KeyframesMaxDist": "240",  
          "FixedGOP": "false",  
          "Codec": "H.264",  
          "Watermarks": [  
            {  
              "SizingPolicy": "Fit",  
              "MaxWidth": "auto",  
              "MaxHeight": "auto",  
              "KeyframesMaxDist": "240",  
              "FixedGOP": "false",  
              "Codec": "H.264",  
              "Watermarks": [  
                {  
                  "SizingPolicy": "Fit",  
                  "MaxWidth": "auto",  
                  "MaxHeight": "auto",  
                  "KeyframesMaxDist": "240",  
                  "FixedGOP": "false",  
                  "Codec": "H.264",  
                  "Watermarks": [  
                    {  
                      "SizingPolicy": "Fit",  
                      "MaxWidth": "auto",  
                      "MaxHeight": "auto",  
                      "KeyframesMaxDist": "240",  
                      "FixedGOP": "false",  
                      "Codec": "H.264",  
                      "Watermarks": [  
                        {  
                          "SizingPolicy": "Fit",  
                          "MaxWidth": "auto",  
                          "MaxHeight": "auto",  
                          "KeyframesMaxDist": "240",  
                          "FixedGOP": "false",  
                          "Codec": "H.264",  
                          "Watermarks": [  
                            {  
                              "SizingPolicy": "Fit",  
                              "MaxWidth": "auto",  
                              "MaxHeight": "auto",  
                              "KeyframesMaxDist": "240",  
                              "FixedGOP": "false",  
                              "Codec": "H.264",  
                              "Watermarks": [  
                                {  
                                  "SizingPolicy": "Fit",  
                                  "MaxWidth": "auto",  
                                  "MaxHeight": "auto",  
                                  "KeyframesMaxDist": "240",  
                                  "FixedGOP": "false",  
                                  "Codec": "H.264",  
                                  "Watermarks": [  
                                    {  
                                      "SizingPolicy": "Fit",  
                                      "MaxWidth": "auto",  
                                      "MaxHeight": "auto",  
                                      "KeyframesMaxDist": "240",  
                                      "FixedGOP": "false",  
                                      "Codec": "H.264",  
                                      "Watermarks": [  
                                        {  
                                          "SizingPolicy": "Fit",  
                                          "MaxWidth": "auto",  
                                          "MaxHeight": "auto",  
                                          "KeyframesMaxDist": "240",  
                                          "FixedGOP": "false",  
                                          "Codec": "H.264",  
                                          "Watermarks": [  
                                            {  
                                              "SizingPolicy": "Fit",  
                                              "MaxWidth": "auto",  
                                              "MaxHeight": "auto",  
                                              "KeyframesMaxDist": "240",  
                                              "FixedGOP": "false",  
                                              "Codec": "H.264",  
                                              "Watermarks": [  
                                                {  
                                                  "SizingPolicy": "Fit",  
                                                  "MaxWidth": "auto",  
                                                  "MaxHeight": "auto",  
                                                  "KeyframesMaxDist": "240",  
                                                  "FixedGOP": "false",  
                                                  "Codec": "H.264",  
                                                  "Watermarks": [  
                                                    {  
                                                      "SizingPolicy": "Fit",  
                                                      "MaxWidth": "auto",  
                                                      "MaxHeight": "auto",  
                                                      "KeyframesMaxDist": "240",  
                                                      "FixedGOP": "false",  
                                                      "Codec": "H.264",  
                                                      "Watermarks": [  
                                                        {  
                                                          "SizingPolicy": "Fit",  
                                                          "MaxWidth": "auto",  
                                                          "MaxHeight": "auto",  
                                                          "KeyframesMaxDist": "240",  
                                                          "FixedGOP": "false",  
                                                          "Codec": "H.264",  
                                                          "Watermarks": [  
                                                            {  
                                                              "SizingPolicy": "Fit",  
                                                              "MaxWidth": "auto",  
                                                              "MaxHeight": "auto",  
                                                              "KeyframesMaxDist": "240",  
                                                              "FixedGOP": "false",  
                                                              "Codec": "H.264",  
                                                              "Watermarks": [  
                                                                {  
                                                                  "SizingPolicy": "Fit",  
                                                                  "MaxWidth": "auto",  
                                                                  "MaxHeight": "auto",  
                                                                  "KeyframesMaxDist": "240",  
                                                                  "FixedGOP": "false",  
                                                                  "Codec": "H.264",  
                                                                  "Watermarks": [  
                                                                    {  
                                                                      "SizingPolicy": "Fit",  
                                                                      "MaxWidth": "auto",  
                                                                      "MaxHeight": "auto",  
                                                                      "KeyframesMaxDist": "240",  
                                                                      "FixedGOP": "false",  
                                                                      "Codec": "H.264",  
                                                                      "Watermarks": [  
                                                                      ]  
                                                                    ]  
                                                                  ]  
                                                                ]  
                                                              ]  
                                                            ]  
                                                          ]  
                                                        ]  
                                                      ]  
                                                    ]  
                                                  ]  
                                                ]  
                                              ]  
                                            ]  
                                          ]  
                                        ]  
                                      ]  
                                    ]  
                                  ]  
                                ]  
                              ]  
                            ]  
                          ]  
                        ]  
                      ]  
                    ]  
                  ]  
                ]  
              ]  
            ]  
          ]  
        ]  
      ]  
    ]  
  ]  
}
```

```
        "SizingPolicy": "ShrinkToFit",
        "VerticalOffset": "10px",
        "VerticalAlign": "Bottom",
        "Target": "Content",
        "MaxWidth": "20%",
        "MaxHeight": "20%",
        "HorizontalAlign": "Right",
        "HorizontalOffset": "10px",
        "Opacity": "55.5",
        "Id": "company logo"
    }
],
"CodecOptions": {
    "Profile": "main",
    "MaxBitRate": "32",
    "InterlacedMode": "Progressive",
    "Level": "2.2",
    "ColorSpaceConversionMode": "None",
    "MaxReferenceFrames": "3",
    "BufferSize": "5"
},
"BitRate": "1600",
"DisplayAspectRatio": "auto"
},
"Audio": {
    "Channels": "2",
    "CodecOptions": {
        "Profile": "AAC-LC"
    },
    "SampleRate": "44100",
    "Codec": "AAC",
    "BitRate": "96"
},
"Type": "Custom",
"Id": "1533765290724-example"
"Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/1533765290724-example",
"Name": "DefaultPreset"
},
"Warning": ""
}
```

- Untuk detail API, lihat [CreatePreset](#) di Referensi AWS CLI Perintah.

## delete-pipeline

Contoh kode berikut menunjukkan cara menggunakan `delete-pipeline`.

### AWS CLI

Untuk menghapus ElasticTranscoder pipeline yang ditentukan

Ini menghapus ElasticTranscoder pipeline yang ditentukan.

Perintah:

```
aws elastictranscoder delete-pipeline --id 111111111111-abcde1
```

Output:

```
{
  "Success": "true"
}
```

- Untuk detail API, lihat [DeletePipeline](#) di Referensi AWS CLI Perintah.

## delete-preset

Contoh kode berikut menunjukkan cara menggunakan `delete-preset`.

### AWS CLI

Untuk menghapus preset yang ElasticTranscoder ditentukan

Ini menghapus preset yang ElasticTranscoder ditentukan.

Perintah:

```
aws elastictranscoder delete-preset --id 555555555555-abcde5
```

- Untuk detail API, lihat [DeletePreset](#) di Referensi AWS CLI Perintah.

## list-jobs-by-pipeline

Contoh kode berikut menunjukkan cara menggunakan `list-jobs-by-pipeline`.



## AWS CLI

Untuk mengambil daftar ElasticTranscoder pekerjaan di pipeline yang ditentukan

Contoh ini mengambil daftar ElasticTranscoder pekerjaan di pipeline yang ditentukan.

Perintah:

```
aws elastictranscoder list-jobs-by-pipeline --pipeline-id 1111111111111-abcde1
```

Output:

```
{
  "Jobs": []
}
```

- Untuk detail API, lihat [ListJobsByPipeline](#) di Referensi AWS CLI Perintah.

## list-jobs-by-status

Contoh kode berikut menunjukkan cara menggunakan `list-jobs-by-status`.

### AWS CLI

Untuk mengambil daftar ElasticTranscoder pekerjaan dengan status Lengkap

Contoh ini mengambil daftar ElasticTranscoder pekerjaan dengan status Lengkap.

Perintah:

```
aws elastictranscoder list-jobs-by-status --status Complete
```

Output:

```
{
  "Jobs": []
}
```

- Untuk detail API, lihat [ListJobsByStatus](#) di Referensi AWS CLI Perintah.

## list-pipelines

Contoh kode berikut menunjukkan cara menggunakan `list-pipelines`.

### AWS CLI

Untuk mengambil daftar saluran pipa ElasticTranscoder

Contoh ini mengambil daftar ElasticTranscoder pipeline.

Perintah:

```
aws elastictranscoder list-pipelines
```

Output:

```
{
  "Pipelines": [
    {
      "Status": "Active",
      "ContentConfig": {
        "Bucket": "ets-example",
        "Permissions": []
      },
      "Name": "example-pipeline",
      "ThumbnailConfig": {
        "Bucket": "ets-example",
        "Permissions": []
      },
      "Notifications": {
        "Completed": "arn:aws:sns:us-west-2:123456789012:ets_example",
        "Warning": "",
        "Progressing": "",
        "Error": ""
      },
      "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
      "InputBucket": "ets-example",
      "OutputBucket": "ets-example",
      "Id": "3333333333333-abcde3",
      "Arn": "arn:aws:elastictranscoder:us-west-2:123456789012:pipeline/3333333333333-abcde3"
    },
    {
      "Status": "Paused",
```

```
    "ContentConfig": {
      "Bucket": "ets-example",
      "Permissions": []
    },
    "Name": "example-php-test",
    "ThumbnailConfig": {
      "Bucket": "ets-example",
      "Permissions": []
    },
    "Notifications": {
      "Completed": "",
      "Warning": "",
      "Progressing": "",
      "Error": ""
    },
    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "ets-example",
    "OutputBucket": "ets-example",
    "Id": "3333333333333-abcde2",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/3333333333333-abcde2"
  },
  {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "ets-west-output",
      "Permissions": []
    },
    "Name": "pipeline-west",
    "ThumbnailConfig": {
      "Bucket": "ets-west-output",
      "Permissions": []
    },
    "Notifications": {
      "Completed": "arn:aws:sns:us-west-2:123456789012:ets-notifications",
      "Warning": "",
      "Progressing": "",
      "Error": ""
    },
    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "ets-west-input",
    "OutputBucket": "ets-west-output",
    "Id": "3333333333333-abcde1",
```

```
        "Arn": "arn:aws:elastictranscoder:us-  
west-2:123456789012:pipeline/333333333333-abcde1"  
      }  
    ]  
  }  
}
```

- Untuk detail API, lihat [ListPipelines](#) di Referensi AWS CLI Perintah.

## list-presets

Contoh kode berikut menunjukkan cara menggunakan `list-presets`.

### AWS CLI

Untuk mengambil daftar preset ElasticTranscoder

Contoh ini mengambil daftar ElasticTranscoder preset.

Perintah:

```
aws elastictranscoder list-presets --max-items 2
```

Output:

```
{  
  "Presets": [  
    {  
      "Container": "mp4",  
      "Name": "KindleFireHD-preset",  
      "Video": {  
        "Resolution": "1280x720",  
        "FrameRate": "30",  
        "KeyframesMaxDist": "90",  
        "FixedGOP": "false",  
        "Codec": "H.264",  
        "Watermarks": [],  
        "CodecOptions": {  
          "Profile": "main",  
          "MaxReferenceFrames": "3",  
          "ColorSpaceConversionMode": "None",  
          "InterlacedMode": "Progressive",  
          "Level": "4"  
        }  
      }  
    },  
  ],  
}
```

```
    "AspectRatio": "16:9",
    "BitRate": "2200"
  },
  "Audio": {
    "Channels": "2",
    "CodecOptions": {
      "Profile": "AAC-LC"
    },
    "SampleRate": "48000",
    "Codec": "AAC",
    "BitRate": "160"
  },
  "Type": "Custom",
  "Id": "333333333333-abcde2",
  "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/333333333333-abcde2",
  "Thumbnails": {
    "AspectRatio": "16:9",
    "Interval": "60",
    "Resolution": "192x108",
    "Format": "png"
  }
},
{
  "Thumbnails": {
    "AspectRatio": "16:9",
    "Interval": "60",
    "Resolution": "192x108",
    "Format": "png"
  },
  "Container": "mp4",
  "Description": "Custom preset for transcoding jobs",
  "Video": {
    "Resolution": "1280x720",
    "FrameRate": "30",
    "KeyframesMaxDist": "90",
    "FixedGOP": "false",
    "Codec": "H.264",
    "Watermarks": [],
    "CodecOptions": {
      "Profile": "main",
      "MaxReferenceFrames": "3",
      "ColorSpaceConversionMode": "None",
      "InterlacedMode": "Progressive",
```

```

        "Level": "3.1"
      },
      "AspectRatio": "16:9",
      "BitRate": "2200"
    },
    "Audio": {
      "Channels": "2",
      "CodecOptions": {
        "Profile": "AAC-LC"
      },
      "SampleRate": "44100",
      "Codec": "AAC",
      "BitRate": "160"
    },
    "Type": "Custom",
    "Id": "333333333333-abcde3",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/333333333333-abcde3",
    "Name": "Roman's Preset"
  }
],
"NextToken": "eyJQYWdlVG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

- Untuk detail API, lihat [ListPresets](#) di Referensi AWS CLI Perintah.

## read-job

Contoh kode berikut menunjukkan cara menggunakan `read-job`.

### AWS CLI

Untuk mengambil pekerjaan ElasticTranscoder

Contoh ini mengambil ElasticTranscoder pekerjaan yang ditentukan.

Perintah:

```
aws elastictranscoder read-job --id 1533838012294-example
```

Output:

```
{
```

```
"Job": {
  "Status": "Progressing",
  "Inputs": [
    {
      "Container": "mp4",
      "FrameRate": "auto",
      "Key": "ETS_example_file.mp4",
      "AspectRatio": "auto",
      "Resolution": "auto",
      "Interlaced": "auto"
    }
  ],
  "Playlists": [],
  "Outputs": [
    {
      "Status": "Progressing",
      "Rotate": "0",
      "PresetId": "1351620000001-100250",
      "Watermarks": [],
      "Key": "webm/ETS_example_file-kindlefirehd.webm",
      "Id": "1"
    }
  ],
  "PipelineId": "333333333333-abcde3",
  "OutputKeyPrefix": "recipes/",
  "UserMetadata": {
    "Cook book": "recipe notebook",
    "Food type": "Italian"
  },
  "Output": {
    "Status": "Progressing",
    "Rotate": "0",
    "PresetId": "1351620000001-100250",
    "Watermarks": [],
    "Key": "webm/ETS_example_file-kindlefirehd.webm",
    "Id": "1"
  },
  "Timing": {
    "SubmitTimeMillis": 1533838012298,
    "StartTimeMillis": 1533838013786
  },
  "Input": {
    "Container": "mp4",
    "FrameRate": "auto",
```

```
        "Key": "ETS_example_file.mp4",
        "AspectRatio": "auto",
        "Resolution": "auto",
        "Interlaced": "auto"
    },
    "Id": "1533838012294-example",
    "Arn": "arn:aws:elastictranscoder:us-west-2:123456789012:job/1533838012294-
example"
    }
}
```

- Untuk detail API, lihat [ReadJob](#) di Referensi AWS CLI Perintah.

## read-pipeline

Contoh kode berikut menunjukkan cara menggunakan `read-pipeline`.

### AWS CLI

Untuk mengambil pipa ElasticTranscoder

Contoh ini mengambil ElasticTranscoder pipeline yang ditentukan.

Perintah:

```
aws elastictranscoder read-pipeline --id 333333333333-abcde3
```

Output:

```
{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "ets-example",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
        }
      ]
    }
  }
}
```



```

        "GranteeType": "Email"
      }
    ]
  },
  "Name": "Default",
  "ThumbnailConfig": {
    "Bucket": "ets-example",
    "StorageClass": "ReducedRedundancy",
    "Permissions": [
      {
        "Access": [
          "FullControl"
        ],
        "Grantee": "marketing-promos@example.com",
        "GranteeType": "Email"
      }
    ]
  },
  "Notifications": {
    "Completed": "",
    "Warning": "",
    "Progressing": "",
    "Error": "arn:aws:sns:us-east-1:123456789012:ETS_Errors"
  },
  "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
  "InputBucket": "ets-example",
  "Id": "3333333333333-abcde3",
  "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/3333333333333-abcde3"
},
"Warnings": [
  {
    "Message": "The SNS notification topic for Error events and the pipeline
are in different regions, which increases processing time for jobs in the pipeline
and can incur additional charges. To decrease processing time and prevent cross-
regional charges, use the same region for the SNS notification topic and the
pipeline.",
    "Code": "6006"
  }
]
}

```

- Untuk detail API, lihat [ReadPipeline](#) di Referensi AWS CLI Perintah.

## read-preset

Contoh kode berikut menunjukkan cara menggunakan `read-preset`.

### AWS CLI

Untuk mengambil preset ElasticTranscoder

Contoh ini mengambil preset yang ElasticTranscoder ditentukan.

Perintah:

```
aws elastictranscoder read-preset --id 1351620000001-500020
```

Output:

```
{
  "Preset": {
    "Thumbnails": {
      "SizingPolicy": "ShrinkToFit",
      "MaxWidth": "192",
      "Format": "png",
      "PaddingPolicy": "NoPad",
      "Interval": "300",
      "MaxHeight": "108"
    },
    "Container": "fmp4",
    "Description": "System preset: MPEG-Dash Video - 4.8M",
    "Video": {
      "SizingPolicy": "ShrinkToFit",
      "MaxWidth": "1280",
      "PaddingPolicy": "NoPad",
      "FrameRate": "30",
      "MaxHeight": "720",
      "KeyframesMaxDist": "60",
      "FixedGOP": "true",
      "Codec": "H.264",
      "Watermarks": [
        {
          "SizingPolicy": "ShrinkToFit",
          "VerticalOffset": "10%",
          "VerticalAlign": "Top",
          "Target": "Content",
          "MaxWidth": "10%",
```

```
        "MaxHeight": "10%",
        "HorizontalAlign": "Left",
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "TopLeft"
    },
    {
        "SizingPolicy": "ShrinkToFit",
        "VerticalOffset": "10%",
        "VerticalAlign": "Top",
        "Target": "Content",
        "MaxWidth": "10%",
        "MaxHeight": "10%",
        "HorizontalAlign": "Right",
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "TopRight"
    },
    {
        "SizingPolicy": "ShrinkToFit",
        "VerticalOffset": "10%",
        "VerticalAlign": "Bottom",
        "Target": "Content",
        "MaxWidth": "10%",
        "MaxHeight": "10%",
        "HorizontalAlign": "Left",
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "BottomLeft"
    },
    {
        "SizingPolicy": "ShrinkToFit",
        "VerticalOffset": "10%",
        "VerticalAlign": "Bottom",
        "Target": "Content",
        "MaxWidth": "10%",
        "MaxHeight": "10%",
        "HorizontalAlign": "Right",
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "BottomRight"
    }
],
"CodecOptions": {
```

```

        "Profile": "main",
        "MaxBitRate": "4800",
        "InterlacedMode": "Progressive",
        "Level": "3.1",
        "ColorSpaceConversionMode": "None",
        "MaxReferenceFrames": "3",
        "BufferSize": "9600"
    },
    "BitRate": "4800",
    "DisplayAspectRatio": "auto"
},
"Type": "System",
"Id": "1351620000001-500020",
"Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/1351620000001-500020",
"Name": "System preset: MPEG-Dash Video - 4.8M"
}
}

```

- Untuk detail API, lihat [ReadPreset](#) di Referensi AWS CLI Perintah.

## update-pipeline-notifications

Contoh kode berikut menunjukkan cara menggunakan `update-pipeline-notifications`.

### AWS CLI

Untuk memperbarui notifikasi ElasticTranscoder pipa

Contoh ini memperbarui pemberitahuan dari ElasticTranscoder pipeline yang ditentukan.

Perintah:

```

aws elastictranscoder update-pipeline-notifications --id 111111111111-
abcde1 --notifications Progressing=arn:aws:sns:us-west-2:0123456789012:my-
topic,Completed=arn:aws:sns:us-west-2:0123456789012:my-topic,Warning=arn:aws:sns:us-
west-2:0123456789012:my-topic,Error=arn:aws:sns:us-east-1:111222333444:ETS_Errors

```

Output:

```
{
```

```
"Pipeline": {
  "Status": "Active",
  "ContentConfig": {
    "Bucket": "ets-example",
    "StorageClass": "Standard",
    "Permissions": [
      {
        "Access": [
          "FullControl"
        ],
        "Grantee": "marketing-promos@example.com",
        "GranteeType": "Email"
      }
    ]
  },
  "Name": "Default",
  "ThumbnailConfig": {
    "Bucket": "ets-example",
    "StorageClass": "ReducedRedundancy",
    "Permissions": [
      {
        "Access": [
          "FullControl"
        ],
        "Grantee": "marketing-promos@example.com",
        "GranteeType": "Email"
      }
    ]
  },
  "Notifications": {
    "Completed": "arn:aws:sns:us-west-2:0123456789012:my-topic",
    "Warning": "arn:aws:sns:us-west-2:0123456789012:my-topic",
    "Progressing": "arn:aws:sns:us-west-2:0123456789012:my-topic",
    "Error": "arn:aws:sns:us-east-1:111222333444:ETS_Errors"
  },
  "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
  "InputBucket": "ets-example",
  "Id": "111111111111-abcde1",
  "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/111111111111-abcde1"
}
```

- Untuk detail API, lihat [UpdatePipelineNotifications](#) di Referensi AWS CLI Perintah.

## update-pipeline-status

Contoh kode berikut menunjukkan cara menggunakan `update-pipeline-status`.

### AWS CLI

Untuk memperbarui status ElasticTranscoder pipa

Contoh ini memperbarui status ElasticTranscoder pipeline yang ditentukan.

Perintah:

```
aws elastictranscoder update-pipeline-status --id 111111111111-abcde1 --status Paused
```

Output:

```
{
  "Pipeline": {
    "Status": "Paused",
    "ContentConfig": {
      "Bucket": "ets-example",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    },
    "Name": "Default",
    "ThumbnailConfig": {
      "Bucket": "ets-example",
      "StorageClass": "ReducedRedundancy",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    }
  }
}
```

```

    }
  ]
},
"Notifications": {
  "Completed": "",
  "Warning": "",
  "Progressing": "",
  "Error": "arn:aws:sns:us-east-1:803981987763:ETS_Errors"
},
"Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
"InputBucket": "ets-example",
"Id": "111111111111-abcde1",
"Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/111111111111-abcde1"
}
}

```

- Untuk detail API, lihat [UpdatePipelineStatus](#) di Referensi AWS CLI Perintah.

## update-pipeline

Contoh kode berikut menunjukkan cara menggunakan `update-pipeline`.

### AWS CLI

Untuk memperbarui ElasticTranscoder pipeline

`update-pipeline` Contoh berikut memperbarui ElasticTranscoder pipeline yang ditentukan.

```

aws elastictranscoder update-pipeline \
  --id 111111111111-abcde1
  --name DefaultExample \
  --input-bucket salesoffice.example.com-source \
  --role arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role \
  --notifications Progressing="",Completed="",Warning="",Error=arn:aws:sns:us-
east-1:111222333444:ETS_Errors \
  --content-config file://content-config.json \
  --thumbnail-config file://thumbnail-config.json

```

Isi dari `content-config.json`:

```
{
```

```
"Bucket": "salesoffice.example.com-public-promos",
"Permissions": [
  {
    "GranteeType": "Email",
    "Grantee": "marketing-promos@example.com",
    "Access": [
      "FullControl"
    ]
  }
],
"StorageClass": "Standard"
}
```

Isi dari thumbnail-config.json:

```
{
  "Bucket": "salesoffice.example.com-public-promos-thumbnails",
  "Permissions": [
    {
      "GranteeType": "Email",
      "Grantee": "marketing-promos@example.com",
      "Access": [
        "FullControl"
      ]
    }
  ],
  "StorageClass": "ReducedRedundancy"
}
```

Output:

```
{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "ets-example",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",

```



```

        "GranteeType": "Email"
      }
    ]
  },
  "Name": "DefaultExample",
  "ThumbnailConfig": {
    "Bucket": "ets-example",
    "StorageClass": "ReducedRedundancy",
    "Permissions": [
      {
        "Access": [
          "FullControl"
        ],
        "Grantee": "marketing-promos@example.com",
        "GranteeType": "Email"
      }
    ]
  },
  "Notifications": {
    "Completed": "",
    "Warning": "",
    "Progressing": "",
    "Error": "arn:aws:sns:us-east-1:111222333444:ETS_Errors"
  },
  "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
  "InputBucket": "ets-example",
  "Id": "333333333333-abcde3",
  "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/333333333333-abcde3"
},
"Warnings": [
  {
    "Message": "The SNS notification topic for Error events and the pipeline
are in different regions, which increases processing time for jobs in the pipeline
and can incur additional charges. To decrease processing time and prevent cross-
regional charges, use the same region for the SNS notification topic and the
pipeline.",
    "Code": "6006"
  }
]
}

```

- Untuk detail API, lihat [UpdatePipeline](#) di Referensi AWS CLI Perintah.

## ElastiCache contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with ElastiCache.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags-to-resource**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-resource`.

AWS CLI

Untuk menambahkan tag ke sumber daya

`add-tags-to-resource` Contoh berikut menambahkan hingga 10 tag, pasangan kunci-nilai, ke sumber daya cluster atau snapshot.

```
aws elasticache add-tags-to-resource \  
  --resource-name "arn:aws:elasticache:us-east-1:1234567890:cluster:my-mem-  
cluster" \  
  --tags '{"20150202":15, "ElastiCache":"Service"}
```

Output:

```
{  
  "TagList": [  
    {
```

```

        "Value": "20150202",
        "Key": "APIVersion"
    },
    {
        "Value": "ElastiCache",
        "Key": "Service"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Memantau Biaya dengan Tag Alokasi Biaya](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [AddTagsToResource](#) di Referensi AWS CLI Perintah.

### **authorize-cache-security-group-ingress**

Contoh kode berikut menunjukkan cara menggunakan `authorize-cache-security-group-ingress`.

#### AWS CLI

Untuk mengotorisasi grup keamanan cache untuk masuknya

`authorize-cache-security-group-ingress` Contoh berikut memungkinkan masuknya jaringan ke grup keamanan cache.

```

aws elasticache authorize-cache-security-group-ingress \
  --cache-security-group-name "my-sec-grp" \
  --ec2-security-group-name "my-ec2-sec-grp" \
  --ec2-security-group-owner-id "1234567890"

```

Perintah tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pembaruan Layanan Mandiri di Amazon ElastiCache di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [AuthorizeCacheSecurityGroupIngress](#) di Referensi AWS CLI Perintah.

### **batch-apply-update-action**

Contoh kode berikut menunjukkan cara menggunakan `batch-apply-update-action`.

## AWS CLI

Untuk menerapkan pembaruan layanan

`batch-apply-update-action` Contoh berikut menerapkan pembaruan layanan ke cluster Redis.

```
aws elasticache batch-apply-update-action \  
  --service-update-name elc-xxxxx406-xxx \  
  --replication-group-ids test-cluster
```

Output:

```
{  
  "ProcessedUpdateActions": [  
    {  
      "ReplicationGroupId": "pat-cluster",  
      "ServiceUpdateName": "elc-xxxxx406-xxx",  
      "UpdateActionStatus": "waiting-to-start"  
    }  
  ],  
  "UnprocessedUpdateActions": []  
}
```

Untuk informasi selengkapnya, lihat [Pembaruan Layanan Mandiri di Amazon ElastiCache di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [BatchApplyUpdateAction](#) di Referensi AWS CLI Perintah.

## **batch-stop-update-action**

Contoh kode berikut menunjukkan cara menggunakan `batch-stop-update-action`.

## AWS CLI

Untuk menghentikan pembaruan layanan

`batch-stop-update-action` Contoh berikut menerapkan pembaruan layanan ke cluster Redis.

```
aws elasticache batch-stop-update-action \  
  --service-update-name elc-xxxxx406-xxx \  
  --replication-group-ids test-cluster
```

```
--service-update-name elc-xxxxx406-xxx \  
--replication-group-ids test-cluster
```

Output:

```
{  
  "ProcessedUpdateActions": [  
    {  
      "ReplicationGroupId": "pat-cluster",  
      "ServiceUpdateName": "elc-xxxxx406-xxx",  
      "UpdateActionStatus": "stopping"  
    }  
  ],  
  "UnprocessedUpdateActions": []  
}
```

Untuk informasi selengkapnya, lihat [Pembaruan Layanan Mandiri di Amazon ElastiCache di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [BatchStopUpdateAction](#) di Referensi AWS CLI Perintah.

## copy-snapshot

Contoh kode berikut menunjukkan cara menggunakan `copy-snapshot`.

### AWS CLI

Untuk menyalin snapshot

`copy-snapshot` Contoh berikut membuat salinan snapshot yang ada.

```
aws elasticache copy-snapshot \  
  --source-snapshot-name "my-snapshot" \  
  --target-snapshot-name "my-snapshot-copy"
```

Output:

```
{  
  "Snapshot": {  
    "Engine": "redis",  
    "CacheParameterGroupName": "default.redis3.2",  
    "VpcId": "vpc-3820329f3",  
  }  
}
```

```

    "CacheClusterId": "my-redis4",
    "SnapshotRetentionLimit": 7,
    "NumCacheNodes": 1,
    "SnapshotName": "my-snapshot-copy",
    "CacheClusterCreateTime": "2016-12-21T22:24:04.955Z",
    "AutoMinorVersionUpgrade": true,
    "PreferredAvailabilityZone": "us-east-1c",
    "SnapshotStatus": "creating",
    "SnapshotSource": "manual",
    "SnapshotWindow": "07:00-08:00",
    "EngineVersion": "3.2.4",
    "NodeSnapshots": [
      {
        "CacheSize": "3 MB",
        "SnapshotCreateTime": "2016-12-28T07:00:52Z",
        "CacheNodeId": "0001",
        "CacheNodeCreateTime": "2016-12-21T22:24:04.955Z"
      }
    ],
    "CacheSubnetGroupName": "default",
    "Port": 6379,
    "PreferredMaintenanceWindow": "tue:09:30-tue:10:30",
    "CacheNodeType": "cache.m3.large"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengeksport Cadangan](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [CopySnapshot](#) di Referensi AWS CLI Perintah.

## create-cache-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-cache-cluster`.

### AWS CLI

Untuk membuat cluster cache

`create-cache-cluster` Contoh berikut membuat cluster cache menggunakan mesin Redis.

```

aws elasticache create-cache-cluster \
  --cache-cluster-id "cluster-test" \
  --engine redis \

```

```
--cache-node-type cache.m5.large \  
--num-cache-nodes 1
```

### Output:

```
{  
  "CacheCluster": {  
    "CacheClusterId": "cluster-test",  
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/  
home#client-download:",  
    "CacheNodeType": "cache.m5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.5",  
    "CacheClusterStatus": "creating",  
    "NumCacheNodes": 1,  
    "PreferredMaintenanceWindow": "sat:13:00-sat:14:00",  
    "PendingModifiedValues": {},  
    "CacheSecurityGroups": [],  
    "CacheParameterGroup": {  
      "CacheParameterGroupName": "default.redis5.0",  
      "ParameterApplyStatus": "in-sync",  
      "CacheNodeIdsToReboot": []  
    },  
    "CacheSubnetGroupName": "default",  
    "AutoMinorVersionUpgrade": true,  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:30-07:30",  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Cluster](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [CreateCacheCluster](#) di Referensi AWS CLI Perintah.

## create-cache-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `create-cache-parameter-group`.

### AWS CLI

Untuk membuat grup parameter cache

`create-cache-parameter-group` Contoh berikut membuat grup parameter ElastiCache cache Amazon baru.

```
aws elasticache create-cache-parameter-group \  
  --cache-parameter-group-family "redis5.0" \  
  --cache-parameter-group-name "mygroup" \  
  --description "mygroup"
```

Output:

```
{  
  "CacheParameterGroup": {  
    "CacheParameterGroupName": "mygroup",  
    "CacheParameterGroupFamily": "redis5.0",  
    "Description": "my group"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Parameter](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [CreateCacheParameterGroup](#) di Referensi AWS CLI Perintah.

## `create-cache-subnet-group`

Contoh kode berikut menunjukkan cara menggunakan `create-cache-subnet-group`.

AWS CLI

Untuk membuat grup subnet cache

`create-cache-subnet-group` Contoh berikut membuat grup subnet cache baru.

```
aws elasticache create-cache-subnet-group \  
  --cache-subnet-group-name "mygroup" \  
  --cache-subnet-group-description "my subnet group" \  
  --subnet-ids "subnet-xxxxec4f"
```

Output:

```
{
```



```

"CacheSubnetGroup": {
  "CacheSubnetGroupName": "mygroup",
  "CacheSubnetGroupDescription": "my subnet group",
  "VpcId": "vpc-a3e97cdb",
  "Subnets": [
    {
      "SubnetIdentifier": "subnet-xxxxec4f",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2d"
      }
    }
  ]
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Grup Subnet Cache di Panduan Pengguna ElastiCache](#).

- Untuk detail API, lihat [CreateCacheSubnetGroup](#) di Referensi AWS CLI Perintah.

## create-global-replication-group

Contoh kode berikut menunjukkan cara menggunakan `create-global-replication-group`.

### AWS CLI

Untuk membuat grup replikasi global

`create-global-replication-group` Contoh berikut membuat grup replikasi global baru.

```

aws elasticache create-global-replication-group \
  --global-replication-group-id-suffix my-global-replication-group \
  --primary-replication-group-id my-primary-cluster

```

Output:

```

{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "sgaui-my-global-replication-group",
    "GlobalReplicationGroupDescription": " ",
    "Status": "creating",
    "CacheNodeType": "cache.r5.large",

```

```
"Engine": "redis",
"EngineVersion": "5.0.6",
"Members": [
  {
    "ReplicationGroupId": "my-primary-cluster",
    "ReplicationGroupRegion": "us-west-2",
    "Role": "PRIMARY",
    "AutomaticFailover": "enabled",
    "Status": "associating"
  }
],
"ClusterEnabled": true,
"GlobalNodeGroups": [
  {
    "GlobalNodeId": "sgaui-my-global-replication-group-0001",
    "Slots": "0-16383"
  }
],
"AuthTokenEnabled": false,
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}
```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [CreateGlobalReplicationGroup](#) di Referensi AWS CLI Perintah.

## create-replication-group

Contoh kode berikut menunjukkan cara menggunakan `create-replication-group`.

### AWS CLI

Untuk membuat grup replikasi

`create-replication-group` Contoh berikut membuat Redis (mode cluster dinonaktifkan) atau Redis (mode cluster diaktifkan) grup replikasi. Operasi ini hanya berlaku untuk Redis.

```
aws elasticache create-replication-group \
  --replication-group-id "mygroup" \
```

```
--replication-group-description "my group" \  
--engine "redis" \  
--cache-node-type "cache.m5.large"
```

### Output:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "mygroup",  
    "Description": "my group",  
    "Status": "creating",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "mygroup-001"  
    ],  
    "AutomaticFailover": "disabled",  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:00-07:00",  
    "ClusterEnabled": false,  
    "CacheNodeType": "cache.m5.large",  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Replikasi Redis di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [CreateReplicationGroup](#) di Referensi AWS CLI Perintah.

## create-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-snapshot`.

### AWS CLI

Untuk membuat snapshot

`create-snapshot` Contoh berikut membuat snapshot menggunakan mesin Redis.

```
aws elasticache create-snapshot \  
--snapshot-name mysnapshot \  

```

```
--cache-cluster-id cluster-test
```

Output:

```
{
  "Snapshot": {
    "SnapshotName": "mysnapshot",
    "CacheClusterId": "cluster-test",
    "SnapshotStatus": "creating",
    "SnapshotSource": "manual",
    "CacheNodeType": "cache.m5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2b",
    "CacheClusterCreateTime": "2020-03-19T03:12:01.483Z",
    "PreferredMaintenanceWindow": "sat:13:00-sat:14:00",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0",
    "CacheSubnetGroupName": "default",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "06:30-07:30",
    "NodeSnapshots": [
      {
        "CacheNodeId": "0001",
        "CacheSize": "",
        "CacheNodeCreateTime": "2020-03-19T03:12:01.483Z"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Backup dan Restore for ElastiCache for Redis](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [CreateSnapshot](#) di Referensi AWS CLI Perintah.

## create-user-group

Contoh kode berikut menunjukkan cara menggunakan `create-user-group`.

## AWS CLI

Untuk membuat grup pengguna

`create-user-group` Contoh berikut membuat grup pengguna baru.

```
aws elasticache create-user-group \  
  --user-group-id myusergroup \  
  --engine redis \  
  --user-ids default
```

Output:

```
{  
  "UserGroupId": "myusergroup",  
  "Status": "creating",  
  "Engine": "redis",  
  "UserIds": [  
    "default"  
  ],  
  "ReplicationGroups": [],  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [CreateUserGroup](#) di Referensi AWS CLI Perintah.

## **create-user**

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

## AWS CLI

Untuk membuat pengguna

`create-user` Contoh berikut menciptakan pengguna baru.

```
aws elasticache create-user \  
  --user-id user1 \  
  --user-name myUser \  
  --engine redis
```

```
--passwords mYnuUzrpAxXw2rdzx \  
--engine redis \  
--access-string "on ~app:* -@all +@read"
```

### Output:

```
{  
  "UserId": "user2",  
  "UserName": "myUser",  
  "Status": "active",  
  "Engine": "redis",  
  "AccessString": "on ~app:* -@all +@read +@hash +@bitmap +@geo -setbit -bitfield  
-hset -hsetnx -hmset -hincrby -hincrbyfloat -hdel -bitop -geoadd -georadius -  
georadiusbymember",  
  "UserGroupIds": [],  
  "Authentication": {  
    "Type": "password",  
    "PasswordCount": 1  
  },  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## decrease-node-groups-in-global-replication-group

Contoh kode berikut menunjukkan cara menggunakan `decrease-node-groups-in-global-replication-group`.

### AWS CLI

Untuk mengurangi jumlah grup node dalam grup replikasi global

Berikut ini `decrease-node-groups-in-global-replication-group` mengurangi jumlah kelompok node menggunakan mesin Redis.

```
aws elasticache decrease-node-groups-in-global-replication-group \  
  --global-replication-group-id sgaui-test \  
  --node-group-count 1 \  
  --engine redis
```

```
--apply-immediately \  
--global-node-groups-to-retain sgaui-test-0003
```

**Output:**

```
{  
  "GlobalReplicationGroup":  
  {  
    "GlobalReplicationGroupId": "sgaui-test",  
    "GlobalReplicationGroupDescription": "test",  
    "Status": "modifying",  
    "CacheNodeType": "cache.r5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.6",  
    "Members": [  
      {  
        "ReplicationGroupId": "test-2",  
        "ReplicationGroupRegion": "us-east-1",  
        "Role": "SECONDARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      },  
      {  
        "ReplicationGroupId": "test-1",  
        "ReplicationGroupRegion": "us-west-2",  
        "Role": "PRIMARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      }  
    ],  
    "ClusterEnabled": true,  
    "GlobalNodeGroups": [  
      {  
        "GlobalNodeGroupId": "sgaui-test-0001",  
        "Slots": "0-449,1816-5461"  
      },  
      {  
        "GlobalNodeGroupId": "sgaui-test-0002",  
        "Slots": "6827-10922"  
      },  
      {  
        "GlobalNodeGroupId": "sgaui-test-0003",  
        "Slots": "10923-14052,15418-16383"  
      }  
    ]  
  }  
}
```

```
    },
    {
      "GlobalNodeGroupId": "sgaui-test-0004",
      "Slots": "450-1815,5462-6826,14053-15417"
    }
  ],
  "AuthTokenEnabled": false,
  "TransitEncryptionEnabled": false,
  "AtRestEncryptionEnabled": false
}
}
```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DecreaseNodeGroupsInGlobalReplicationGroup](#) di Referensi AWS CLI Perintah.

## decrease-replica-count

Contoh kode berikut menunjukkan cara menggunakan `decrease-replica-count`.

### AWS CLI

Untuk mengurangi jumlah replika

`decrease-replica-count` Contoh berikut secara dinamis mengurangi jumlah replika dalam grup replikasi Redis (mode cluster dinonaktifkan) atau jumlah node replika dalam satu atau lebih grup node (pecahan) dari grup replikasi Redis (mode cluster diaktifkan). Operasi ini dilakukan tanpa downtime cluster.

```
aws elasticache decrease-replica-count \
  --replication-group-id my-cluster \
  --apply-immediately \
  --new-replica-count 2
```

### Output:

```
{
  "ReplicationGroup": {
    "ReplicationGroupId": "my-cluster",
    "Description": " ",
```



```
"Status": "modifying",
"PendingModifiedValues": {},
"MemberClusters": [
  "myrepliac",
  "my-cluster-001",
  "my-cluster-002",
  "my-cluster-003"
],
"NodeGroups": [
  {
    "NodeGroupId": "0001",
    "Status": "modifying",
    "PrimaryEndpoint": {
      "Address": "my-cluster.xxxxx.ng.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "ReaderEndpoint": {
      "Address": "my-cluster-
ro.xxxxx.ng.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "NodeGroupMembers": [
      {
        "CacheClusterId": "myrepliac",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
          "Address":
"myrepliac.xxxxx.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
      },
      {
        "CacheClusterId": "my-cluster-001",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
          "Address": "my-
cluster-001.xxxxx.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "primary"
      },
    ]
  },

```

```

        {
            "CacheClusterId": "my-cluster-002",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
                "Address": "my-
cluster-002.xxxxx.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2a",
            "CurrentRole": "replica"
        },
        {
            "CacheClusterId": "my-cluster-003",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
                "Address": "my-
cluster-003.xxxxx.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2a",
            "CurrentRole": "replica"
        }
    ]
}
],
"AutomaticFailover": "disabled",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.xlarge",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Untuk informasi selengkapnya, lihat [Mengubah Jumlah Replika](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DecreaseReplicaCount](#) di Referensi AWS CLI Perintah.

## delete-cache-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-cache-cluster`.

## AWS CLI

Untuk menghapus cluster cache

`delete-cache-cluster` Contoh berikut menghapus cluster yang sebelumnya disediakan sebelumnya. Perintah menghapus semua node cache terkait, titik akhir simpul. dan cluster itu sendiri. Ketika Anda menerima respons yang berhasil dari operasi ini, Amazon ElastiCache segera mulai menghapus klaster; Anda tidak dapat membatalkan atau mengembalikan operasi ini.

Operasi ini tidak berlaku untuk hal-hal berikut:

Redis (mode cluster diaktifkan) ClusterSa cluster yang merupakan replika baca terakhir dari grup node GroUpa replikasi (shard) yang memiliki mode multi-AZ EnableDA cluster dari Redis (cluster mode enabled) replikasi GroUpa cluster yang tidak dalam status tersedia

```
aws elasticache delete-cache-cluster \  
  --cache-cluster-id "my-cluster-002"
```

Output:

```
{  
  "CacheCluster": {  
    "CacheClusterId": "my-cluster-002",  
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/  
home#client-download:",  
    "CacheNodeType": "cache.r5.xlarge",  
    "Engine": "redis",  
    "EngineVersion": "5.0.5",  
    "CacheClusterStatus": "deleting",  
    "NumCacheNodes": 1,  
    "PreferredAvailabilityZone": "us-west-2a",  
    "CacheClusterCreateTime": "2019-11-26T03:35:04.546Z",  
    "PreferredMaintenanceWindow": "mon:04:05-mon:05:05",  
    "PendingModifiedValues": {},  
    "NotificationConfiguration": {  
      "TopicArn": "arn:aws:sns:us-west-x:xxxxxxx4152:My_Topic",  
      "TopicStatus": "active"  
    },  
    "CacheSecurityGroups": [],  
    "CacheParameterGroup": {  
      "CacheParameterGroupName": "mygroup",  
      "ParameterApplyStatus": "in-sync",  
    }  
  }  
}
```

```
    "CacheNodeIdsToReboot": []
  },
  "CacheSubnetGroupName": "kxkxk",
  "AutoMinorVersionUpgrade": true,
  "SecurityGroups": [
    {
      "SecurityGroupId": "sg-xxxxxxxxxx9836",
      "Status": "active"
    },
    {
      "SecurityGroupId": "sg-xxxxxxxxxx7b",
      "Status": "active"
    }
  ],
  "ReplicationGroupId": "my-cluster",
  "SnapshotRetentionLimit": 0,
  "SnapshotWindow": "07:30-08:30",
  "TransitEncryptionEnabled": false,
  "AtRestEncryptionEnabled": false
}
}
```

Untuk informasi selengkapnya, lihat [Menghapus Cluster](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DeleteCacheCluster](#) di Referensi AWS CLI Perintah.

## delete-cache-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `delete-cache-parameter-group`.

### AWS CLI

Untuk menghapus grup parameter cache

`delete-cache-parameter-group` Contoh berikut menghapus kelompok parameter cache yang ditentukan. Anda tidak dapat menghapus grup parameter cache jika dikaitkan dengan cluster cache apa pun.

```
aws elasticache delete-cache-parameter-group \
  --cache-parameter-group-name myparamgroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Grup Parameter di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [DeleteCacheParameterGroup](#) di Referensi AWS CLI Perintah.

## **delete-cache-subnet-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-cache-subnet-group`.

### AWS CLI

Untuk menghapus grup subnet cache

`delete-cache-subnet-group` Contoh berikut menghapus kelompok subnet cache yang ditentukan. Anda tidak dapat menghapus grup subnet cache jika dikaitkan dengan cluster apa pun.

```
aws elasticache delete-cache-subnet-group \  
  --cache-subnet-group-name "mygroup"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Grup Subnet](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DeleteCacheSubnetGroup](#) di Referensi AWS CLI Perintah.

## **delete-global-replication-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-global-replication-group`.

### AWS CLI

Untuk menghapus grup replikasi global

`delete-global-replication-group` Contoh berikut menghapus grup replikasi global baru.

```
aws elasticache delete-global-replication-group \  
  --global-replication-group-id my-global-replication-group \  
  --retain-primary-replication-group
```

Output:

```

{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "sgaui-my-grg",
    "GlobalReplicationGroupDescription": "my-grg",
    "Status": "deleting",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.6",
    "Members": [
      {
        "ReplicationGroupId": "my-cluster-grg",
        "ReplicationGroupRegion": "us-west-2",
        "Role": "PRIMARY",
        "AutomaticFailover": "enabled",
        "Status": "associated"
      }
    ],
    "ClusterEnabled": false,
    "AuthTokenEnabled": false,
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}

```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DeleteGlobalReplicationGroup](#) di Referensi AWS CLI Perintah.

## delete-replication-group

Contoh kode berikut menunjukkan cara menggunakan `delete-replication-group`.

### AWS CLI

Untuk menghapus grup replikasi

`delete-replication-group` Contoh berikut menghapus grup replikasi yang ada. Secara default, operasi ini menghapus seluruh grup replikasi, termasuk primer/primer dan semua replika baca. Jika grup replikasi hanya memiliki satu primer, Anda dapat secara opsional menghapus hanya replika baca, sambil mempertahankan primer dengan menyetel `=true`. `RetainPrimaryCluster`

Ketika Anda menerima respons yang berhasil dari operasi ini, Amazon ElastiCache segera mulai menghapus sumber daya yang dipilih; Anda tidak dapat membatalkan atau mengembalikan operasi ini. Berlaku hanya untuk Redis.

```
aws elasticache delete-replication-group \  
  --replication-group-id "mygroup"
```

Output:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "mygroup",  
    "Description": "my group",  
    "Status": "deleting",  
    "PendingModifiedValues": {},  
    "AutomaticFailover": "disabled",  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:00-07:00",  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

- Untuk detail API, lihat [DeleteReplicationGroup](#) di Referensi AWS CLI Perintah.

## delete-snapshot

Contoh kode berikut menunjukkan cara menggunakan `delete-snapshot`.

### AWS CLI

Untuk menghapus snapshot

`delete-snapshot` Contoh berikut menghapus snapshot menggunakan mesin Redis.

```
aws elasticache delete-snapshot \  
  --snapshot-name mysnapshot
```

Output:

```
{  
  "Snapshot": {
```

```
"SnapshotName": "my-cluster-snapshot",
"ReplicationGroupId": "mycluster",
"ReplicationGroupDescription": "mycluster",
"SnapshotStatus": "deleting",
"SnapshotSource": "manual",
"CacheNodeType": "cache.r5.xlarge",
"Engine": "redis",
"EngineVersion": "5.0.5",
"PreferredMaintenanceWindow": "thu:12:00-thu:13:00",
"TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxxxx152:My_Topic",
"Port": 6379,
"CacheParameterGroupName": "default.redis5.0.cluster.on",
"CacheSubnetGroupName": "default",
"VpcId": "vpc-a3e97cdb",
"AutoMinorVersionUpgrade": true,
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "13:00-14:00",
"NumNodeGroups": 4,
"AutomaticFailover": "enabled",
"NodeSnapshots": [
  {
    "CacheClusterId": "mycluster-0002-003",
    "NodeGroupId": "0002",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2020-06-18T00:05:44.719000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  },
  {
    "CacheClusterId": "mycluster-0003-003",
    "NodeGroupId": "0003",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2019-12-05T19:13:15.912000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  },
  {
    "CacheClusterId": "mycluster-0004-002",
    "NodeGroupId": "0004",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2019-12-09T19:44:34.324000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  },
],
```



```

    {
      "CacheClusterId": "mycluster-0005-003",
      "NodeGroupId": "0005",
      "CacheNodeId": "0001",
      "CacheSize": "6 MB",
      "CacheNodeCreateTime": "2020-06-18T00:05:44.775000+00:00",
      "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Backup dan Restore for ElastiCache for Redis](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DeleteSnapshot](#) di Referensi AWS CLI Perintah.

## delete-user-group

Contoh kode berikut menunjukkan cara menggunakan `delete-user-group`.

### AWS CLI

Untuk menghapus grup pengguna

`delete-user-group` Contoh berikut menghapus grup pengguna.

```
aws elasticache delete-user-group \
  --user-group-id myusergroup
```

Output:

```

{
  "UserGroupId": "myusergroup",
  "Status": "deleting",
  "Engine": "redis",
  "UserIds": [
    "default"
  ],
  "ReplicationGroups": [],
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"
}

```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DeleteUserGroup](#) di Referensi AWS CLI Perintah.

## delete-user

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

### AWS CLI

Untuk menghapus pengguna

`delete-user` Contoh berikut menghapus pengguna.

```
aws elasticache delete-user \  
  --user-id user2
```

Output:

```
{  
  "UserId": "user1",  
  "UserName": "myUser",  
  "Status": "deleting",  
  "Engine": "redis",  
  "AccessString": "on ~* +@all",  
  "UserGroupIds": [  
    "myusergroup"  
  ],  
  "Authentication": {  
    "Type": "password",  
    "PasswordCount": 1  
  },  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx52:user:user1"  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## describe-cache-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-cache-clusters`.

### AWS CLI

Untuk mendeskripsikan cluster cache

`describe-cache-clusters` Contoh berikut menjelaskan cluster cache.

```
aws elasticache describe-cache-clusters
```

Output:

```
{
  "CacheClusters": [
    {
      "CacheClusterId": "my-cluster-003",
      "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
      "CacheNodeType": "cache.r5.large",
      "Engine": "redis",
      "EngineVersion": "5.0.5",
      "CacheClusterStatus": "available",
      "NumCacheNodes": 1,
      "PreferredAvailabilityZone": "us-west-2a",
      "CacheClusterCreateTime": "2019-11-26T01:22:52.396Z",
      "PreferredMaintenanceWindow": "mon:17:30-mon:18:30",
      "PendingModifiedValues": {},
      "NotificationConfiguration": {
        "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxx152:My_Topic",
        "TopicStatus": "active"
      },
      "CacheSecurityGroups": [],
      "CacheParameterGroup": {
        "CacheParameterGroupName": "default.redis5.0",
        "ParameterApplyStatus": "in-sync",
        "CacheNodeIdsToReboot": []
      },
      "CacheSubnetGroupName": "kxkxk",
      "AutoMinorVersionUpgrade": true,
      "SecurityGroups": [
        {
          "SecurityGroupId": "sg-xxxxxd7b",
```

```

        "Status": "active"
      }
    ],
    "ReplicationGroupId": "my-cluster",
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "06:30-07:30",
    "AuthTokenEnabled": false,
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false,
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx152:cluster:my-cache-
cluster",
    "ReplicationGroupLogDeliveryEnabled": false,
    "LogDeliveryConfigurations": [
      {
        "LogType": "slow-log",
        "DestinationType": "cloudwatch-logs",
        "DestinationDetails": {
          "CloudWatchLogsDetails": {
            "LogGroup": "test-log"
          }
        },
        "LogFormat": "text",
        "Status": "active"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola Cluster](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DescribeCacheClusters](#) di Referensi AWS CLI Perintah.

## describe-cache-engine-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-cache-engine-versions`.

### AWS CLI

Untuk menggambarkan versi mesin cache

`describe-cache-engine-versions` Contoh berikut mengembalikan daftar mesin cache yang tersedia dan versinya.

```
aws elasticache describe-cache-engine-versions \  
  --engine "Redis"
```

### Output:

```
{  
  "CacheEngineVersions": [  
    {  
      "Engine": "redis",  
      "EngineVersion": "2.6.13",  
      "CacheParameterGroupFamily": "redis2.6",  
      "CacheEngineDescription": "Redis",  
      "CacheEngineVersionDescription": "redis version 2.6.13"  
    },  
    {  
      "Engine": "redis",  
      "EngineVersion": "2.8.19",  
      "CacheParameterGroupFamily": "redis2.8",  
      "CacheEngineDescription": "Redis",  
      "CacheEngineVersionDescription": "redis version 2.8.19"  
    },  
    {  
      "Engine": "redis",  
      "EngineVersion": "2.8.21",  
      "CacheParameterGroupFamily": "redis2.8",  
      "CacheEngineDescription": "Redis",  
      "CacheEngineVersionDescription": "redis version 2.8.21"  
    },  
    {  
      "Engine": "redis",  
      "EngineVersion": "2.8.22",  
      "CacheParameterGroupFamily": "redis2.8",  
      "CacheEngineDescription": "Redis",  
      "CacheEngineVersionDescription": "redis version 2.8.22"  
    },  
    {  
      "Engine": "redis",  
      "EngineVersion": "2.8.23",  
      "CacheParameterGroupFamily": "redis2.8",  
      "CacheEngineDescription": "Redis",  
      "CacheEngineVersionDescription": "redis version 2.8.23"  
    }  
  ]  
}
```

```
    "Engine": "redis",
    "EngineVersion": "2.8.24",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.24"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.6",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.6"
  },
  {
    "Engine": "redis",
    "EngineVersion": "3.2.10",
    "CacheParameterGroupFamily": "redis3.2",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 3.2.10"
  },
  {
    "Engine": "redis",
    "EngineVersion": "3.2.4",
    "CacheParameterGroupFamily": "redis3.2",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 3.2.4"
  },
  {
    "Engine": "redis",
    "EngineVersion": "3.2.6",
    "CacheParameterGroupFamily": "redis3.2",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 3.2.6"
  },
  {
    "Engine": "redis",
    "EngineVersion": "4.0.10",
    "CacheParameterGroupFamily": "redis4.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 4.0.10"
  },
  {
    "Engine": "redis",
    "EngineVersion": "5.0.0",
```

```

    "CacheParameterGroupFamily": "redis5.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 5.0.0"
  },
  {
    "Engine": "redis",
    "EngineVersion": "5.0.3",
    "CacheParameterGroupFamily": "redis5.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 5.0.3"
  },
  {
    "Engine": "redis",
    "EngineVersion": "5.0.4",
    "CacheParameterGroupFamily": "redis5.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 5.0.4"
  },
  {
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "CacheParameterGroupFamily": "redis5.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 5.0.5"
  }
]
}

```

- Untuk detail API, lihat [DescribeCacheEngineVersions](#) di Referensi AWS CLI Perintah.

## describe-cache-parameter-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-cache-parameter-groups`.

### AWS CLI

Untuk menggambarkan grup parameter cache

`describe-cache-parameter-groups` Contoh berikut mengembalikan daftar deskripsi kelompok parameter cache.

```
aws elasticache describe-cache-parameter-groups \
```

```
--cache-parameter-group-name "mygroup"
```

Output:

```
{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "mygroup",
      "CacheParameterGroupFamily": "redis5.0",
      "Description": " "
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Parameter Mesin Menggunakan Grup Parameter](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DescribeCacheParameterGroups](#) di Referensi AWS CLI Perintah.

## describe-cache-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-cache-parameters`.

### AWS CLI

Untuk menggambarkan parameter cache

Berikut "describe-cache-parameters" contoh mengembalikan daftar parameter rinci untuk kelompok parameter cache tertentu.

```
aws elasticache describe-cache-parameters \
  --cache-parameter-group-name "myparamgroup"
```

Output:

```
{
  "Parameters": [
    {
      "ParameterName": "activedefrag",
      "ParameterValue": "yes",
      "Description": "Enabled active memory defragmentation",
      "Source": "user",
    }
  ]
}
```



```

    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-cycle-max",
    "ParameterValue": "75",
    "Description": "Maximal effort for defrag in CPU percentage",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-75",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-cycle-min",
    "ParameterValue": "5",
    "Description": "Minimal effort for defrag in CPU percentage",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-75",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-ignore-bytes",
    "ParameterValue": "104857600",
    "Description": "Minimum amount of fragmentation waste to start active
defrag",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1048576-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-max-scan-fields",
    "ParameterValue": "1000",

```

```

    "Description": "Maximum number of set/hash/zset/list fields that will be
processed from the main dictionary scan",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-1000000",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-threshold-lower",
    "ParameterValue": "10",
    "Description": "Minimum percentage of fragmentation to start active
defrag",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-threshold-upper",
    "ParameterValue": "100",
    "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "activeresharding",
    "ParameterValue": "yes",
    "Description": "Apply rehashing or not.",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },

```

```
{
  "ParameterName": "appendfsync",
  "ParameterValue": "everysec",
  "Description": "fsync policy for AOF persistence",
  "Source": "system",
  "DataType": "string",
  "AllowedValues": "always, everysec, no",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "appendonly",
  "ParameterValue": "no",
  "Description": "Enable Redis persistence.",
  "Source": "system",
  "DataType": "string",
  "AllowedValues": "yes, no",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-normal-hard-limit",
  "ParameterValue": "0",
  "Description": "Normal client output buffer hard limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-normal-soft-limit",
  "ParameterValue": "0",
  "Description": "Normal client output buffer soft limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
},
```

```
{
  "ParameterName": "client-output-buffer-limit-normal-soft-seconds",
  "ParameterValue": "0",
  "Description": "Normal client output buffer soft limit in seconds.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-pubsub-hard-limit",
  "ParameterValue": "33554432",
  "Description": "Pubsub client output buffer hard limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-pubsub-soft-limit",
  "ParameterValue": "8388608",
  "Description": "Pubsub client output buffer soft limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-pubsub-soft-seconds",
  "ParameterValue": "60",
  "Description": "Pubsub client output buffer soft limit in seconds.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
}
```

```
{
  "ParameterName": "client-output-buffer-limit-replica-soft-seconds",
  "ParameterValue": "60",
  "Description": "Replica client output buffer soft limit in seconds.",
  "Source": "system",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-query-buffer-limit",
  "ParameterValue": "1073741824",
  "Description": "Max size of a single client query buffer",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "1048576-1073741824",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "close-on-replica-write",
  "ParameterValue": "yes",
  "Description": "If enabled, clients who attempt to write to a read-only replica will be disconnected. Applicable to 2.8.23 and higher.",
  "Source": "user",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "cluster-enabled",
  "ParameterValue": "no",
  "Description": "Enable cluster mode",
  "Source": "user",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "requires-reboot"
}
```

```
    },
    {
      "ParameterName": "cluster-require-full-coverage",
      "ParameterValue": "no",
      "Description": "Whether cluster becomes unavailable if one or more slots
are not covered",
      "Source": "user",
      "DataType": "string",
      "AllowedValues": "yes,no",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "immediate"
    },
    {
      "ParameterName": "databases",
      "ParameterValue": "16",
      "Description": "Set the number of databases.",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "1-1200000",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "requires-reboot"
    },
    {
      "ParameterName": "hash-max-ziplist-entries",
      "ParameterValue": "512",
      "Description": "The maximum number of hash entries in order for the
dataset to be compressed.",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "immediate"
    },
    {
      "ParameterName": "hash-max-ziplist-value",
      "ParameterValue": "64",
      "Description": "The threshold of biggest hash entries in order for the
dataset to be compressed.",
      "Source": "user",
      "DataType": "integer",
      "AllowedValues": "0-",
```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "hll-sparse-max-bytes",
    "ParameterValue": "3000",
    "Description": "HyperLogLog sparse representation bytes limit",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-16000",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-eviction",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on evictions",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-expire",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on expired keys",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-server-del",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on key updates",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-decay-time",
    "ParameterValue": "1",
    "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-log-factor",
    "ParameterValue": "10",
    "Description": "The log factor for incrementing key counter for LFU
eviction policy",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-compress-depth",
    "ParameterValue": "0",
    "Description": "Number of quicklist ziplist nodes from each side of
the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-max-ziplist-size",
    "ParameterValue": "-2",
```



```

    "Description": "The number of entries allowed per internal list node can
be specified as a fixed maximum size or a maximum number of elements",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "-5,-4,-3,-2,-1,1-",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-replicate-commands",
    "ParameterValue": "yes",
    "Description": "Always enable Lua effect replication or not",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-time-limit",
    "ParameterValue": "5000",
    "Description": "Max execution time of a Lua script in milliseconds. 0
for unlimited execution without warnings.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "5000",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "maxclients",
    "ParameterValue": "65000",
    "Description": "The maximum number of Redis clients.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-65000",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {

```

```
    "ParameterName": "maxmemory-policy",
    "ParameterValue": "volatile-lru",
    "Description": "Max memory policy.",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "maxmemory-samples",
    "ParameterValue": "3",
    "Description": "Max memory samples.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "min-replicas-max-lag",
    "ParameterValue": "10",
    "Description": "The maximum amount of replica lag in seconds beyond
which the master would stop taking writes. A value of 0 means the master always
takes writes.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "min-replicas-to-write",
    "ParameterValue": "0",
    "Description": "The minimum number of replicas that must be present with
lag no greater than min-replicas-max-lag for master to take writes. Setting this to
0 means the master always takes writes.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
```

```

        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "notify-keyspace-events",
        "Description": "The keyspace events for Redis to notify Pub/Sub clients
about. By default all notifications are disabled",
        "Source": "user",
        "DataType": "string",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "proto-max-bulk-len",
        "ParameterValue": "536870912",
        "Description": "Max size of a single element request",
        "Source": "user",
        "DataType": "integer",
        "AllowedValues": "1048576-536870912",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "rename-commands",
        "ParameterValue": "",
        "Description": "Redis commands that can be dynamically renamed by the
customer",
        "Source": "user",
        "DataType": "string",
        "AllowedValues":
"APPEND,BITCOUNT,BITFIELD,BITOP,BITPOS,BLPOP,BRPOP,BRPOPLUSH,BZPOPMIN,BZPOPMAX,CLIENT,COMM
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.3",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "repl-backlog-size",
        "ParameterValue": "1048576",
        "Description": "The replication backlog size in bytes for PSYNC. This is
the size of the buffer which accumulates slave data when slave is disconnected for

```

```
some time, so that when slave reconnects again, only transfer the portion of data
which the slave missed. Minimum value is 16K.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "16384-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "repl-backlog-ttl",
  "ParameterValue": "3600",
  "Description": "The amount of time in seconds after the master no longer
have any slaves connected for the master to free the replication backlog. A value
of 0 means to never release the backlog.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "replica-allow-chaining",
  "ParameterValue": "no",
  "Description": "Configures if chaining of replicas is allowed",
  "Source": "system",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "replica-ignore-maxmemory",
  "ParameterValue": "yes",
  "Description": "Determines if replica ignores maxmemory setting by not
evicting items independent from the master",
  "Source": "system",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
}
```

```
  },
  {
    "ParameterName": "replica-lazy-flush",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous flushDB during replica sync",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "reserved-memory-percent",
    "ParameterValue": "25",
    "Description": "The percent of memory reserved for non-cache memory
usage. You may want to increase this parameter for nodes with read replicas, AOF
enabled, etc, to reduce swap usage.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "set-max-intset-entries",
    "ParameterValue": "512",
    "Description": "The limit in the size of the set in order for the
dataset to be compressed.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "slowlog-log-slower-than",
    "ParameterValue": "10000",
    "Description": "The execution time, in microseconds, to exceed in order
for the command to get logged. Note that a negative number disables the slow log,
while a value of zero forces the logging of every command.",
    "Source": "user",
```

```
    "DataType": "integer",
    "AllowedValues": "-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "slowlog-max-len",
    "ParameterValue": "128",
    "Description": "The length of the slow log. There is no limit to this
length. Just be aware that it will consume memory. You can reclaim memory used by
the slow log with SLOWLOG RESET.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "stream-node-max-bytes",
    "ParameterValue": "4096",
    "Description": "The maximum size of a single node in a stream in bytes",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "stream-node-max-entries",
    "ParameterValue": "100",
    "Description": "The maximum number of items a single node in a stream
can contain",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "tcp-keepalive",
```

```
    "ParameterValue": "300",
    "Description": "If non-zero, send ACKs every given number of seconds.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "timeout",
    "ParameterValue": "0",
    "Description": "Close connection if client is idle for a given number of
seconds, or never if 0.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0,20-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "zset-max-ziplist-entries",
    "ParameterValue": "128",
    "Description": "The maximum number of sorted set entries in order for
the dataset to be compressed.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "zset-max-ziplist-value",
    "ParameterValue": "64",
    "Description": "The threshold of biggest sorted set entries in order for
the dataset to be compressed.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  }
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Manajemen Parameter](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DescribeCacheParameters](#) di Referensi AWS CLI Perintah.

## describe-cache-subnet-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-cache-subnet-groups`.

### AWS CLI

Untuk menggambarkan grup subnet cache

`describe-cache-subnet-groups` Contoh berikut mengembalikan daftar kelompok subnet.

```
aws elasticache describe-cache-subnet-groups
```

Output:

```
{  
  "CacheSubnetGroups": [  
    {  
      "CacheSubnetGroupName": "default",  
      "CacheSubnetGroupDescription": "Default CacheSubnetGroup",  
      "VpcId": "vpc-a3e97cdb",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-8d4bacf5",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2b"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-dde21380",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2c"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-6485ec4f",
```



```

        "SubnetAvailabilityZone": {
            "Name": "us-west-2d"
        }
    },
    {
        "SubnetIdentifier": "subnet-b4ebebff",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2a"
        }
    }
]
},
{
    "CacheSubnetGroupName": "kxkxk",
    "CacheSubnetGroupDescription": "mygroup",
    "VpcId": "vpc-a3e97cdb",
    "Subnets": [
        {
            "SubnetIdentifier": "subnet-b4ebebff",
            "SubnetAvailabilityZone": {
                "Name": "us-west-2a"
            }
        }
    ]
},
{
    "CacheSubnetGroupName": "test",
    "CacheSubnetGroupDescription": "test",
    "VpcId": "vpc-a3e97cdb",
    "Subnets": [
        {
            "SubnetIdentifier": "subnet-b4ebebff",
            "SubnetAvailabilityZone": {
                "Name": "us-west-2a"
            }
        }
    ]
}
]
}
}

```

Untuk informasi selengkapnya, lihat [Subnet dan Grup Subnet](#) di Panduan Pengguna Elasticache atau [Subnet dan Grup Subnet](#) di Panduan Pengguna untuk Memcached. ElastiCache

- Untuk detail API, lihat [DescribeCacheSubnetGroups](#) di Referensi AWS CLI Perintah.

## describe-engine-default-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-engine-default-parameters`.

### AWS CLI

Untuk menggambarkan parameter default mesin

`describe-engine-default-parameters` Contoh berikut mengembalikan mesin default dan informasi parameter sistem untuk mesin cache tertentu.

```
aws elasticache describe-engine-default-parameters \  
  --cache-parameter-group-family "redis5.0"
```

### Output:

```
{  
  "EngineDefaults": {  
    "Parameters": [  
      {  
        "ParameterName": "activedefrag",  
        "ParameterValue": "no",  
        "Description": "Enabled active memory defragmentation",  
        "Source": "system",  
        "DataType": "string",  
        "AllowedValues": "yes,no",  
        "IsModifiable": true,  
        "MinimumEngineVersion": "5.0.0",  
        "ChangeType": "immediate"  
      },  
      {  
        "ParameterName": "active-defrag-cycle-max",  
        "ParameterValue": "75",  
        "Description": "Maximal effort for defrag in CPU percentage",  
        "Source": "system",  
        "DataType": "integer",  
        "AllowedValues": "1-75",  
        "IsModifiable": true,  
        "MinimumEngineVersion": "5.0.0",  
        "ChangeType": "immediate"  
      },  
    ],  
  },  
}
```

```
{
  "ParameterName": "active-defrag-cycle-min",
  "ParameterValue": "5",
  "Description": "Minimal effort for defrag in CPU percentage",
  "Source": "system",
  "DataType": "integer",
  "AllowedValues": "1-75",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "active-defrag-ignore-bytes",
  "ParameterValue": "104857600",
  "Description": "Minimum amount of fragmentation waste to start
active defrag",
  "Source": "system",
  "DataType": "integer",
  "AllowedValues": "1048576-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "active-defrag-max-scan-fields",
  "ParameterValue": "1000",
  "Description": "Maximum number of set/hash/zset/list fields that
will be processed from the main dictionary scan",
  "Source": "system",
  "DataType": "integer",
  "AllowedValues": "1-1000000",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "active-defrag-threshold-lower",
  "ParameterValue": "10",
  "Description": "Minimum percentage of fragmentation to start active
defrag",
  "Source": "system",
  "DataType": "integer",
  "AllowedValues": "1-100",
  "IsModifiable": true,
```

```
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-threshold-upper",
    "ParameterValue": "100",
    "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "activeresharding",
    "ParameterValue": "yes",
    "Description": "Apply rehashing or not.",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "appendfsync",
    "ParameterValue": "everysec",
    "Description": "fsync policy for AOF persistence",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "always,everysec,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "appendonly",
    "ParameterValue": "no",
    "Description": "Enable Redis persistence.",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
```

```
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-normal-hard-limit",
    "ParameterValue": "0",
    "Description": "Normal client output buffer hard limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-normal-soft-limit",
    "ParameterValue": "0",
    "Description": "Normal client output buffer soft limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-normal-soft-seconds",
    "ParameterValue": "0",
    "Description": "Normal client output buffer soft limit in seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-pubsub-hard-limit",
    "ParameterValue": "33554432",
    "Description": "Pubsub client output buffer hard limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-pubsub-soft-limit",
    "ParameterValue": "8388608",
    "Description": "Pubsub client output buffer soft limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-pubsub-soft-seconds",
    "ParameterValue": "60",
    "Description": "Pubsub client output buffer soft limit in seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-replica-soft-seconds",
    "ParameterValue": "60",
    "Description": "Replica client output buffer soft limit in
seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-query-buffer-limit",
    "ParameterValue": "1073741824",
    "Description": "Max size of a single client query buffer",
    "Source": "system",
    "DataType": "integer",
```

```
    "AllowedValues": "1048576-1073741824",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "close-on-replica-write",
    "ParameterValue": "yes",
    "Description": "If enabled, clients who attempt to write to a read-only replica will be disconnected. Applicable to 2.8.23 and higher.",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "cluster-enabled",
    "ParameterValue": "no",
    "Description": "Enable cluster mode",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "cluster-require-full-coverage",
    "ParameterValue": "no",
    "Description": "Whether cluster becomes unavailable if one or more slots are not covered",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "databases",
    "ParameterValue": "16",
    "Description": "Set the number of databases.",
```

```
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1-1200000",
        "IsModifiable": false,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "requires-reboot"
    },
    {
        "ParameterName": "hash-max-ziplist-entries",
        "ParameterValue": "512",
        "Description": "The maximum number of hash entries in order for the
dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "hash-max-ziplist-value",
        "ParameterValue": "64",
        "Description": "The threshold of biggest hash entries in order for
the dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "hll-sparse-max-bytes",
        "ParameterValue": "3000",
        "Description": "HyperLogLog sparse representation bytes limit",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1-16000",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "lazyfree-lazy-eviction",
```



```
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on evictions",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-expire",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on expired keys",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-server-del",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on key updates",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-decay-time",
    "ParameterValue": "1",
    "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
```

```

    "ParameterName": "lfu-log-factor",
    "ParameterValue": "10",
    "Description": "The log factor for incrementing key counter for LFU
eviction policy",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-compress-depth",
    "ParameterValue": "0",
    "Description": "Number of quicklist ziplist nodes from each side
of the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-max-ziplist-size",
    "ParameterValue": "-2",
    "Description": "The number of entries allowed per internal list node
can be specified as a fixed maximum size or a maximum number of elements",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "-5,-4,-3,-2,-1,1-",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-replicate-commands",
    "ParameterValue": "yes",
    "Description": "Always enable Lua effect replication or not",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,

```

```
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-time-limit",
    "ParameterValue": "5000",
    "Description": "Max execution time of a Lua script in milliseconds.
0 for unlimited execution without warnings.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "5000",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "maxclients",
    "ParameterValue": "65000",
    "Description": "The maximum number of Redis clients.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-65000",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "maxmemory-policy",
    "ParameterValue": "volatile-lru",
    "Description": "Max memory policy.",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "maxmemory-samples",
    "ParameterValue": "3",
    "Description": "Max memory samples.",
    "Source": "system",
    "DataType": "integer",
```

```
    "AllowedValues": "1-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "min-replicas-max-lag",
    "ParameterValue": "10",
    "Description": "The maximum amount of replica lag in seconds beyond
which the master would stop taking writes. A value of 0 means the master always
takes writes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "min-replicas-to-write",
    "ParameterValue": "0",
    "Description": "The minimum number of replicas that must be present
with lag no greater than min-replicas-max-lag for master to take writes. Setting
this to 0 means the master always takes writes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "notify-keyspace-events",
    "Description": "The keyspace events for Redis to notify Pub/Sub
clients about. By default all notifications are disabled",
    "Source": "system",
    "DataType": "string",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "proto-max-bulk-len",
    "ParameterValue": "536870912",
```

```

    "Description": "Max size of a single element request",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1048576-536870912",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "rename-commands",
    "ParameterValue": "",
    "Description": "Redis commands that can be dynamically renamed by
the customer",
    "Source": "system",
    "DataType": "string",
    "AllowedValues":
"APPEND,BITCOUNT,BITFIELD,BITOP,BITPOS,BLPOP,BRPOP,BRPOPLPUSH,BZPOPMIN,BZPOPMAX,CLIENT,COMM
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.3",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "repl-backlog-size",
    "ParameterValue": "1048576",
    "Description": "The replication backlog size in bytes for PSYNC.
This is the size of the buffer which accumulates slave data when slave is
disconnected for some time, so that when slave reconnects again, only transfer the
portion of data which the slave missed. Minimum value is 16K.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "16384-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "repl-backlog-ttl",
    "ParameterValue": "3600",
    "Description": "The amount of time in seconds after the master no
longer have any slaves connected for the master to free the replication backlog. A
value of 0 means to never release the backlog.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",

```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-allow-chaining",
    "ParameterValue": "no",
    "Description": "Configures if chaining of replicas is allowed",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-ignore-maxmemory",
    "ParameterValue": "yes",
    "Description": "Determines if replica ignores maxmemory setting by
not evicting items independent from the master",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-lazy-flush",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous flushDB during replica
sync",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "reserved-memory-percent",
    "ParameterValue": "25",
```

```

        "Description": "The percent of memory reserved for non-cache memory
usage. You may want to increase this parameter for nodes with read replicas, AOF
enabled, etc, to reduce swap usage.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-100",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "set-max-intset-entries",
        "ParameterValue": "512",
        "Description": "The limit in the size of the set in order for the
dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "slowlog-log-slower-than",
        "ParameterValue": "10000",
        "Description": "The execution time, in microseconds, to exceed in
order for the command to get logged. Note that a negative number disables the slow
log, while a value of zero forces the logging of every command.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "slowlog-max-len",
        "ParameterValue": "128",
        "Description": "The length of the slow log. There is no limit to
this length. Just be aware that it will consume memory. You can reclaim memory used
by the slow log with SLOWLOG RESET.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",

```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "stream-node-max-bytes",
    "ParameterValue": "4096",
    "Description": "The maximum size of a single node in a stream in
bytes",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "stream-node-max-entries",
    "ParameterValue": "100",
    "Description": "The maximum number of items a single node in a
stream can contain",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "tcp-keepalive",
    "ParameterValue": "300",
    "Description": "If non-zero, send ACKs every given number of
seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "timeout",
    "ParameterValue": "0",
```



```

        "Description": "Close connection if client is idle for a given
number of seconds, or never if 0.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0,20-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "zset-max-ziplist-entries",
        "ParameterValue": "128",
        "Description": "The maximum number of sorted set entries in order
for the dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "zset-max-ziplist-value",
        "ParameterValue": "64",
        "Description": "The threshold of biggest sorted set entries in order
for the dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    }
}
]
}
}

```

- Untuk detail API, lihat [DescribeEngineDefaultParameters](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

## AWS CLI

Untuk menggambarkan peristiwa kelompok replikasi

`describe-events` Contoh berikut mengembalikan daftar peristiwa untuk grup replikasi.

```
aws elasticache describe-events \  
  --source-identifier test-cluster \  
  --source-type replication-group
```

Output:

```
{  
  "Events": [  
    {  
      "SourceIdentifier": "test-cluster",  
      "SourceType": "replication-group",  
      "Message": "Automatic failover has been turned on for replication group  
test-cluster",  
      "Date": "2020-03-18T23:51:34.457Z"  
    },  
    {  
      "SourceIdentifier": "test-cluster",  
      "SourceType": "replication-group",  
      "Message": "Replication group test-cluster created",  
      "Date": "2020-03-18T23:50:31.378Z"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Memantau Peristiwa](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## `describe-global-replication-groups`

Contoh kode berikut menunjukkan cara menggunakan `describe-global-replication-groups`.

## AWS CLI

Untuk menggambarkan kelompok replikasi global

`describe-global-replication-groups` Contoh berikut mengembalikan rincian datastore Global.

```
aws elasticache describe-global-replication-groups \
  --global-replication-group-id my-grg
```

Output:

```
{
  "GlobalReplicationGroups": [
    {
      "GlobalReplicationGroupId": "my-grg",
      "GlobalReplicationGroupDescription": "my-grg",
      "Status": "creating",
      "CacheNodeType": "cache.r5.large",
      "Engine": "redis",
      "EngineVersion": "5.0.6",
      "ClusterEnabled": false,
      "AuthTokenEnabled": false,
      "TransitEncryptionEnabled": false,
      "AtRestEncryptionEnabled": false
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DescribeGlobalReplicationGroups](#) di Referensi AWS CLI Perintah.

## **describe-replication-groups**

Contoh kode berikut menunjukkan cara menggunakan `describe-replication-groups`.

AWS CLI

Untuk mengembalikan daftar rincian grup replikasi

`describe-replication-groups` Contoh berikut mengembalikan kelompok replikasi.

```
aws elasticache describe-replication-groups
```

## Output:

```
{
  "ReplicationGroups": [
    {
      "ReplicationGroupId": "my-cluster",
      "Description": "mycluster",
      "Status": "available",
      "PendingModifiedValues": {},
      "MemberClusters": [
        "pat-cluster-001",
        "pat-cluster-002",
        "pat-cluster-003",
        "pat-cluster-004"
      ],
      "NodeGroups": [
        {
          "NodeGroupId": "0001",
          "Status": "available",
          "PrimaryEndpoint": {
            "Address": "my-
cluster.xxxxih.ng.0001.usw2.cache.amazonaws.com",
            "Port": 6379
          },
          "ReaderEndpoint": {
            "Address": "my-cluster-
ro.xxxxih.ng.0001.usw2.cache.amazonaws.com",
            "Port": 6379
          },
          "NodeGroupMembers": [
            {
              "CacheClusterId": "my-cluster-001",
              "CacheNodeId": "0001",
              "ReadEndpoint": {
                "Address": "pat-
cluster-001.xxxxih.0001.usw2.cache.amazonaws.com",
                "Port": 6379
              },
              "PreferredAvailabilityZone": "us-west-2a",
              "CurrentRole": "primary"
            },
            {
              "CacheClusterId": "my-cluster-002",
              "CacheNodeId": "0001",
```

```

        "ReadEndpoint": {
            "Address": "pat-
cluster-002.xxxxih.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
    },
    {
        "CacheClusterId": "my-cluster-003",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Address": "pat-
cluster-003.xxxxih.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
    },
    {
        "CacheClusterId": "my-cluster-004",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Address": "pat-
cluster-004.xxxxih.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
    }
]
}
],
"AutomaticFailover": "disabled",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.xlarge",
"AuthTokenEnabled": false,
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false,
"ARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxxxx152:replicationgroup:my-cluster",
"LogDeliveryConfigurations": [

```

```

    {
      "LogType": "slow-log",
      "DestinationType": "cloudwatch-logs",
      "DestinationDetails": {
        "CloudWatchLogsDetails": {
          "LogGroup": "test-log"
        }
      },
      "LogFormat": "json",
      "Status": "active"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Mengelola Cluster](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DescribeReplicationGroups](#) di Referensi AWS CLI Perintah.

## describe-reserved-cache-nodes-offerings

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-cache-nodes-offerings`.

### AWS CLI

Untuk menggambarkan `reserved-cache-nodes-offerings`

`describe-reserved-cache-nodes-offerings` Contoh berikut mengembalikan rincian `reserved-cache-node` pilihan.

```
aws elasticache describe-reserved-cache-nodes-offerings
```

Output:

```

{
  "ReservedCacheNodesOfferings": [
    {
      "ReservedCacheNodesOfferingId": "01ce0a19-a476-41cb-8aee-48eacbcd8e5",
      "CacheNodeType": "cache.t3.small",

```

```

    "Duration": 31536000,
    "FixedPrice": 97.0,
    "UsagePrice": 0.0,
    "ProductDescription": "memcached",
    "OfferingType": "Partial Upfront",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.011,
        "RecurringChargeFrequency": "Hourly"
      }
    ]
  },
  {
    "ReservedCacheNodesOfferingId": "0443a27b-4da5-4b90-b92d-929fbd7abed2",
    "CacheNodeType": "cache.m3.2xlarge",
    "Duration": 31536000,
    "FixedPrice": 1772.0,
    "UsagePrice": 0.0,
    "ProductDescription": "redis",
    "OfferingType": "Heavy Utilization",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.25,
        "RecurringChargeFrequency": "Hourly"
      }
    ]
  },
  ...
]
}

```

Untuk informasi selengkapnya, lihat [Mendapatkan Info Tentang Penawaran Node Cadangan](#) di Panduan Pengguna Elasticache Redis atau [Mendapatkan Info Tentang Penawaran Node Cadangan](#) di Panduan Pengguna [Memcached Elasticache](#).

- Untuk detail API, lihat [DescribeReservedCacheNodesOfferings](#) di Referensi AWS CLI Perintah.

## **describe-reserved-cache-nodes**

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-cache-nodes`.

## AWS CLI

Untuk menggambarkan node cache yang dicadangkan

`describe-reserved-cache-nodes` Contoh berikut mengembalikan informasi tentang node cache reserved untuk akun ini, atau tentang node cache reserved yang ditentukan.

```
aws elasticache describe-reserved-cache-nodes
```

Output:

```
{
  "ReservedCacheNodes": [
    {
      "ReservedCacheNodeId": "mynode",
      "ReservedCacheNodesOfferingId": "xxxxxxxxxx-xxxxxx-xxxxxx-xxxxx-xxxxxxxxxx71",
      "CacheNodeType": "cache.t3.small",
      "StartTime": "2019-12-06T02:50:44.003Z",
      "Duration": 31536000,
      "FixedPrice": 0.0,
      "UsagePrice": 0.0,
      "CacheNodeCount": 1,
      "ProductDescription": "redis",
      "OfferingType": "No Upfront",
      "State": "payment-pending",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": 0.023,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "ReservationARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxxxx52:reserved-instance:mynode"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Biaya dengan Node Cadangan](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DescribeReservedCacheNodes](#) di Referensi AWS CLI Perintah.



## describe-service-updates

Contoh kode berikut menunjukkan cara menggunakan `describe-service-updates`.

### AWS CLI

Untuk menggambarkan pembaruan layanan

`describe-service-updates` Contoh berikut mengembalikan rincian tentang update layanan.

```
aws elasticache describe-service-updates
```

Output:

```
{
  "ServiceUpdates": [
    {
      "ServiceUpdateName": "elc-xxxxxxx7-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateEndDate": "2020-02-09T15:59:59Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateDescription": "Upgrades to improve the security,
reliability, and operational performance of your ElastiCache nodes",
      "ServiceUpdateType": "security-update",
      "Engine": "redis, memcached",
      "EngineVersion": "redis 2.6.13 and onwards, memcached 1.4.5 and
onwards",
      "AutoUpdateAfterRecommendedApplyByDate": false,
      "EstimatedUpdateTime": "30 minutes per node"
    },
    {
      "ServiceUpdateName": "elc-xxxxxxx4-001",
      "ServiceUpdateReleaseDate": "2019-06-11T15:00:00Z",
      "ServiceUpdateEndDate": "2019-10-01T09:24:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateRecommendedApplyByDate": "2019-07-11T14:59:59Z",
      "ServiceUpdateStatus": "expired",
      "ServiceUpdateDescription": "Upgrades to improve the security,
reliability, and operational performance of your ElastiCache nodes",
      "ServiceUpdateType": "security-update",
      "Engine": "redis",
      "EngineVersion": "redis 3.2.6, redis 4.0 and onwards",
    }
  ]
}
```

```

        "AutoUpdateAfterRecommendedApplyByDate": false,
        "EstimatedUpdateTime": "30 minutes per node"
    }
]
}

```

- Untuk detail API, lihat [DescribeServiceUpdates](#) di Referensi AWS CLI Perintah.

## describe-snapshots

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshots`.

### AWS CLI

Untuk menggambarkan snapshot

Contoh “deskripsi-snapshot” berikut menampilkan informasi tentang snapshot cluster atau grup replikasi Anda.

```
aws elasticache describe-snapshots
```

Output:

```

{
  "Snapshots": [
    {
      "SnapshotName": "automatic.my-cluster2-002-2019-12-05-06-38",
      "CacheClusterId": "my-cluster2-002",
      "SnapshotStatus": "available",
      "SnapshotSource": "automated",
      "CacheNodeType": "cache.r5.large",
      "Engine": "redis",
      "EngineVersion": "5.0.5",
      "NumCacheNodes": 1,
      "PreferredAvailabilityZone": "us-west-2a",
      "CacheClusterCreateTime": "2019-11-26T01:22:52.396Z",
      "PreferredMaintenanceWindow": "mon:17:30-mon:18:30",
      "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxx52:My_Topic",
      "Port": 6379,
      "CacheParameterGroupName": "default.redis5.0",
      "CacheSubnetGroupName": "kxkxk",
      "VpcId": "vpc-a3e97cdb",
    }
  ]
}

```

```

    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 1,
    "SnapshotWindow": "06:30-07:30",
    "NodeSnapshots": [
      {
        "CacheNodeId": "0001",
        "CacheSize": "5 MB",
        "CacheNodeCreateTime": "2019-11-26T01:22:52.396Z",
        "SnapshotCreateTime": "2019-12-05T06:38:23Z"
      }
    ]
  },
  {
    "SnapshotName": "myreplica-backup",
    "CacheClusterId": "myreplica",
    "SnapshotStatus": "available",
    "SnapshotSource": "manual",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2a",
    "CacheClusterCreateTime": "2019-11-26T00:14:52.439Z",
    "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0",
    "CacheSubnetGroupName": "kxkxk",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "09:00-10:00",
    "NodeSnapshots": [
      {
        "CacheNodeId": "0001",
        "CacheSize": "5 MB",
        "CacheNodeCreateTime": "2019-11-26T00:14:52.439Z",
        "SnapshotCreateTime": "2019-11-26T00:25:01Z"
      }
    ]
  },
  {
    "SnapshotName": "my-cluster",
    "CacheClusterId": "my-cluster-003",

```

```

    "SnapshotStatus": "available",
    "SnapshotSource": "manual",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2a",
    "CacheClusterCreateTime": "2019-11-25T23:56:17.186Z",
    "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0",
    "CacheSubnetGroupName": "kxkxk",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "09:00-10:00",
    "NodeSnapshots": [
      {
        "CacheNodeId": "0001",
        "CacheSize": "5 MB",
        "CacheNodeCreateTime": "2019-11-25T23:56:17.186Z",
        "SnapshotCreateTime": "2019-11-26T03:08:33Z"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Backup dan Restore for ElastiCache for Redis](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DescribeSnapshots](#) di Referensi AWS CLI Perintah.

## describe-update-actions

Contoh kode berikut menunjukkan cara menggunakan `describe-update-actions`.

### AWS CLI

Untuk menjelaskan tindakan pembaruan

`describe-update-actions` Contoh berikut mengembalikan rincian tindakan update.

```
aws elasticache describe-update-actions
```

**Output:**

```
{
  "UpdateActions": [
    {
      "ReplicationGroupId": "mycluster",
      "ServiceUpdateName": "elc-20191007-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
      "ServiceUpdateType": "security-update",
      "UpdateActionAvailableDate": "2019-12-05T19:15:19.995Z",
      "UpdateActionStatus": "complete",
      "NodesUpdated": "9/9",
      "UpdateActionStatusModifiedDate": "2019-12-05T19:15:20.461Z",
      "SlaMet": "n/a",
      "Engine": "redis"
    },
    {
      "CacheClusterId": "my-memcached-cluster",
      "ServiceUpdateName": "elc-20191007-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
      "ServiceUpdateType": "security-update",
      "UpdateActionAvailableDate": "2019-12-04T18:26:05.349Z",
      "UpdateActionStatus": "complete",
      "NodesUpdated": "1/1",
      "UpdateActionStatusModifiedDate": "2019-12-04T18:26:05.352Z",
      "SlaMet": "n/a",
      "Engine": "redis"
    },
    {
      "ReplicationGroupId": "my-cluster",
      "ServiceUpdateName": "elc-20191007-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
```

```

    "ServiceUpdateType": "security-update",
    "UpdateActionAvailableDate": "2019-11-26T03:36:26.320Z",
    "UpdateActionStatus": "complete",
    "NodesUpdated": "4/4",
    "UpdateActionStatusModifiedDate": "2019-12-04T22:11:12.664Z",
    "SlaMet": "n/a",
    "Engine": "redis"
  },
  {
    "ReplicationGroupId": "my-cluster2",
    "ServiceUpdateName": "elc-20191007-001",
    "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
    "ServiceUpdateSeverity": "important",
    "ServiceUpdateStatus": "available",
    "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
    "ServiceUpdateType": "security-update",
    "UpdateActionAvailableDate": "2019-11-26T01:26:01.617Z",
    "UpdateActionStatus": "complete",
    "NodesUpdated": "3/3",
    "UpdateActionStatusModifiedDate": "2019-11-26T01:26:01.753Z",
    "SlaMet": "n/a",
    "Engine": "redis"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Pembaruan Layanan Mandiri di Amazon ElastiCache di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [DescribeUpdateActions](#) di Referensi AWS CLI Perintah.

## describe-user-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-user-groups`.

### AWS CLI

Untuk menggambarkan grup pengguna

`describe-user-groups` Contoh berikut mengembalikan daftar kelompok pengguna.

```
aws elasticache describe-user-groups
```

**Output:**

```
{
  "UserGroups": [
    {
      "UserGroupId": "myusergroup",
      "Status": "active",
      "Engine": "redis",
      "UserIds": [
        "default"
      ],
      "ReplicationGroups": [],
      "ARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxx52:usergroup:myusergroup"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [DescribeUserGroups](#) di Referensi AWS CLI Perintah.

**describe-users**

Contoh kode berikut menunjukkan cara menggunakan `describe-users`.

**AWS CLI**

Untuk mendeskripsikan pengguna

`describe-users` Contoh berikut mengembalikan daftar pengguna.

```
aws elasticache describe-users
```

**Output:**

```
{
  "Users": [
    {
      "UserId": "default",
      "UserName": "default",
      "Status": "active",
```

```

    "Engine": "redis",
    "AccessString": "on ~* +@all",
    "UserGroupIds": [
        "myusergroup"
    ],
    "Authentication": {
        "Type": "no-password"
    },
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:default"
},
{
    "UserId": "user1",
    "UserName": "myUser",
    "Status": "active",
    "Engine": "redis",
    "AccessString": "on ~* +@all",
    "UserGroupIds": [],
    "Authentication": {
        "Type": "password",
        "PasswordCount": 1
    },
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user1"
},
{
    "UserId": "user2",
    "UserName": "myUser",
    "Status": "active",
    "Engine": "redis",
    "AccessString": "on ~app:* -@all +@read +@hash +@bitmap +@geo -setbit -
bitfield -hset -hsetnx -hmset -hincrby -hincrbyfloat -hdel -bitop -geoadd -georadius
-georadiusbymember",
    "UserGroupIds": [],
    "Authentication": {
        "Type": "password",
        "PasswordCount": 1
    },
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"
}
]
}

```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.



- Untuk detail API, lihat [DescribeUsers](#) di Referensi AWS CLI Perintah.

## disassociate-global-replication-group

Contoh kode berikut menunjukkan cara menggunakan `disassociate-global-replication-group`.

### AWS CLI

Untuk memisahkan cluster sekunder dari grup replikasi global

`disassociate-global-replication-group` Contoh berikut menghapus cluster sekunder dari datastore Global

```
aws elasticache disassociate-global-replication-group \
  --global-replication-group-id my-grg \
  --replication-group-id my-cluster-grg-secondary \
  --replication-group-region us-east-1
```

Output:

```
{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "my-grg",
    "GlobalReplicationGroupDescription": "my-grg",
    "Status": "modifying",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.6",
    "Members": [
      {
        "ReplicationGroupId": "my-cluster-grg-secondary",
        "ReplicationGroupRegion": "us-east-1",
        "Role": "SECONDARY",
        "AutomaticFailover": "enabled",
        "Status": "associated"
      },
      {
        "ReplicationGroupId": "my-cluster-grg",
        "ReplicationGroupRegion": "us-west-2",
        "Role": "PRIMARY",
        "AutomaticFailover": "enabled",

```

```

        "Status": "associated"
      }
    ],
    "ClusterEnabled": false,
    "AuthTokenEnabled": false,
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}

```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [DisassociateGlobalReplicationGroup](#) di Referensi AWS CLI Perintah.

## increase-node-groups-in-global-replication-group

Contoh kode berikut menunjukkan cara menggunakan `increase-node-groups-in-global-replication-group`.

### AWS CLI

Untuk meningkatkan jumlah grup node dalam grup replikasi global

Berikut ini `increase-node-groups-in-global-replication-group` meningkatkan jumlah kelompok node menggunakan mesin Redis.

```

aws elasticache increase-node-groups-in-global-replication-group \
  --global-replication-group-id sgau-pat-test-4 \
  --node-group-count 6 \
  --apply-immediately

```

### Output:

```

{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "sgau-test-4",
    "GlobalReplicationGroupDescription": "test-4",
    "Status": "modifying",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.6",
    "Members": [

```

```
{
  "ReplicationGroupId": "my-cluster-b",
  "ReplicationGroupRegion": "us-east-1",
  "Role": "SECONDARY",
  "AutomaticFailover": "enabled",
  "Status": "associated"
},
{
  "ReplicationGroupId": "my-cluster-a",
  "ReplicationGroupRegion": "us-west-2",
  "Role": "PRIMARY",
  "AutomaticFailover": "enabled",
  "Status": "associated"
}
],
"ClusterEnabled": true,
"GlobalNodeGroups": [
  {
    "GlobalNodeGroupId": "sgau-test-4-0001",
    "Slots": "0-234,2420-5461"
  },
  {
    "GlobalNodeGroupId": "sgau-test-4-0002",
    "Slots": "5462-5904,6997-9830"
  },
  {
    "GlobalNodeGroupId": "sgau-test-4-0003",
    "Slots": "10923-11190,13375-16383"
  },
  {
    "GlobalNodeGroupId": "sgau-test-4-0004",
    "Slots": "235-2419,5905-6996"
  },
  {
    "GlobalNodeGroupId": "sgau-test-4-0005",
    "Slots": "9831-10922,11191-13374"
  }
],
"AuthTokenEnabled": false,
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [IncreaseNodeGroupsInGlobalReplicationGroup](#) di Referensi AWS CLI Perintah.

## increase-replica-count

Contoh kode berikut menunjukkan cara menggunakan `increase-replica-count`.

### AWS CLI

Untuk meningkatkan jumlah replika

`increase-replica-count` Contoh berikut melakukan salah satu dari dua hal. Ini dapat secara dinamis meningkatkan jumlah replika dalam grup replikasi Redis (mode cluster dinonaktifkan). Atau dapat secara dinamis meningkatkan jumlah node replika dalam satu atau lebih grup node (pecahan) dari grup replikasi Redis (mode cluster diaktifkan). Operasi ini dilakukan tanpa downtime cluster.

```
aws elasticache increase-replica-count \  
  --replication-group-id "my-cluster" \  
  --apply-immediately \  
  --new-replica-count 3
```

Output:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "my-cluster",  
    "Description": " ",  
    "Status": "modifying",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "my-cluster-001",  
      "my-cluster-002",  
      "my-cluster-003",  
      "my-cluster-004"  
    ],  
    "NodeGroups": [  
      {  
        "NodeGroupId": "0001",
```

```
        "Status": "modifying",
        "PrimaryEndpoint": {
            "Address": "my-
cluster.xxxxxih.ng.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "ReaderEndpoint": {
            "Address": "my-cluster-
ro.xxxxxxih.ng.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "NodeGroupMembers": [
            {
                "CacheClusterId": "my-cluster-001",
                "CacheNodeId": "0001",
                "ReadEndpoint": {
                    "Address": "my-
cluster-001.xxxxxih.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "PreferredAvailabilityZone": "us-west-2a",
                "CurrentRole": "primary"
            },
            {
                "CacheClusterId": "my-cluster-003",
                "CacheNodeId": "0001",
                "ReadEndpoint": {
                    "Address": "my-
cluster-003.xxxxxih.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "PreferredAvailabilityZone": "us-west-2a",
                "CurrentRole": "replica"
            }
        ]
    }
},
"AutomaticFailover": "disabled",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.xlarge",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Jumlah Replika dalam Shard](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [IncreaseReplicaCount](#) di Referensi AWS CLI Perintah.

## list-allowed-node-type-modifications

Contoh kode berikut menunjukkan cara menggunakan `list-allowed-node-type-modifications`.

### AWS CLI

Untuk membuat daftar modifikasi node yang diizinkan

`list-allowed-node-type-modifications` Contoh berikut mencantumkan semua tipe node yang tersedia yang dapat Anda gunakan untuk menskalakan tipe node kluster Redis atau grup replikasi Anda saat ini.

```
aws elasticache list-allowed-node-type-modifications \  
  --replication-group-id "my-replication-group"
```

### Output:

```
{  
  "ScaleUpModifications": [  
    "cache.m5.12xlarge",  
    "cache.m5.24xlarge",  
    "cache.m5.4xlarge",  
    "cache.r5.12xlarge",  
    "cache.r5.24xlarge",  
    "cache.r5.2xlarge",  
    "cache.r5.4xlarge"  
  ],  
  "ScaleDownModifications": [  
    "cache.m3.large",  
    "cache.m3.medium",  
    "cache.m3.xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",
```

```
    "cache.m5.2xlarge",
    "cache.m5.large",
    "cache.m5.xlarge",
    "cache.r3.large",
    "cache.r4.large",
    "cache.r4.xlarge",
    "cache.r5.large",
    "cache.t2.medium",
    "cache.t2.micro",
    "cache.t2.small",
    "cache.t3.medium",
    "cache.t3.micro",
    "cache.t3.small"
  ]
}
```

Untuk informasi selengkapnya, lihat [Penskalaan ElastiCache untuk Redis Cluster](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [ListAllowedNodeTypeModifications](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk sumber daya.

```
aws elasticache list-tags-for-resource \
  --resource-name "arn:aws:elasticache:us-east-1:123456789012:cluster:my-cluster"
```

Output:

```
{
  "TagList": [
    {
      "Key": "Project",
      "Value": "querySpeedUp"
    },
    {
```

```

        "Key": "Environment",
        "Value": "PROD"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Daftar Tag Menggunakan AWS CLI](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## modify-cache-cluster

Contoh kode berikut menunjukkan cara menggunakan `modify-cache-cluster`.

### AWS CLI

Untuk memodifikasi cluster cache

`modify-cache-cluster` Contoh berikut memodifikasi pengaturan untuk cluster tertentu.

```

aws elasticache modify-cache-cluster \
  --cache-cluster-id "my-cluster" \
  --num-cache-nodes 1

```

### Output:

```

{
  "CacheCluster": {
    "CacheClusterId": "my-cluster",
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "CacheNodeType": "cache.m5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "CacheClusterStatus": "available",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2c",
    "CacheClusterCreateTime": "2019-12-04T18:24:56.652Z",
    "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
    "PendingModifiedValues": {},
    "CacheSecurityGroups": [],
    "CacheParameterGroup": {

```



```

        "CacheParameterGroupName": "default.redis5.0",
        "ParameterApplyStatus": "in-sync",
        "CacheNodeIdsToReboot": []
    },
    "CacheSubnetGroupName": "default",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "07:00-08:00",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
}
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi ElastiCache Cluster](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [ModifyCacheCluster](#) di Referensi AWS CLI Perintah.

## modify-cache-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `modify-cache-parameter-group`.

### AWS CLI

Untuk memodifikasi grup parameter cache

`modify-cache-parameter-group` Contoh berikut memodifikasi parameter kelompok parameter cache yang ditentukan.

```

aws elasticache modify-cache-parameter-group \
  --cache-parameter-group-name "mygroup" \
  --parameter-name-values "ParameterName=activedefrag, ParameterValue=no"

```

Output:

```

{
  "CacheParameterGroupName": "mygroup"
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Grup Parameter](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [ModifyCacheParameterGroup](#) di Referensi AWS CLI Perintah.

## modify-cache-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `modify-cache-subnet-group`.

### AWS CLI

Untuk memodifikasi grup subnet cache

`modify-cache-subnet-group` Contoh berikut memodifikasi kelompok subnet cache yang ditentukan.

```
aws elasticache modify-cache-subnet-group \  
  --cache-subnet-group-name kxxkx \  
  --cache-subnet-group-description "mygroup"
```

Output:

```
{  
  "CacheSubnetGroup": {  
    "CacheSubnetGroupName": "kxxkx",  
    "CacheSubnetGroupDescription": "mygroup",  
    "VpcId": "vpc-xxxxcdb",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-xxxxbff",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2a"  
        }  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Grup Subnet di Panduan Pengguna Elasticache](#).

- Untuk detail API, lihat [ModifyCacheSubnetGroup](#) di Referensi AWS CLI Perintah.

## modify-global-replication-group

Contoh kode berikut menunjukkan cara menggunakan `modify-global-replication-group`.

## AWS CLI

Untuk memodifikasi grup replikasi global

Berikut ini `modify-global-replication-group` memodifikasi properti grup replikasi global, dalam hal ini menonaktifkan failover otomatis, menggunakan mesin Redis.

```
aws elasticache modify-global-replication-group \  
  --global-replication-group-id sgaui-pat-group \  
  --apply-immediately \  
  --no-automatic-failover-enabled
```

### Output

```
{  
  "GlobalReplicationGroup": {  
    "GlobalReplicationGroupId": "sgaui-test-group",  
    "GlobalReplicationGroupDescription": " ",  
    "Status": "modifying",  
    "CacheNodeType": "cache.r5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.6",  
    "ClusterEnabled": false,  
    "AuthTokenEnabled": false,  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Replikasi Lintas AWS Wilayah Menggunakan Global Datastore](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [ModifyGlobalReplicationGroup](#) di Referensi AWS CLI Perintah.

## **modify-replication-group-shard-configuration**

Contoh kode berikut menunjukkan cara menggunakan `modify-replication-group-shard-configuration`.

### AWS CLI

Untuk memodifikasi konfigurasi pecahan grup replikasi

Berikut ini `modify-replication-group-shard-configuration` mengurangi jumlah kelompok node menggunakan mesin Redis.

```
aws elasticache modify-replication-group-shard-configuration \  
  --replication-group-id mycluster \  
  --node-group-count 3 \  
  --apply-immediately \  
  --node-groups-to-remove 0002
```

## Output

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "mycluster",  
    "Description": "mycluster",  
    "GlobalReplicationGroupInfo": {},  
    "Status": "modifying",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "mycluster-0002-001",  
      "mycluster-0002-002",  
      "mycluster-0002-003",  
      "mycluster-0003-001",  
      "mycluster-0003-002",  
      "mycluster-0003-003",  
      "mycluster-0003-004",  
      "mycluster-0004-001",  
      "mycluster-0004-002",  
      "mycluster-0004-003",  
      "mycluster-0005-001",  
      "mycluster-0005-002",  
      "mycluster-0005-003"  
    ],  
    "NodeGroups": [  
      {  
        "NodeGroupId": "0002",  
        "Status": "modifying",  
        "Slots": "894-1767,3134-4443,5149-5461,6827-7332,12570-13662",  
        "NodeGroupMembers": [  
          {  
            "CacheClusterId": "mycluster-0002-001",  
            "CacheNodeId": "0001",  
            "PreferredAvailabilityZone": "us-west-2c"  
          }  
        ]  
      }  
    ]  
  }  
}
```

```
    },
    {
      "CacheClusterId": "mycluster-0002-002",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2a"
    },
    {
      "CacheClusterId": "mycluster-0002-003",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2b"
    }
  ]
},
{
  "NodeGroupId": "0003",
  "Status": "modifying",
  "Slots":
"0-324,5462-5692,6784-6826,7698-8191,10923-11075,12441-12569,13663-16383",
  "NodeGroupMembers": [
    {
      "CacheClusterId": "mycluster-0003-001",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2c"
    },
    {
      "CacheClusterId": "mycluster-0003-002",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2b"
    },
    {
      "CacheClusterId": "mycluster-0003-003",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2a"
    },
    {
      "CacheClusterId": "mycluster-0003-004",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2c"
    }
  ]
},
{
  "NodeGroupId": "0004",
  "Status": "modifying",
```

```
"Slots": "325-336,4706-5148,7333-7697,9012-10922,11076-12440",
"NodeGroupMembers": [
  {
    "CacheClusterId": "mycluster-0004-001",
    "CacheNodeId": "0001",
    "PreferredAvailabilityZone": "us-west-2b"
  },
  {
    "CacheClusterId": "mycluster-0004-002",
    "CacheNodeId": "0001",
    "PreferredAvailabilityZone": "us-west-2a"
  },
  {
    "CacheClusterId": "mycluster-0004-003",
    "CacheNodeId": "0001",
    "PreferredAvailabilityZone": "us-west-2c"
  }
]
},
{
  "NodeGroupId": "0005",
  "Status": "modifying",
  "Slots": "337-893,1768-3133,4444-4705,5693-6783,8192-9011",
  "NodeGroupMembers": [
    {
      "CacheClusterId": "mycluster-0005-001",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2a"
    },
    {
      "CacheClusterId": "mycluster-0005-002",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2c"
    },
    {
      "CacheClusterId": "mycluster-0005-003",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2b"
    }
  ]
}
],
"AutomaticFailover": "enabled",
"MultiAZ": "enabled",
```

```

    "ConfigurationEndpoint": {
      "Address": "mycluster.g2xbih.clustercfg.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "SnapshotRetentionLimit": 1,
    "SnapshotWindow": "13:00-14:00",
    "ClusterEnabled": true,
    "CacheNodeType": "cache.r5.xlarge",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}

```

Untuk informasi selengkapnya, lihat [Penskalaan ElastiCache untuk Redis Cluster](#) di Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [ModifyReplicationGroupShardConfiguration](#) di Referensi AWS CLI Perintah.

## modify-replication-group

Contoh kode berikut menunjukkan cara menggunakan `modify-replication-group`.

### AWS CLI

Untuk memodifikasi grup replikasi

Berikut ini `modify-replication-group` menonaktifkan Multi-AZ menggunakan mesin Redis.

```

aws elasticache modify-replication-group \
  --replication-group-id test-cluster \
  --no-multi-az-enabled \
  --apply-immediately

```

### Output

```

{
  "ReplicationGroup": {
    "ReplicationGroupId": "test-cluster",
    "Description": "test-cluster",
    "GlobalReplicationGroupInfo": {
      "GlobalReplicationGroupId": "sgaui-pat-group",
      "GlobalReplicationGroupMemberRole": "PRIMARY"
    }
  }
}

```

```
    },
    "Status": "available",
    "PendingModifiedValues": {},
    "MemberClusters": [
      "test-cluster-001",
      "test-cluster-002",
      "test-cluster-003"
    ],
    "NodeGroups": [
      {
        "NodeGroupId": "0001",
        "Status": "available",
        "PrimaryEndpoint": {
          "Address": "test-
cluster.g2xbih.ng.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "ReaderEndpoint": {
          "Address": "test-cluster-
ro.g2xbih.ng.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "NodeGroupMembers": [
          {
            "CacheClusterId": "test-cluster-001",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Address": "test-
cluster-001.g2xbih.0001.usw2.cache.amazonaws.com",
              "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2c",
            "CurrentRole": "primary"
          },
          {
            "CacheClusterId": "test-cluster-002",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Address": "test-
cluster-002.g2xbih.0001.usw2.cache.amazonaws.com",
              "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2b",
            "CurrentRole": "replica"
          }
        ]
      }
    ]
  }
}
```



```

        },
        {
            "CacheClusterId": "test-cluster-003",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
                "Address": "test-
cluster-003.g2xbih.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2a",
            "CurrentRole": "replica"
        }
    ]
}
],
"SnapshottingClusterId": "test-cluster-002",
"AutomaticFailover": "enabled",
"MultiAZ": "disabled",
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "08:00-09:00",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.large",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Grup Replikasi di Panduan Pengguna ElastiCache](#).

- Untuk detail API, lihat [ModifyReplicationGroup](#) di Referensi AWS CLI Perintah.

## modify-user-group

Contoh kode berikut menunjukkan cara menggunakan `modify-user-group`.

### AWS CLI

Untuk memodifikasi grup pengguna

`modify-user-group` Contoh berikut menambahkan pengguna ke grup pengguna.

```
aws elasticache modify-user-group \
```

```
--user-group-id myusergroup \  
--user-ids-to-add user1
```

Output:

```
{  
  "UserGroupId": "myusergroup",  
  "Status": "modifying",  
  "Engine": "redis",  
  "UserIds": [  
    "default"  
  ],  
  "PendingChanges": {  
    "UserIdsToAdd": [  
      "user1"  
    ]  
  },  
  "ReplicationGroups": [],  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx52:usergroup:myusergroup"  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [ModifyUserGroup](#) di Referensi AWS CLI Perintah.

## modify-user

Contoh kode berikut menunjukkan cara menggunakan `modify-user`.

### AWS CLI

Untuk memodifikasi pengguna

`modify-user` Contoh berikut memodifikasi string akses pengguna.

```
aws elasticache modify-user \  
--user-id user2 \  
--append-access-string "on ~* +@all"
```

Output:

```
{
  "UserId": "user2",
  "UserName": "myUser",
  "Status": "modifying",
  "Engine": "redis",
  "AccessString": "on ~* +@all",
  "UserGroupIds": [],
  "Authentication": {
    "Type": "password",
    "PasswordCount": 1
  },
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxx52:user:user2"
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi Pengguna dengan Kontrol Akses Berbasis Peran \(RBAC\)](#) di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [ModifyUser](#) di Referensi AWS CLI Perintah.

## **purchase-reserved-cache-nodes-offering**

Contoh kode berikut menunjukkan cara menggunakan `purchase-reserved-cache-nodes-offering`.

### AWS CLI

Untuk membeli `reserved-cache-node-offering`

`purchase-reserved-cache-nodes-offering` Contoh berikut memungkinkan Anda untuk membeli penawaran node cache yang dicadangkan.

```
aws elasticache purchase-reserved-cache-nodes-offering \
  --reserved-cache-nodes-offering-id xxxxxxx-4da5-4b90-b92d-929fbd7abed2
```

### Output

```
{
  "ReservedCacheNode": {
    "ReservedCacheNodeId": "ri-2020-06-30-17-59-40-474",
    "ReservedCacheNodesOfferingId": "xxxxxxx-4da5-4b90-b92d-929fbd7abed2",
    "CacheNodeType": "cache.m3.2xlarge",
    "StartTime": "2020-06-30T17:59:40.474000+00:00",
```

```

    "Duration": 31536000,
    "FixedPrice": 1772.0,
    "UsagePrice": 0.0,
    "CacheNodeCount": 1,
    "ProductDescription": "redis",
    "OfferingType": "Heavy Utilization",
    "State": "payment-pending",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.25,
        "RecurringChargeFrequency": "Hourly"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Mendapatkan Info Tentang Penawaran Node Cadangan](#) di Panduan Pengguna Elasticache Redis atau [Mendapatkan Info Tentang Penawaran Node Cadangan](#) di Panduan Pengguna [Memcached Elasticache](#).

- Untuk detail API, lihat [PurchaseReservedCacheNodesOffering](#) di Referensi AWS CLI Perintah.

## reboot-cache-cluster

Contoh kode berikut menunjukkan cara menggunakan `reboot-cache-cluster`.

### AWS CLI

Untuk me-reboot cluster cache

`reboot-cache-cluster` Contoh berikut me-reboot beberapa, atau semua, dari node cache dalam cluster yang disediakan. Operasi ini menerapkan setiap kelompok parameter cache yang dimodifikasi ke cluster. Operasi reboot berlangsung sesegera mungkin, dan mengakibatkan pemadaman sesaat ke cluster. Selama reboot, status cluster diatur ke `REBOOTING`.

```

aws elasticache reboot-cache-cluster \
  --cache-cluster-id "my-cluster-001" \
  --cache-node-ids-to-reboot "0001"

```

Output:

```
{
```

```
"CacheCluster": {
  "CacheClusterId": "my-cluster-001",
  "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
  "CacheNodeType": "cache.r5.xlarge",
  "Engine": "redis",
  "EngineVersion": "5.0.5",
  "CacheClusterStatus": "rebooting cache cluster nodes",
  "NumCacheNodes": 1,
  "PreferredAvailabilityZone": "us-west-2a",
  "CacheClusterCreateTime": "2019-11-26T03:35:04.546Z",
  "PreferredMaintenanceWindow": "mon:04:05-mon:05:05",
  "PendingModifiedValues": {},
  "NotificationConfiguration": {
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
    "TopicStatus": "active"
  },
  "CacheSecurityGroups": [],
  "CacheParameterGroup": {
    "CacheParameterGroupName": "mygroup",
    "ParameterApplyStatus": "in-sync",
    "CacheNodeIdsToReboot": []
  },
  "CacheSubnetGroupName": "kxkxk",
  "AutoMinorVersionUpgrade": true,
  "SecurityGroups": [
    {
      "SecurityGroupId": "sg-xxxxxxxxxxxx836",
      "Status": "active"
    },
    {
      "SecurityGroupId": "sg-xxxxxxx7b",
      "Status": "active"
    }
  ],
  "ReplicationGroupId": "my-cluster",
  "SnapshotRetentionLimit": 0,
  "SnapshotWindow": "07:30-08:30",
  "TransitEncryptionEnabled": false,
  "AtRestEncryptionEnabled": false
}
}
```

Untuk informasi selengkapnya, lihat Mem-boot ulang Cluster < <https://docs.aws.amazon.com/AmazonElastiCache/Latest/Red-ug/Clusters.Rebooting.html> di Panduan Pengguna Elasticache.

- Untuk detail API, lihat [RebootCacheCluster](#) di Referensi AWS CLI Perintah.

## reset-cache-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `reset-cache-parameter-group`.

### AWS CLI

Untuk mengatur ulang grup parameter cache

`reset-cache-parameter-group` Contoh berikut memodifikasi parameter kelompok parameter cache ke mesin atau nilai default sistem. Anda dapat mengatur ulang parameter tertentu dengan mengirimkan daftar nama parameter. Untuk mengatur ulang seluruh grup parameter cache, tentukan `--cache-parameter-group-name parameter --reset-all-parameters` dan.

```
aws elasticache reset-cache-parameter-group \  
  --cache-parameter-group-name "mygroup" \  
  --reset-all-parameters
```

Output:

```
{  
  "CacheParameterGroupName": "mygroup"  
}
```

- Untuk detail API, lihat [ResetCacheParameterGroup](#) di Referensi AWS CLI Perintah.

## start-migration

Contoh kode berikut menunjukkan cara menggunakan `start-migration`.

### AWS CLI

Untuk memulai migrasi

Berikut ini `start-migration` memigrasikan data Anda dari Redis yang dihosting sendiri di Amazon EC2 ke Amazon ElastiCache, menggunakan mesin Redis.

```
aws elasticache start-migration \  
  --replication-group-id test \  
  --customer-node-endpoint-list  
  "Address='test.g2xbih.ng.0001.usw2.cache.amazonaws.com',Port=6379"
```

## Output

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "test",  
    "Description": "test",  
    "GlobalReplicationGroupInfo": {},  
    "Status": "modifying",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "test-001",  
      "test-002",  
      "test-003"  
    ],  
    "NodeGroups": [  
      {  
        "NodeGroupId": "0001",  
        "Status": "available",  
        "PrimaryEndpoint": {  
          "Address": "test.g2xbih.ng.0001.usw2.cache.amazonaws.com",  
          "Port": 6379  
        },  
        "ReaderEndpoint": {  
          "Address": "test-ro.g2xbih.ng.0001.usw2.cache.amazonaws.com",  
          "Port": 6379  
        },  
        "NodeGroupMembers": [  
          {  
            "CacheClusterId": "test-001",  
            "CacheNodeId": "0001",  
            "ReadEndpoint": {  
              "Address":  
"test-001.g2xbih.0001.usw2.cache.amazonaws.com",  
              "Port": 6379  
            },  
            "PreferredAvailabilityZone": "us-west-2a",  
            "CurrentRole": "primary"  
          },  
          ],  
        },  
      ],  
    },  
  },  
}
```

```

        {
            "CacheClusterId": "test-002",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
                "Address":
"test-002.g2xbih.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2c",
            "CurrentRole": "replica"
        },
        {
            "CacheClusterId": "test-003",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
                "Address":
"test-003.g2xbih.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2b",
            "CurrentRole": "replica"
        }
    ]
}
],
"SnapshottingClusterId": "test-002",
"AutomaticFailover": "enabled",
"MultiAZ": "enabled",
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.large",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Untuk informasi selengkapnya, lihat [Migrasi Online ke ElastiCache](#) dalam Panduan Pengguna ElastiCache.

- Untuk detail API, lihat [StartMigration](#) di Referensi AWS CLI Perintah.



## test-failover

Contoh kode berikut menunjukkan cara menggunakan `test-failover`.

### AWS CLI

Untuk menguji failover dari grup node

`test-failover` Contoh berikut menguji failover otomatis pada grup node tertentu (disebut pecahan di konsol) dalam grup replikasi (disebut cluster di konsol).

```
aws elasticache test-failover /
  --replication-group-id "mycluster" /
  --node-group-id "0001"
```

Output:

```
{
  "ReplicationGroup": {
    "ReplicationGroupId": "mycluster",
    "Description": "My Cluster",
    "Status": "available",
    "PendingModifiedValues": {},
    "MemberClusters": [
      "mycluster-0001-001",
      "mycluster-0001-002",
      "mycluster-0001-003",
      "mycluster-0002-001",
      "mycluster-0002-002",
      "mycluster-0002-003",
      "mycluster-0003-001",
      "mycluster-0003-002",
      "mycluster-0003-003"
    ],
    "NodeGroups": [
      {
        "NodeGroupId": "0001",
        "Status": "available",
        "Slots": "0-5461",
        "NodeGroupMembers": [
          {
            "CacheClusterId": "mycluster-0001-001",
            "CacheNodeId": "0001",
```

```
        "PreferredAvailabilityZone": "us-west-2b"
    },
    {
        "CacheClusterId": "mycluster-0001-002",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2a"
    },
    {
        "CacheClusterId": "mycluster-0001-003",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2c"
    }
]
},
{
    "NodeGroupId": "0002",
    "Status": "available",
    "Slots": "5462-10922",
    "NodeGroupMembers": [
        {
            "CacheClusterId": "mycluster-0002-001",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2a"
        },
        {
            "CacheClusterId": "mycluster-0002-002",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2b"
        },
        {
            "CacheClusterId": "mycluster-0002-003",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2c"
        }
    ]
},
{
    "NodeGroupId": "0003",
    "Status": "available",
    "Slots": "10923-16383",
    "NodeGroupMembers": [
        {
            "CacheClusterId": "mycluster-0003-001",
            "CacheNodeId": "0001",
```

```

        "PreferredAvailabilityZone": "us-west-2c"
    },
    {
        "CacheClusterId": "mycluster-0003-002",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2b"
    },
    {
        "CacheClusterId": "mycluster-0003-003",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2a"
    }
]
}
],
"AutomaticFailover": "enabled",
"ConfigurationEndpoint": {
    "Address": "mycluster.xxxxih.clustercfg.usw2.cache.amazonaws.com",
    "Port": 6379
},
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "13:00-14:00",
"ClusterEnabled": true,
"CacheNodeType": "cache.r5.large",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

- Untuk detail API, lihat [TestFailover](#) di Referensi AWS CLI Perintah.

## MediaStore contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with MediaStore.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **create-container**

Contoh kode berikut menunjukkan cara menggunakan `create-container`.

AWS CLI

Untuk membuat wadah

`create-container` Contoh berikut membuat wadah baru yang kosong.

```
aws mediastore create-container --container-name ExampleContainer
```

Output:

```
{
  "Container": {
    "AccessLoggingEnabled": false,
    "CreationTime": 1563557265,
    "Name": "ExampleContainer",
    "Status": "CREATING",
    "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Kontainer](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateContainer](#) di Referensi AWS CLI Perintah.

## **delete-container-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-container-policy`.

## AWS CLI

Untuk menghapus kebijakan kontainer

`delete-container-policy` Contoh berikut menghapus kebijakan yang ditetapkan ke kontainer yang ditentukan. Ketika kebijakan dihapus, AWS Elemental MediaStore secara otomatis menetapkan kebijakan default ke container.

```
aws mediastore delete-container-policy \  
  --container-name LiveEvents
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteContainerPolicy](#) di referensi AWS Elemental MediaStore API.

- Untuk detail API, lihat [DeleteContainerPolicy](#) di Referensi AWS CLI Perintah.

### **delete-container**

Contoh kode berikut menunjukkan cara menggunakan `delete-container`.

## AWS CLI

Untuk menghapus wadah

`delete-container` Contoh berikut menghapus wadah yang ditentukan. Anda dapat menghapus wadah hanya jika tidak memiliki objek.

```
aws mediastore delete-container \  
  --container-name=ExampleLiveDemo
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Kontainer](#) di MediaStore Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeleteContainer](#) di Referensi AWS CLI Perintah.

### **delete-cors-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-cors-policy`.

## AWS CLI

Untuk menghapus kebijakan CORS

`delete-cors-policy` Contoh berikut menghapus kebijakan cross-origin resource sharing (CORS) yang ditetapkan ke container yang ditentukan.

```
aws mediastore delete-cors-policy \  
  --container-name ExampleContainer
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Kebijakan CORS di Panduan Pengguna AWS Elemental MediaStore](#).

- Untuk detail API, lihat [DeleteCorsPolicy](#) di Referensi AWS CLI Perintah.

## **delete-lifecycle-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-lifecycle-policy`.

## AWS CLI

Untuk menghapus kebijakan siklus hidup objek

`delete-lifecycle-policy` Contoh berikut menghapus kebijakan siklus hidup objek yang dilampirkan ke wadah yang ditentukan. Perubahan ini dapat memakan waktu hingga 20 menit untuk diterapkan.

```
aws mediastore delete-lifecycle-policy \  
  --container-name LiveEvents
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Kebijakan Siklus Hidup Objek di Panduan Pengguna AWS Elemental MediaStore](#).

- Untuk detail API, lihat [DeleteLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## **describe-container**

Contoh kode berikut menunjukkan cara menggunakan `describe-container`.

## AWS CLI

Untuk melihat detail kontainer

`describe-container` Contoh berikut menampilkan rincian wadah yang ditentukan.

```
aws mediastore describe-container \  
  --container-name ExampleContainer
```

Output:

```
{  
  "Container": {  
    "CreationTime": 1563558086,  
    "AccessLoggingEnabled": false,  
    "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/  
ExampleContainer",  
    "Status": "ACTIVE",  
    "Name": "ExampleContainer",  
    "Endpoint": "https://aaabbbcccddee.data.mediastore.us-west-2.amazonaws.com"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail untuk Kontainer](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribeContainer](#) di Referensi AWS CLI Perintah.

## **describe-object**

Contoh kode berikut menunjukkan cara menggunakan `describe-object`.

## AWS CLI

Untuk melihat daftar objek dan folder dalam wadah tertentu

`describe-object` Contoh berikut menampilkan item (objek dan folder) yang disimpan dalam wadah tertentu.

```
aws mediastore-data describe-object \  
  --endpoint https://aaabbbcccddee.data.mediastore.us-west-2.amazonaws.com \  
  --path /folder_name/file1234.jpg
```

## Output:

```
{
  "ContentType": "image/jpeg",
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
  "ContentLength": "2307346",
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeeee4dd89ff7f555555555555da6d3"
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Objek](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribeObject](#) di Referensi AWS CLI Perintah.

**get-container-policy**

Contoh kode berikut menunjukkan cara menggunakan `get-container-policy`.

## AWS CLI

Untuk melihat kebijakan kontainer

`get-container-policy` Contoh berikut menampilkan kebijakan berbasis sumber daya dari wadah yang ditentukan.

```
aws mediastore get-container-policy \
  --container-name ExampleLiveDemo
```

## Output:

```
{
  "Policy": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "PublicReadOverHttps",
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::111122223333:root"
        },
        "Action": [
          "mediastore:GetObject",

```



```
        "mediastore:DescribeObject"
      ],
      "Resource": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo/",
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
}
```

Untuk informasi selengkapnya, lihat [Melihat Kebijakan Penampung](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [GetContainerPolicy](#) di Referensi AWS CLI Perintah.

## get-cors-policy

Contoh kode berikut menunjukkan cara menggunakan `get-cors-policy`.

### AWS CLI

Untuk melihat kebijakan CORS

`get-cors-policy` Contoh berikut menampilkan kebijakan cross-origin resource sharing (CORS) yang ditetapkan ke container yang ditentukan.

```
aws mediastore get-cors-policy \
  --container-name ExampleContainer \
  --region us-west-2
```

Output:

```
{
  "CorsPolicy": [
    {
      "AllowedMethods": [
        "GET",
        "HEAD"
      ]
    }
  ]
}
```

```

    ],
    "MaxAgeSeconds": 3000,
    "AllowedOrigins": [
        ""
    ],
    "AllowedHeaders": [
        ""
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat Kebijakan CORS](#) di MediaStore Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [GetCorsPolicy](#) di Referensi AWS CLI Perintah.

## get-lifecycle-policy

Contoh kode berikut menunjukkan cara menggunakan `get-lifecycle-policy`.

### AWS CLI

Untuk melihat kebijakan siklus hidup objek

`get-lifecycle-policy` Contoh berikut menampilkan kebijakan siklus hidup objek yang dilampirkan ke wadah yang ditentukan.

```
aws mediastore get-lifecycle-policy \
  --container-name LiveEvents
```

Output:

```
{
  "LifecyclePolicy": {
    "rules": [
      {
        "definition": {
          "path": [
            {
              "prefix": "Football/"
            }
          ],

```

```

        {
            "prefix": "Baseball/"
        }
    ],
    "days_since_create": [
        {
            "numeric": [
                ">",
                28
            ]
        }
    ]
},
"action": "EXPIRE"
}
]
}
}
}

```

Untuk informasi selengkapnya, lihat [Melihat Kebijakan Siklus Hidup Objek di Panduan Pengguna AWS Elemental MediaStore](#) .

- Untuk detail API, lihat [GetLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## get-object

Contoh kode berikut menunjukkan cara menggunakan `get-object`.

### AWS CLI

Untuk mengunduh objek

`get-object` Contoh berikut men-download objek ke endpoint tertentu.

```

aws mediastore-data get-object \
  --endpoint https://aaabbbcccddee.data.mediastore.us-west-2.amazonaws.com \
  --path=/folder_name/README.md README.md

```

Output:

```

{
  "ContentLength": "2307346",
  "ContentType": "image/jpeg",

```

```
"LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
"ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e4dd89ff7f5555555555555555da6d3",  
"StatusCode": 200  
}
```

Untuk mengunduh bagian dari suatu objek

`get-object` Contoh berikut mendownload sebagian objek ke titik akhir yang ditentukan.

```
aws mediastore-data get-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path /folder_name/README.md \  
  --range="bytes=0-100" README2.md
```

Output:

```
{  
  "StatusCode": 206,  
  "ContentRange": "bytes 0-100/2307346",  
  "ContentLength": "101",  
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
  "ContentType": "image/jpeg",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e4dd89ff7f5555555555555555da6d3"  
}
```

Untuk informasi selengkapnya, lihat [Mengunduh Objek](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [GetObject](#) di Referensi AWS CLI Perintah.

## list-containers

Contoh kode berikut menunjukkan cara menggunakan `list-containers`.

### AWS CLI

Untuk melihat daftar kontainer

`list-containers` Contoh berikut menampilkan daftar semua kontainer yang terkait dengan akun Anda.

```
aws mediastore list-containers
```

**Output:**

```
{
  "Containers": [
    {
      "CreationTime": 1505317931,
      "Endpoint": "https://aaabbbcccddee.data.mediastore.us-
west-2.amazonaws.com",
      "Status": "ACTIVE",
      "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo",
      "AccessLoggingEnabled": false,
      "Name": "ExampleLiveDemo"
    },
    {
      "CreationTime": 1506528818,
      "Endpoint": "https://fffghghhhiiijj.data.mediastore.us-
west-2.amazonaws.com",
      "Status": "ACTIVE",
      "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
      "AccessLoggingEnabled": false,
      "Name": "ExampleContainer"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Daftar Kontainer](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [ListContainers](#) di Referensi AWS CLI Perintah.

**list-items**

Contoh kode berikut menunjukkan cara menggunakan `list-items`.

**AWS CLI**

Contoh 1: Untuk melihat daftar objek dan folder dalam wadah tertentu

`list-items` Contoh berikut menampilkan item (objek dan folder) yang disimpan dalam wadah yang ditentukan.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com
```

**Output:**

```
{  
  "Items": [  
    {  
      "ContentType": "image/jpeg",  
      "LastModified": 1563571859.379,  
      "Name": "filename.jpg",  
      "Type": "OBJECT",  
      "ETag":  
"543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",  
      "ContentLength": 3784  
    },  
    {  
      "Type": "FOLDER",  
      "Name": "ExampleLiveDemo"  
    }  
  ]  
}
```

Contoh 2: Untuk melihat daftar objek dan folder dalam folder tertentu

`list-items` Contoh berikut menampilkan item (objek dan folder) yang disimpan dalam folder tertentu.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com
```

**Output:**

```
{  
  "Items": [  
    {  
      "ContentType": "image/jpeg",  
      "LastModified": 1563571859.379,  
      "Name": "filename.jpg",  
      "Type": "OBJECT",  
      "ETag":  
"543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",  
      "ContentLength": 3784  
    }  
  ]  
}
```

```
        "ContentLength": 3784
      },
      {
        "Type": "FOLDER",
        "Name": "ExampleLiveDemo"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat Daftar Objek](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [ListItems](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag untuk wadah

`list-tags-for-resource` Contoh berikut menampilkan kunci tag dan nilai-nilai yang ditetapkan untuk wadah tertentu.

```
aws mediastore list-tags-for-resource \
  --resource arn:aws:mediastore:us-west-2:1213456789012:container/ExampleContainer
```

Output:

```
{
  "Tags": [
    {
      "Value": "Test",
      "Key": "Environment"
    },
    {
      "Value": "West",
      "Key": "Region"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ListTagsForResource](#) di Referensi MediaStore API AWS Elemental.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **put-container-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-container-policy`.

### AWS CLI

Untuk mengedit kebijakan kontainer

`put-container-policy` Contoh berikut menetapkan kebijakan yang berbeda untuk kontainer yang ditentukan. Dalam contoh ini, kebijakan yang diperbarui didefinisikan dalam file bernama `LiveEventsContainerPolicy.json`.

```
aws mediastore put-container-policy \  
  --container-name LiveEvents \  
  --policy file://LiveEventsContainerPolicy.json
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengedit Kebijakan Penampung](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [PutContainerPolicy](#) di Referensi AWS CLI Perintah.

## **put-cors-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-cors-policy`.

### AWS CLI

Contoh 1: Untuk menambahkan kebijakan CORS

`put-cors-policy` Contoh berikut menambahkan kebijakan cross-origin resource sharing (CORS) ke kontainer yang ditentukan. Isi kebijakan CORS ada di file bernama `corsPolicy.json`.

```
aws mediastore put-cors-policy \  
  --container-name ExampleContainer \  
  --policy file://corsPolicy.json
```



```
--cors-policy file://corsPolicy.json
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan Kebijakan CORS ke Container](#) di MediaStore Panduan Pengguna AWS Elemental.

Contoh 2: Untuk mengedit kebijakan CORS

`put-cors-policy` Contoh berikut memperbarui kebijakan cross-origin resource sharing (CORS) yang ditetapkan ke container yang ditentukan. Isi kebijakan CORS yang diperbarui ada di file bernama `corsPolicy2.json`.

Untuk informasi selengkapnya, lihat [Mengedit Kebijakan CORS](#) di MediaStore Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [PutCorsPolicy](#) di Referensi AWS CLI Perintah.

## **put-lifecycle-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-lifecycle-policy`.

### AWS CLI

Untuk membuat kebijakan siklus hidup objek

`put-lifecycle-policy` Contoh berikut melampirkan kebijakan siklus hidup objek ke wadah yang ditentukan. Ini memungkinkan Anda untuk menentukan berapa lama layanan harus menyimpan objek dalam wadah Anda. MediaStore menghapus objek dalam wadah setelah mereka mencapai tanggal kedaluwarsa, seperti yang ditunjukkan dalam kebijakan, yang ada dalam file bernama `LiveEventsLifecyclePolicy.json`

```
aws mediastore put-lifecycle-policy \  
  --container-name ExampleContainer \  
  --lifecycle-policy file://ExampleLifecyclePolicy.json
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan Kebijakan Siklus Hidup Objek ke Penampung di Panduan](#) Pengguna AWS Elemental MediaStore .

- Untuk detail API, lihat [PutLifecyclePolicy](#) di Referensi AWS CLI Perintah.

## put-object

Contoh kode berikut menunjukkan cara menggunakan `put-object`.

### AWS CLI

Untuk mengunggah objek

`put-object` Contoh berikut mengunggah objek ke wadah tertentu. Anda dapat menentukan jalur folder tempat objek akan disimpan di dalam wadah. Jika folder sudah ada, AWS Elemental MediaStore menyimpan objek di folder. Jika folder tidak ada, layanan membuatnya, dan kemudian menyimpan objek di folder.

```
aws mediastore-data put-object \  
  --endpoint https://aaabbbcccddee.data.mediastore.us-west-2.amazonaws.com \  
  --body README.md \  
  --path /folder_name/README.md \  
  --cache-control "max-age=6, public" \  
  --content-type binary/octet-stream
```

Output:

```
{  
  "ContentSHA256":  
    "74b5fdb517f423ed750ef214c44adfe2be36e37d861eafe9c842cbe1bf387a9d",  
  "StorageClass": "TEMPORAL",  
  "ETag": "af3e4731af032167a106015d1f2fe934e68b32ed1aa297a9e325f5c64979277b"  
}
```

Untuk informasi selengkapnya, lihat [Mengunggah Objek](#) di MediaStore Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [PutObject](#) di Referensi AWS CLI Perintah.

## start-access-logging

Contoh kode berikut menunjukkan cara menggunakan `start-access-logging`.

### AWS CLI

Untuk mengaktifkan akses masuk pada kontainer

`start-access-logging` Contoh berikut mengaktifkan akses logging pada wadah yang ditentukan.

```
aws mediastore start-access-logging \  
  --container-name LiveEvents
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Pencatatan Akses untuk Penampung](#) di MediaStore Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [StartAccessLogging](#) di Referensi AWS CLI Perintah.

## **stop-access-logging**

Contoh kode berikut menunjukkan cara menggunakan `stop-access-logging`.

### AWS CLI

Untuk menonaktifkan akses masuk pada wadah

`stop-access-logging` Contoh berikut menonaktifkan akses logging pada wadah yang ditentukan.

```
aws mediastore stop-access-logging \  
  --container-name LiveEvents
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan Pencatatan Akses untuk Kontainer](#) di Panduan Pengguna AWS Elemental MediaStore .

- Untuk detail API, lihat [StopAccessLogging](#) di Referensi AWS CLI Perintah.

## **tag-resource**

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke wadah

`tag-resource` Contoh berikut menambahkan kunci tag dan nilai ke wadah yang ditentukan.

```
aws mediastore tag-resource \  
  --resource arn:aws:mediastore:us-west-2:123456789012:container/ExampleContainer \  
  --tags '[{"Key": "Region", "Value": "West"}, {"Key": "Environment", "Value": "Test"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [TagResource](#) di Referensi MediaStore API AWS Elemental.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari wadah

`untag-resource` Contoh berikut menghapus kunci tag yang ditentukan dan nilai yang terkait dari wadah.

```
aws mediastore untag-resource \  
  --resource arn:aws:mediastore:us-west-2:123456789012:container/ExampleContainer \  
  --tag-keys Region
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UntagResource](#) di Referensi MediaStore API AWS Elemental. .

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Amazon EMR contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan EMR AWS Command Line Interface dengan Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-instance-fleet**

Contoh kode berikut menunjukkan cara menggunakan `add-instance-fleet`.

AWS CLI

Untuk menambahkan armada instance tugas ke cluster

Contoh ini menambahkan armada instance tugas baru ke cluster yang ditentukan.

Perintah:

```
aws emr add-instance-fleet --cluster-id 'j-12ABCDEFGH134JK' --instance-fleet
InstanceFleetType=TASK,TargetSpotCapacity=1,LaunchSpecifications={SpotSpecification={Timeo
```

Output:

```
{
  "ClusterId": "j-12ABCDEFGH134JK",
  "InstanceFleetId": "if-23ABCDEFGH145JJ"
}
```

- Untuk detail API, lihat [AddInstanceFleet](#) di Referensi AWS CLI Perintah.

## add-steps

Contoh kode berikut menunjukkan cara menggunakan `add-steps`.

### AWS CLI

#### 1. Untuk menambahkan langkah-langkah Custom JAR ke cluster

Perintah:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps
  Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://mybucket/
mytest.jar,Args=arg1,arg2,arg3
  Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://mybucket/
mytest.jar,MainClass=mymainclass,Args=arg1,arg2,arg3
```

Parameter yang diperlukan:

Jar

Parameter opsional:

Type, Name, ActionOnFailure, Args

Output:

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

#### 2. Untuk menambahkan langkah Streaming ke cluster

Perintah:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=STREAMING,Name='Streaming
Program',ActionOnFailure=CONTINUE,Args=[-files,s3://elasticmapreduce/samples/
wordcount/wordSplitter.py,-mapper,wordSplitter.py,-reducer,aggregate,-input,s3://
elasticmapreduce/samples/wordcount/input,-output,s3://mybucket/wordcount/output]
```

**Parameter yang diperlukan:**

```
Type, Args
```

**Parameter opsional:**

```
Name, ActionOnFailure
```

**Setara JSON (isi step.json):**

```
[
  {
    "Name": "JSON Streaming Step",
    "Args": ["-files","s3://elasticmapreduce/samples/wordcount/wordSplitter.py","-mapper","wordSplitter.py","-reducer","aggregate","-input","s3://elasticmapreduce/samples/wordcount/input","-output","s3://mybucket/wordcount/output"],
    "ActionOnFailure": "CONTINUE",
    "Type": "STREAMING"
  }
]
```

**CATATAN:** Argumen JSON harus menyertakan opsi dan nilai sebagai item mereka sendiri dalam daftar.

**Perintah (menggunakan step.json):**

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps file://./step.json
```

**Output:**

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

**3. Untuk menambahkan langkah Streaming dengan beberapa file ke cluster (hanya JSON)****JSON (multiplefiles.json):**

```
[
  {
    "Name": "JSON Streaming Step",
    "Type": "STREAMING",
    "ActionOnFailure": "CONTINUE",
    "Args": [
      "-files",
      "s3://mybucket/mapper.py,s3://mybucket/reducer.py",
      "-mapper",
      "mapper.py",
      "-reducer",
      "reducer.py",
      "-input",
      "s3://mybucket/input",
      "-output",
      "s3://mybucket/output"]
  }
]
```

#### Perintah:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps file://./multiplefiles.json
```

#### Parameter yang diperlukan:

Type, Args

#### Parameter opsional:

Name, ActionOnFailure

#### Output:

```
{
  "StepIds":[
    "s-XXXXXXXX",
  ]
}
```

## 4. Untuk menambahkan langkah Hive ke cluster



**Perintah:**

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=HIVE,Name='Hive
program',ActionOnFailure=CONTINUE,Args=[-f,s3://mybucket/myhivescript.q,-
d,INPUT=s3://mybucket/myhiveinput,-d,OUTPUT=s3://mybucket/myhiveoutput,arg1,arg2]
Type=HIVE,Name='Hive steps',ActionOnFailure=TERMINATE_CLUSTER,Args=[-
f,s3://elasticmapreduce/samples/hive-ads/libs/model-build.q,-d,INPUT=s3://
elasticmapreduce/samples/hive-ads/tables,-d,OUTPUT=s3://mybucket/hive-ads/
output/2014-04-18/11-07-32,-d,LIBS=s3://elasticmapreduce/samples/hive-ads/libs]
```

**Parameter yang diperlukan:**

Type, Args

**Parameter opsional:**

Name, ActionOnFailure

**Output:**

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

**5. Untuk menambahkan langkah Babi ke cluster****Perintah:**

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=PIG,Name='Pig
program',ActionOnFailure=CONTINUE,Args=[-f,s3://mybucket/mypigscript.pig,-
p,INPUT=s3://mybucket/mypiginput,-p,OUTPUT=s3://mybucket/mypigoutput,arg1,arg2]
Type=PIG,Name='Pig program',Args=[-f,s3://elasticmapreduce/samples/pig-apache/do-
reports2.pig,-p,INPUT=s3://elasticmapreduce/samples/pig-apache/input,-p,OUTPUT=s3://
mybucket/pig-apache/output,arg1,arg2]
```

**Parameter yang diperlukan:**

```
Type, Args
```

Parameter opsional:

```
Name, ActionOnFailure
```

Output:

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

## 6. Untuk menambahkan langkah Impala ke cluster

Perintah:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=IMPALA,Name='Impala
program',ActionOnFailure=CONTINUE,Args=--impala-script,s3://myimpala/input,--
console-output-path,s3://myimpala/output
```

Parameter yang diperlukan:

```
Type, Args
```

Parameter opsional:

```
Name, ActionOnFailure
```

Output:

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

```
}
```

- Untuk detail API, lihat [AddSteps](#) di Referensi AWS CLI Perintah.

## add-tags

Contoh kode berikut menunjukkan cara menggunakan `add-tags`.

### AWS CLI

1. Untuk menambahkan tag ke cluster

Perintah:

```
aws emr add-tags --resource-id j-xxxxxxx --tags name="John Doe" age=29 sex=male  
address="123 East NW Seattle"
```

Output:

```
None
```

2. Untuk daftar tag dari sebuah cluster

--Perintah:

```
aws emr describe-cluster --cluster-id j-XXXXXXYY --query Cluster.Tags
```

Output:

```
[  
  {  
    "Value": "male",  
    "Key": "sex"  
  },  
  {  
    "Value": "123 East NW Seattle",  
    "Key": "address"  
  },  
  {  
    "Value": "John Doe",
```

```
    "Key": "name"
  },
  {
    "Value": "29",
    "Key": "age"
  }
]
```

- Untuk detail API, lihat [AddTags](#) di Referensi AWS CLI Perintah.

## create-cluster-examples

Contoh kode berikut menunjukkan cara menggunakan `create-cluster-examples`.

### AWS CLI

Sebagian besar contoh berikut mengasumsikan bahwa Anda menentukan peran layanan EMR Amazon dan profil instans Amazon EC2. Jika Anda belum melakukan ini, Anda harus menentukan setiap peran IAM yang diperlukan atau menggunakan `--use-default-roles` parameter saat membuat cluster Anda. Untuk informasi selengkapnya tentang menentukan peran IAM, lihat [Mengonfigurasi Peran IAM untuk Izin EMR Amazon ke Layanan di AWS](#) Panduan Manajemen EMR Amazon.

Contoh 1: Untuk membuat cluster

`create-cluster` Contoh berikut membuat cluster EMR sederhana.

```
aws emr create-cluster \  
  --release-label emr-5.14.0 \  
  --instance-type m4.large \  
  --instance-count 2
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk membuat klaster EMR Amazon dengan default ServiceRole dan peran InstanceProfile

`create-cluster` Contoh berikut membuat cluster EMR Amazon yang menggunakan konfigurasi `--instance-groups`

```
aws emr create-cluster \  
  --instance-groups
```

```
--release-label emr-5.14.0 \
--service-role EMR_DefaultRole \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

Contoh 3: Untuk membuat kluster EMR Amazon yang menggunakan armada instans

`create-cluster` Contoh berikut membuat kluster EMR Amazon yang menggunakan `--instance-fleets` konfigurasi, menentukan dua jenis instans untuk setiap armada dan dua Subnet EC2.

```
aws emr create-cluster \
  --release-label emr-5.14.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,SubnetIds=['subnet-
ab12345c','subnet-de67890f'] \
  --instance-fleets
InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m4.large,
InstanceFleetType=CORE,TargetSpotCapacity=11,InstanceTypeConfigs=['{InstanceType=m4.large,B
```

Contoh 4: Untuk membuat cluster dengan peran default

`create-cluster` Contoh berikut menggunakan `--use-default-roles` parameter untuk menentukan peran layanan default dan profil instance.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --use-default-roles \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
  --auto-terminate
```

Contoh 5: Untuk membuat cluster dan menentukan aplikasi yang akan diinstal

`create-cluster` Contoh berikut menggunakan `--applications` parameter untuk menentukan aplikasi yang Amazon EMR menginstal. Contoh ini menginstal Hadoop, Hive dan Pig.

```
aws emr create-cluster \
  --applications Name=Hadoop Name=Hive Name=Pig \
```

```
--release-label emr-5.9.0 \  
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \  
--auto-terminate
```

Contoh 6: Untuk membuat cluster yang menyertakan Spark

Contoh berikut menginstal Spark.

```
aws emr create-cluster \  
--release-label emr-5.9.0 \  
--applications Name=Spark \  
--ec2-attributes KeyName=myKey \  
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \  
--auto-terminate
```

Contoh 7: Untuk menentukan AMI kustom yang akan digunakan untuk instance cluster

`create-cluster` Contoh berikut membuat instance cluster berdasarkan Amazon Linux AMI dengan ID `ami-a518e6df`.

```
aws emr create-cluster \  
--name "Cluster with My Custom AMI" \  
--custom-ami-id ami-a518e6df \  
--ebs-root-volume-size 20 \  
--release-label emr-5.9.0 \  
--use-default-roles \  
--instance-count 2 \  
--instance-type m4.large
```

Contoh 8: Untuk menyesuaikan konfigurasi aplikasi

Contoh berikut menggunakan `--configurations` parameter untuk menentukan file konfigurasi JSON yang berisi kustomisasi aplikasi untuk Hadoop. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Aplikasi](#) dalam Panduan Rilis Amazon EMR.

Isi dari `configurations.json`:

```
[  
  {
```

```

    "Classification": "mapred-site",
    "Properties": {
      "mapred.tasktracker.map.tasks.maximum": 2
    }
  },
  {
    "Classification": "hadoop-env",
    "Properties": {},
    "Configurations": [
      {
        "Classification": "export",
        "Properties": {
          "HADOOP_DATANODE_HEAPSIZE": 2048,
          "HADOOP_NAMENODE_OPTS": "-XX:GCTimeRatio=19"
        }
      }
    ]
  }
]

```

Contoh referensi berikut `configurations.json` sebagai file lokal.

```

aws emr create-cluster \
  --configurations file://configurations.json \
  --release-label emr-5.9.0 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
  InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
  --auto-terminate

```

Contoh referensi berikut `configurations.json` sebagai file di Amazon S3.

```

aws emr create-cluster \
  --configurations https://s3.amazonaws.com/myBucket/configurations.json \
  --release-label emr-5.9.0 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
  InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
  --auto-terminate

```

Contoh 9: Untuk membuat cluster dengan master, inti, dan grup instance tugas

`create-cluster` Contoh berikut digunakan `--instance-groups` untuk menentukan jenis dan jumlah instans EC2 yang akan digunakan untuk kelompok master, inti, dan instance tugas.

```
aws emr create-cluster \  
  --release-label emr-5.9.0 \  
  --instance-groups  
Name=Master,InstanceGroupType=MASTER,InstanceType=m4.large,InstanceCount=1  
Name=Core,InstanceGroupType=CORE,InstanceType=m4.large,InstanceCount=2  
Name=Task,InstanceGroupType=TASK,InstanceType=m4.large,InstanceCount=2
```

Contoh 10: Untuk menentukan bahwa cluster harus dihentikan setelah menyelesaikan semua langkah

`create-cluster` Contoh berikut digunakan `--auto-terminate` untuk menentukan bahwa cluster harus dimatikan secara otomatis setelah menyelesaikan semua langkah.

```
aws emr create-cluster \  
  --release-label emr-5.9.0 \  
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \  
  --auto-terminate
```

Contoh 11: Untuk menentukan detail konfigurasi cluster seperti key pair Amazon EC2, konfigurasi jaringan, dan grup keamanan

`create-cluster` Contoh berikut membuat cluster dengan key pair Amazon EC2 bernama `myKey` dan profil instance khusus bernama `myProfile`. Pasangan kunci digunakan untuk mengotorisasi koneksi SSH ke node cluster, paling sering node master. Untuk informasi selengkapnya, lihat [Menggunakan Pasangan Kunci Amazon EC2 untuk Kredensial SSH](#) di Panduan Manajemen EMR Amazon.

```
aws emr create-cluster \  
  --ec2-attributes KeyName=myKey,InstanceProfile=myProfile \  
  --release-label emr-5.9.0 \  
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \  
  --auto-terminate
```

Contoh berikut membuat cluster di subnet Amazon VPC.

```
aws emr create-cluster \  
  --ec2-attributes SubnetId=subnet-xxxxx \  
  --auto-terminate
```



```
--release-label emr-5.9.0 \  
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \  
--auto-terminate
```

Contoh berikut membuat cluster di zona us-east-1b ketersediaan.

```
aws emr create-cluster \  
--ec2-attributes AvailabilityZone=us-east-1b \  
--release-label emr-5.9.0 \  
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

Contoh berikut membuat cluster dan hanya menentukan grup keamanan Amazon EMR-managed.

```
aws emr create-cluster \  
--release-label emr-5.9.0 \  
--service-role myServiceRole \  
--ec2-attributes InstanceProfile=myRole,EmrManagedMasterSecurityGroup=sg-  
master1,EmrManagedSlaveSecurityGroup=sg-slave1 \  
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

Contoh berikut membuat cluster dan hanya menentukan grup keamanan Amazon EC2 tambahan.

```
aws emr create-cluster \  
--release-label emr-5.9.0 \  
--service-role myServiceRole \  
--ec2-attributes InstanceProfile=myRole,AdditionalMasterSecurityGroups=[sg-  
addMaster1,sg-addMaster2,sg-addMaster3,sg-  
addMaster4],AdditionalSlaveSecurityGroups=[sg-addSlave1,sg-addSlave2,sg-  
addSlave3,sg-addSlave4] \  
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

Contoh berikut membuat cluster dan menentukan grup keamanan yang dikelola EMR, serta grup keamanan tambahan.

```
aws emr create-cluster \  
--release-label emr-5.9.0 \  
--service-role myServiceRole \  
--service-role myServiceRole \  
--service-role myServiceRole
```

```

--ec2-attributes InstanceProfile=myRole,EmrManagedMasterSecurityGroup=sg-
master1,EmrManagedSlaveSecurityGroup=sg-slave1,AdditionalMasterSecurityGroups=[sg-
addMaster1,sg-addMaster2,sg-addMaster3,sg-
addMaster4],AdditionalSlaveSecurityGroups=[sg-addSlave1,sg-addSlave2,sg-
addSlave3,sg-addSlave4] \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large

```

Contoh berikut membuat cluster di subnet pribadi VPC dan menggunakan grup keamanan Amazon EC2 tertentu untuk mengaktifkan akses layanan Amazon EMR, yang diperlukan untuk cluster di subnet pribadi.

```

aws emr create-cluster \
--release-label emr-5.9.0 \
--service-role myServiceRole \
--ec2-attributes InstanceProfile=myRole,ServiceAccessSecurityGroup=sg-service-
access,EmrManagedMasterSecurityGroup=sg-master,EmrManagedSlaveSecurityGroup=sg-slave
\
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large

```

Contoh berikut menentukan parameter konfigurasi grup keamanan menggunakan file JSON bernama `ec2_attributes.json` yang disimpan secara lokal. CATATAN: Argumen JSON harus menyertakan opsi dan nilai sebagai item mereka sendiri dalam daftar.

```

aws emr create-cluster \
--release-label emr-5.9.0 \
--service-role myServiceRole \
--ec2-attributes file://ec2_attributes.json \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large

```

Isi dari `ec2_attributes.json`:

```

[
  {
    "SubnetId": "subnet-xxxxx",
    "KeyName": "myKey",
    "InstanceProfile": "myRole",
    "EmrManagedMasterSecurityGroup": "sg-master1",
    "EmrManagedSlaveSecurityGroup": "sg-slave1",
    "ServiceAccessSecurityGroup": "sg-service-access",

```

```

    "AdditionalMasterSecurityGroups": ["sg-addMaster1", "sg-addMaster2", "sg-
addMaster3", "sg-addMaster4"],
    "AdditionalSlaveSecurityGroups": ["sg-addSlave1", "sg-addSlave2", "sg-
addSlave3", "sg-addSlave4"]
  }
]

```

Contoh 12: Untuk mengaktifkan debugging dan menentukan URI log

`create-cluster` Contoh berikut menggunakan `--enable-debugging` parameter, yang memungkinkan Anda untuk melihat file log lebih mudah menggunakan alat debugging di konsol EMR Amazon. `--log-uri` parameter diperlukan dengan `--enable-debugging`.

```

aws emr create-cluster \
  --enable-debugging \
  --log-uri s3://myBucket/myLog \
  --release-label emr-5.9.0 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
  --auto-terminate

```

Contoh 13: Untuk menambahkan tag saat membuat cluster

Tag adalah pasangan nilai kunci yang membantu Anda mengidentifikasi dan mengelola cluster. `create-cluster` Contoh berikut menggunakan `--tags` parameter untuk membuat tiga tag untuk sebuah cluster, satu dengan nama kunci `name` dan nilai `Shirley Rodriguez`, yang kedua dengan nama kunci `age` dan nilai `29`, dan tag ketiga dengan nama kunci `department` dan nilai `Analytics`.

```

aws emr create-cluster \
  --tags name="Shirley Rodriguez" age=29 department="Analytics" \
  --release-label emr-5.32.0 \
  --instance-type m5.xlarge \
  --instance-count 3 \
  --use-default-roles

```

Contoh berikut mencantumkan tag yang diterapkan ke cluster.

```

aws emr describe-cluster \
  --cluster-id j-XXXXXXYY \
  --query Cluster.Tags

```

Contoh 14: Untuk menggunakan konfigurasi keamanan yang memungkinkan enkripsi dan fitur keamanan lainnya

`create-cluster` Contoh berikut menggunakan `--security-configuration` parameter untuk menentukan konfigurasi keamanan untuk cluster EMR. Anda dapat menggunakan konfigurasi keamanan dengan Amazon EMR versi 4.8.0 atau yang lebih baru.

```
aws emr create-cluster \
  --instance-type m4.large \
  --release-label emr-5.9.0 \
  --security-configuration mySecurityConfiguration
```

Contoh 15: Untuk membuat cluster dengan volume penyimpanan EBS tambahan yang dikonfigurasi untuk grup instans

Saat menentukan volume EBS tambahan, argumen berikut diperlukan: `VolumeType`, `SizeInGB` jika `EbsBlockDeviceConfigs` ditentukan.

`create-cluster` Contoh berikut membuat cluster dengan beberapa volume EBS yang dilampirkan ke instans EC2 di grup instans inti.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --use-default-roles \
  --instance-groups
  InstanceGroupType=MASTER,InstanceCount=1,InstanceType=d2.xlarge
  'InstanceGroupType=CORE,InstanceCount=2,InstanceType=d2.xlarge,EbsConfiguration={EbsOptimized=true,
  {VolumeSpecification={VolumeType=io1,SizeInGB=100,Iops=100},VolumesPerInstance=4}}' \
  --auto-terminate
```

Contoh berikut membuat cluster dengan beberapa volume EBS yang dilampirkan ke instans EC2 di grup instans master.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --use-default-roles \
  --instance-groups 'InstanceGroupType=MASTER, InstanceCount=1,
  InstanceType=d2.xlarge, EbsConfiguration={EbsOptimized=true,
  EbsBlockDeviceConfigs=[{VolumeSpecification={VolumeType=io1, SizeInGB=100,
  Iops=100}}],
```

```
{VolumeSpecification={VolumeType=standard,SizeInGB=50},VolumesPerInstance=3}}]'
InstanceGroupType=CORE,InstanceCount=2,InstanceType=d2.xlarge \
--auto-terminate
```

### Contoh 16: Untuk membuat kluster dengan kebijakan penskalaan otomatis

Anda dapat melampirkan kebijakan penskalaan otomatis ke grup instance inti dan tugas menggunakan Amazon EMR versi 4.0 dan yang lebih baru. Kebijakan penskalaan otomatis menambahkan dan menghapus instans EC2 secara dinamis sebagai respons terhadap metrik Amazon CloudWatch. Untuk informasi selengkapnya, lihat *Menggunakan Penskalaan Otomatis di Amazon EMR* < ManagementGuide <https://docs.aws.amazon.com/emr/latest/emr-automatic-scaling/> >`\_ di Panduan Manajemen EMR Amazon.

Saat melampirkan kebijakan penskalaan otomatis, Anda juga harus menentukan peran default untuk menggunakan penskalaan otomatis. `--auto-scaling-role EMR_AutoScaling_DefaultRole`

`create-cluster` Contoh berikut menentukan kebijakan penskalaan otomatis untuk grup CORE instance menggunakan `AutoScalingPolicy` argumen dengan struktur JSON tertanam, yang menentukan konfigurasi kebijakan penskalaan. Grup instance dengan struktur JSON tertanam harus memiliki seluruh kumpulan argumen yang diapit tanda kutip tunggal. Menggunakan tanda kutip tunggal adalah opsional untuk grup contoh tanpa struktur JSON yang disematkan.

```
aws emr create-cluster
--release-label emr-5.9.0 \
--use-default-roles --auto-scaling-role EMR_AutoScaling_DefaultRole \
--instance-groups
InstanceGroupType=MASTER,InstanceType=d2.xlarge,InstanceCount=1
'InstanceGroupType=CORE,InstanceType=d2.xlarge,InstanceCount=2,AutoScalingPolicy={Constrain
```

Contoh berikut menggunakan file JSON, `instancegroupconfig.json`, untuk menentukan konfigurasi semua kelompok instance dalam sebuah cluster. File JSON menentukan konfigurasi kebijakan penskalaan otomatis untuk grup instance inti.

```
aws emr create-cluster \
--release-label emr-5.9.0 \
--service-role EMR_DefaultRole \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
--instance-groups file://myfolder/instancegroupconfig.json \
--auto-scaling-role EMR_AutoScaling_DefaultRole
```

Isi dari `instancegroupconfig.json`:

```
[
  {
    "InstanceCount": 1,
    "Name": "MyMasterIG",
    "InstanceGroupType": "MASTER",
    "InstanceType": "m4.large"
  },
  {
    "InstanceCount": 2,
    "Name": "MyCoreIG",
    "InstanceGroupType": "CORE",
    "InstanceType": "m4.large",
    "AutoScalingPolicy": {
      "Constraints": {
        "MinCapacity": 2,
        "MaxCapacity": 10
      },
      "Rules": [
        {
          "Name": "Default-scale-out",
          "Description": "Replicates the default scale-out rule in the
console for YARN memory.",
          "Action": {
            "SimpleScalingPolicyConfiguration": {
              "AdjustmentType": "CHANGE_IN_CAPACITY",
              "ScalingAdjustment": 1,
              "CoolDown": 300
            }
          },
          "Trigger": {
            "CloudWatchAlarmDefinition": {
              "ComparisonOperator": "LESS_THAN",
              "EvaluationPeriods": 1,
              "MetricName": "YARNMemoryAvailablePercentage",
              "Namespace": "AWS/ElasticMapReduce",
              "Period": 300,
              "Threshold": 15,
              "Statistic": "AVERAGE",
              "Unit": "PERCENT",
              "Dimensions": [
                {
                  "Key": "JobFlowId",
```

```
    "Value": "${emr.clusterId}"
  }
]
}
```

### Contoh 17: Tambahkan langkah JAR khusus saat membuat cluster

`create-cluster` Contoh berikut menambahkan langkah-langkah dengan menentukan file JAR yang disimpan di Amazon S3. Langkah-langkah mengirimkan pekerjaan ke cluster. Fungsi utama yang didefinisikan dalam file JAR dijalankan setelah instance EC2 disediakan, tindakan bootstrap apa pun telah dijalankan, dan aplikasi diinstal. Langkah-langkah ditentukan menggunakan `Type=CUSTOM_JAR`.

Langkah-langkah JAR khusus memerlukan `Jar=` parameter, yang menentukan jalur dan nama file JAR. Parameter opsional adalah `TypeName`, `ActionOnFailure`, `Args`, dan `MainClass`. Jika kelas utama tidak ditentukan, file JAR harus menentukan `Main-Class` dalam file manifestnya.

```
aws emr create-cluster \
  --steps Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://
myBucket/mytest.jar,Args=arg1,arg2,arg3
  Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://myBucket/
mytest.jar,MainClass=mymainclass,Args=arg1,arg2,arg3 \
  --release-label emr-5.3.1 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
  --auto-terminate
```

### Contoh 18: Untuk menambahkan langkah streaming saat membuat cluster

`create-cluster` Contoh berikut menambahkan langkah streaming ke cluster yang berakhir setelah semua langkah berjalan. Langkah streaming membutuhkan parameter `Type` dan `Args`. Langkah streaming parameter opsional adalah `Name` dan `ActionOnFailure`.

Contoh berikut menentukan langkah inline.

```
aws emr create-cluster \
```

```

--steps Type=STREAMING,Name='Streaming Program',ActionOnFailure=CONTINUE,Args=[-
files,s3://elasticmapreduce/samples/wordcount/wordSplitter.py,-
mapper,wordSplitter.py,-reducer,aggregate,-input,s3://elasticmapreduce/samples/
wordcount/input,-output,s3://mybucket/wordcount/output] \
--release-label emr-5.3.1 \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
--auto-terminate

```

Contoh berikut menggunakan file konfigurasi JSON yang disimpan secara lokal bernama `multiplefiles.json`. Konfigurasi JSON menentukan beberapa file. Untuk menentukan beberapa file dalam satu langkah, Anda harus menggunakan file konfigurasi JSON untuk menentukan langkahnya. Argumen JSON harus menyertakan opsi dan nilai sebagai item mereka sendiri dalam daftar.

```

aws emr create-cluster \
--steps file://./multiplefiles.json \
--release-label emr-5.9.0 \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
--auto-terminate

```

Isi dari `multiplefiles.json`:

```

[
  {
    "Name": "JSON Streaming Step",
    "Args": [
      "-files",
      "s3://elasticmapreduce/samples/wordcount/wordSplitter.py",
      "-mapper",
      "wordSplitter.py",
      "-reducer",
      "aggregate",
      "-input",
      "s3://elasticmapreduce/samples/wordcount/input",
      "-output",
      "s3://mybucket/wordcount/output"
    ],
    "ActionOnFailure": "CONTINUE",
    "Type": "STREAMING"
  }
]

```



]

Contoh 19: Untuk menambahkan langkah Hive saat membuat cluster

Contoh berikut menambahkan langkah-langkah Hive saat membuat cluster. Langkah-langkah sarang membutuhkan parameter `Type` dan `Args`. Parameter opsional langkah sarang adalah `Name` dan `ActionOnFailure`.

```
aws emr create-cluster \
  --steps Type=HIVE,Name='Hive
  program',ActionOnFailure=CONTINUE,ActionOnFailure=TERMINATE_CLUSTER,Args=[-
  f,s3://elasticmapreduce/samples/hive-ads/libs/model-build.q,-d,INPUT=s3://
  elasticmapreduce/samples/hive-ads/tables,-d,OUTPUT=s3://mybucket/hive-ads/
  output/2014-04-18/11-07-32,-d,LIBS=s3://elasticmapreduce/samples/hive-ads/libs] \
  --applications Name=Hive \
  --release-label emr-5.3.1 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
  InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

Contoh 20: Untuk menambahkan langkah Babi saat membuat cluster

Contoh berikut menambahkan langkah Babi saat membuat cluster. Parameter langkah babi yang diperlukan adalah `Type` dan `Args`. Parameter opsional langkah babi adalah `Name` dan `ActionOnFailure`.

```
aws emr create-cluster \
  --steps Type=PIG,Name='Pig program',ActionOnFailure=CONTINUE,Args=[-f,s3://
  elasticmapreduce/samples/pig-apache/do-reports2.pig,-p,INPUT=s3://elasticmapreduce/
  samples/pig-apache/input,-p,OUTPUT=s3://mybucket/pig-apache/output] \
  --applications Name=Pig \
  --release-label emr-5.3.1 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
  InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

Contoh 21: Untuk menambahkan tindakan bootstrap

`create-cluster` Contoh berikut menjalankan dua tindakan bootstrap didefinisikan sebagai skrip yang disimpan di Amazon S3.

```
aws emr create-cluster \
```

```

--bootstrap-actions Path=s3://mybucket/
myscript1,Name=BootstrapAction1,Args=[arg1,arg2] Path=s3://mybucket/
myscript2,Name=BootstrapAction2,Args=[arg1,arg2] \
--release-label emr-5.3.1 \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
--auto-terminate

```

Contoh 22: Untuk mengaktifkan tampilan konsisten EMRFS dan menyesuaikan pengaturan dan `RetryCount` `RetryPeriod`

`create-cluster` Contoh berikut menentukan hitungan coba lagi dan periode coba lagi untuk tampilan konsisten EMRFS. `Consistent=true` Argumen diperlukan.

```

aws emr create-cluster \
--instance-type m4.large \
--release-label emr-5.9.0 \
--emrfs Consistent=true,RetryCount=6,RetryPeriod=30

```

Contoh berikut menentukan konfigurasi EMRFS yang sama seperti contoh sebelumnya, menggunakan file konfigurasi JSON yang disimpan secara lokal bernama `emrfsconfig.json`

```

aws emr create-cluster \
--instance-type m4.large \
--release-label emr-5.9.0 \
--emrfs file://emrfsconfig.json

```

Isi dari `emrfsconfig.json`:

```

{
  "Consistent": true,
  "RetryCount": 6,
  "RetryPeriod": 30
}

```

Contoh 23: Untuk membuat cluster dengan Kerberos dikonfigurasi

`create-cluster` Contoh berikut membuat cluster menggunakan konfigurasi keamanan dengan Kerberos diaktifkan, dan menetapkan parameter Kerberos untuk cluster menggunakan `--kerberos-attributes`

Perintah berikut menentukan atribut Kerberos untuk cluster inline.

```
aws emr create-cluster \  
  --instance-type m3.xlarge \  
  --release-label emr-5.10.0 \  
  --service-role EMR_DefaultRole \  
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \  
  --security-configuration mySecurityConfiguration \  
  --kerberos-attributes  
  Realm=EC2.INTERNAL,KdcAdminPassword=123,CrossRealmTrustPrincipalPassword=123
```

Perintah berikut menentukan atribut yang sama, tetapi referensi file JSON yang disimpan secara lokal bernama `kerberos_attributes.json`. Dalam contoh ini, file disimpan di direktori yang sama tempat Anda menjalankan perintah. Anda juga dapat mereferensikan file konfigurasi yang disimpan di Amazon S3.

```
aws emr create-cluster \  
  --instance-type m3.xlarge \  
  --release-label emr-5.10.0 \  
  --service-role EMR_DefaultRole \  
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \  
  --security-configuration mySecurityConfiguration \  
  --kerberos-attributes file://kerberos_attributes.json
```

Isi dari `kerberos_attributes.json`:

```
{  
  "Realm": "EC2.INTERNAL",  
  "KdcAdminPassword": "123",  
  "CrossRealmTrustPrincipalPassword": "123",  
}
```

Contoh berikut membuat kluster EMR Amazon yang menggunakan `--instance-groups` konfigurasi dan memiliki kebijakan penskalaan terkelola.

```
aws emr create-cluster \  
  --release-label emr-5.30.0 \  
  --service-role EMR_DefaultRole \  
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \  
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large  
  InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

```
--managed-scaling-policy
ComputeLimits='{MinimumCapacityUnits=2,MaximumCapacityUnits=4,UnitType=Instances}'
```

`create-cluster` Contoh berikut membuat kluster EMR Amazon yang menggunakan “`--log-encryption-kms-key-id`” untuk menentukan ID kunci KMS yang digunakan untuk enkripsi Log.

```
aws emr create-cluster \
  --release-label emr-5.30.0 \
  --log-uri s3://myBucket/myLog \
  --log-encryption-kms-key-id arn:aws:kms:us-east-1:110302272565:key/
dd559181-283e-45d7-99d1-66da348c4d33 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large
```

`create-cluster` Contoh berikut membuat kluster EMR Amazon yang menggunakan konfigurasi “`--placement-group-configs`” untuk menempatkan node master dalam kluster ketersediaan tinggi (HA) dalam grup penempatan EC2 menggunakan strategi penempatan. SPREAD

```
aws emr create-cluster \
  --release-label emr-5.30.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-groups
InstanceGroupType=MASTER,InstanceCount=3,InstanceType=m4.largeInstanceGroupType=CORE,Instan
\
  --placement-group-configs InstanceRole=MASTER
```

`create-cluster` Contoh berikut membuat kluster EMR Amazon yang menggunakan konfigurasi “`--auto-termination-policy`” untuk menempatkan ambang terminasi idle otomatis untuk kluster.

```
aws emr create-cluster \
  --release-label emr-5.34.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=1,InstanceType=m4.large \
  --auto-termination-policy IdleTimeout=100
```

`create-cluster` Contoh berikut membuat kluster EMR Amazon yang menggunakan “`--os-release-label`” untuk menentukan rilis Amazon Linux untuk peluncuran kluster

```
aws emr create-cluster \
  --release-label emr-6.6.0 \
  --os-release-label 2.0.20220406.1 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
  InstanceGroupType=CORE,InstanceCount=1,InstanceType=m4.large
```

Contoh 24: Untuk menentukan atribut volume root EBS: size, iops, dan throughput untuk instance cluster yang dibuat dengan rilis EMR 6.15.0 dan yang lebih baru

`create-cluster` Contoh berikut membuat kluster EMR Amazon yang menggunakan atribut volume root untuk mengonfigurasi spesifikasi volume root untuk instans EC2.

```
aws emr create-cluster \
  --name "Cluster with My Custom AMI" \
  --custom-ami-id ami-a518e6df \
  --ebs-root-volume-size 20 \
  --ebs-root-volume-iops 3000 \
  --ebs-root-volume-throughput 125 \
  --release-label emr-6.15.0 \
  --use-default-roles \
  --instance-count 2 \
  --instance-type m4.large
```

- Untuk detail API, lihat [CreateClusterExamples](#) di Referensi AWS CLI Perintah.

## create-default-roles

Contoh kode berikut menunjukkan cara menggunakan `create-default-roles`.

### AWS CLI

1. Untuk membuat peran IAM default untuk EC2

Perintah:

```
aws emr create-default-roles
```

Output:

If the role already exists then the command returns nothing.

If the role does not exist then the output will be:

```
[
  {
    "RolePolicy": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Action": [
            "cloudwatch:*",
            "dynamodb:*",
            "ec2:Describe*",
            "elasticmapreduce:Describe*",
            "elasticmapreduce:ListBootstrapActions",
            "elasticmapreduce:ListClusters",
            "elasticmapreduce:ListInstanceGroups",
            "elasticmapreduce:ListInstances",
            "elasticmapreduce:ListSteps",
            "kinesis:CreateStream",
            "kinesis>DeleteStream",
            "kinesis:DescribeStream",
            "kinesis:GetRecords",
            "kinesis:GetShardIterator",
            "kinesis:MergeShards",
            "kinesis:PutRecord",
            "kinesis:SplitShard",
            "rds:Describe*",
            "s3:*",
            "sdb:*",
            "sns:*",
            "sqs:*"
          ],
          "Resource": "*",
          "Effect": "Allow"
        }
      ]
    },
    "Role": {
      "AssumeRolePolicyDocument": {
        "Version": "2008-10-17",
        "Statement": [
```

```

        {
            "Action": "sts:AssumeRole",
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            }
        }
    ]
},
"RoleId": "AR0AIQ5SIUGL5KMYBJX6",
"CreateDate": "2015-06-09T17:09:04.602Z",
"RoleName": "EMR_EC2_DefaultRole",
"Path": "/",
"Arn": "arn:aws:iam::176430881729:role/EMR_EC2_DefaultRole"
}
},
{
  "RolePolicy": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": [
          "ec2:AuthorizeSecurityGroupIngress",
          "ec2:CancelSpotInstanceRequests",
          "ec2:CreateSecurityGroup",
          "ec2:CreateTags",
          "ec2>DeleteTags",
          "ec2:DescribeAvailabilityZones",
          "ec2:DescribeAccountAttributes",
          "ec2:DescribeInstances",
          "ec2:DescribeInstanceStatus",
          "ec2:DescribeKeyPairs",
          "ec2:DescribePrefixLists",
          "ec2:DescribeRouteTables",
          "ec2:DescribeSecurityGroups",
          "ec2:DescribeSpotInstanceRequests",
          "ec2:DescribeSpotPriceHistory",
          "ec2:DescribeSubnets",
          "ec2:DescribeVpcAttribute",
          "ec2:DescribeVpcEndpoints",
          "ec2:DescribeVpcEndpointServices",
          "ec2:DescribeVpcs",
          "ec2:ModifyImageAttribute",

```

```

        "ec2:ModifyInstanceAttribute",
        "ec2:RequestSpotInstances",
        "ec2:RunInstances",
        "ec2:TerminateInstances",
        "iam:GetRole",
        "iam:GetRolePolicy",
        "iam:ListInstanceProfiles",
        "iam:ListRolePolicies",
        "iam:PassRole",
        "s3:CreateBucket",
        "s3:Get*",
        "s3:List*",
        "sdb:BatchPutAttributes",
        "sdb:Select",
        "sqs:CreateQueue",
        "sqs:Delete*",
        "sqs:GetQueue*",
        "sqs:ReceiveMessage"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
},
"Role": {
    "AssumeRolePolicyDocument": {
        "Version": "2008-10-17",
        "Statement": [
            {
                "Action": "sts:AssumeRole",
                "Sid": "",
                "Effect": "Allow",
                "Principal": {
                    "Service": "elasticmapreduce.amazonaws.com"
                }
            }
        ]
    },
    "RoleId": "AROAI3SRVPPVSRDLARBPY",
    "CreateDate": "2015-06-09T17:09:10.401Z",
    "RoleName": "EMR_DefaultRole",
    "Path": "/",
    "Arn": "arn:aws:iam::176430881729:role/EMR_DefaultRole"
}

```



```
    }
  ]
}
```

- Untuk detail API, lihat [CreateDefaultRoles](#) di Referensi AWS CLI Perintah.

## create-security-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-security-configuration`.

### AWS CLI

1. Untuk membuat konfigurasi keamanan dengan enkripsi dalam transit yang diaktifkan dengan PEM untuk penyedia sertifikat, dan enkripsi saat istirahat diaktifkan dengan SSE-S3 untuk enkripsi S3 dan `-KMS` untuk penyedia kunci disk lokal AWS

Perintah:

```
aws emr create-security-configuration --name MySecurityConfig --security-
configuration '{
  "EncryptionConfiguration": {
    "EnableInTransitEncryption" : true,
    "EnableAtRestEncryption" : true,
    "InTransitEncryptionConfiguration" : {
      "TLSCertificateConfiguration" : {
        "CertificateProviderType" : "PEM",
        "S3Object" : "s3://mycertstore/artifacts/
MyCerts.zip"
      }
    },
    "AtRestEncryptionConfiguration" : {
      "S3EncryptionConfiguration" : {
        "EncryptionMode" : "SSE-S3"
      },
      "LocalDiskEncryptionConfiguration" : {
        "EncryptionKeyProviderType" : "AwsKms",
        "AwsKmsKey" : "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
      }
    }
  }
}'
```

**Output:**

```
{
  "CreationDateTime": 1474070889.129,
  "Name": "MySecurityConfig"
}
```

**Setara JSON (isi security\_configuration.json):**

```
{
  "EncryptionConfiguration": {
    "EnableInTransitEncryption": true,
    "EnableAtRestEncryption": true,
    "InTransitEncryptionConfiguration": {
      "TLSCertificateConfiguration": {
        "CertificateProviderType": "PEM",
        "S3Object": "s3://mycertstore/artifacts/MyCerts.zip"
      }
    },
    "AtRestEncryptionConfiguration": {
      "S3EncryptionConfiguration": {
        "EncryptionMode": "SSE-S3"
      },
      "LocalDiskEncryptionConfiguration": {
        "EncryptionKeyProviderType": "AwsKms",
        "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
      }
    }
  }
}
```

**Perintah (menggunakan security\_configuration.json):**

```
aws emr create-security-configuration --name "MySecurityConfig" --security-
configuration file://./security_configuration.json
```

**Output:**

```
{
  "CreationDateTime": 1474070889.129,
  "Name": "MySecurityConfig"
}
```

```
}

```

2. Untuk membuat konfigurasi keamanan dengan Kerberos diaktifkan menggunakan KDC khusus cluster dan kepercayaan lintas alam

Perintah:

```
aws emr create-security-configuration --name MySecurityConfig --security-
configuration '{
  "AuthenticationConfiguration": {
    "KerberosConfiguration": {
      "Provider": "ClusterDedicatedKdc",
      "ClusterDedicatedKdcConfiguration": {
        "TicketLifetimeInHours": 24,
        "CrossRealmTrustConfiguration": {
          "Realm": "AD.DOMAIN.COM",
          "Domain": "ad.domain.com",
          "AdminServer": "ad.domain.com",
          "KdcServer": "ad.domain.com"
        }
      }
    }
  }
}'

```

Output:

```
{
  "CreationDateTime": 1490225558.982,
  "Name": "MySecurityConfig"
}
```

Setara JSON (isi security\_configuration.json):

```
{
  "AuthenticationConfiguration": {
    "KerberosConfiguration": {
      "Provider": "ClusterDedicatedKdc",
      "ClusterDedicatedKdcConfiguration": {
        "TicketLifetimeInHours": 24,
        "CrossRealmTrustConfiguration": {
          "Realm": "AD.DOMAIN.COM",

```

```
        "Domain": "ad.domain.com",
        "AdminServer": "ad.domain.com",
        "KdcServer": "ad.domain.com"
    }
}
}
```

Perintah (menggunakan `security_configuration.json`):

```
aws emr create-security-configuration --name "MySecurityConfig" --security-configuration file:///./security_configuration.json
```

Output:

```
{
  "CreationDateTime": 1490225558.982,
  "Name": "MySecurityConfig"
}
```

- Untuk detail API, lihat [CreateSecurityConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-security-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-security-configuration`.

AWS CLI

Untuk menghapus konfigurasi keamanan di wilayah saat ini

Perintah:

```
aws emr delete-security-configuration --name MySecurityConfig
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteSecurityConfiguration](#) di Referensi AWS CLI Perintah.

## describe-cluster

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster`.

### AWS CLI

Perintah:

```
aws emr describe-cluster --cluster-id j-XXXXXXXX
```

Output:

For release-label based uniform instance groups cluster:

```
{
  "Cluster": {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1436475075.199,
        "CreationDateTime": 1436474656.563,
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting for steps to run"
      }
    },
    "Ec2InstanceAttributes": {
      "ServiceAccessSecurityGroup": "sg-xxxxxxx",
      "EmrManagedMasterSecurityGroup": "sg-xxxxxxx",
      "IamInstanceProfile": "EMR_EC2_DefaultRole",
      "Ec2KeyName": "myKey",
      "Ec2AvailabilityZone": "us-east-1c",
      "EmrManagedSlaveSecurityGroup": "sg-yyyyyyyyyy"
    },
    "Name": "My Cluster",
    "ServiceRole": "EMR_DefaultRole",
    "Tags": [],
    "TerminationProtected": true,
    "UnhealthyNodeReplacement": true,
    "ReleaseLabel": "emr-4.0.0",
    "NormalizedInstanceHours": 96,
    "InstanceGroups": [
      {
```

```
    "RequestedInstanceCount": 2,
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1436475074.245,
        "CreationDateTime": 1436474656.564,
        "EndDateTime": 1436638158.387
      },
      "State": "RUNNING",
      "StateChangeReason": {
        "Message": "",
      }
    },
    "Name": "CORE",
    "InstanceGroupType": "CORE",
    "Id": "ig-YYYYYYYY",
    "Configurations": [],
    "InstanceType": "m3.large",
    "Market": "ON_DEMAND",
    "RunningInstanceCount": 2
  },
  {
    "RequestedInstanceCount": 1,
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1436475074.245,
        "CreationDateTime": 1436474656.564,
        "EndDateTime": 1436638158.387
      },
      "State": "RUNNING",
      "StateChangeReason": {
        "Message": "",
      }
    },
    "Name": "MASTER",
    "InstanceGroupType": "MASTER",
    "Id": "ig-XXXXXXXXXX",
    "Configurations": [],
    "InstanceType": "m3.large",
    "Market": "ON_DEMAND",
    "RunningInstanceCount": 1
  }
],
"Applications": [
  {
```

```

        "Name": "Hadoop"
      }
    ],
    "VisibleToAllUsers": true,
    "BootstrapActions": [],
    "MasterPublicDnsName": "ec2-54-147-144-78.compute-1.amazonaws.com",
    "AutoTerminate": false,
    "Id": "j-XXXXXXXX",
    "Configurations": [
      {
        "Properties": {
          "fs.s3.consistent.retryPeriodSeconds": "20",
          "fs.s3.enableServerSideEncryption": "true",
          "fs.s3.consistent": "false",
          "fs.s3.consistent.retryCount": "2"
        },
        "Classification": "emrfs-site"
      }
    ]
  }
}

```

For release-label based instance fleet cluster:

```

{
  "Cluster": {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1487897289.705,
        "CreationDateTime": 1487896933.942
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting for steps to run"
      }
    },
    "Ec2InstanceAttributes": {
      "EmrManagedMasterSecurityGroup": "sg-xxxxx",
      "RequestedEc2AvailabilityZones": [],
      "RequestedEc2SubnetIds": [],
      "IamInstanceProfile": "EMR_EC2_DefaultRole",
      "Ec2AvailabilityZone": "us-east-1a",
      "EmrManagedSlaveSecurityGroup": "sg-xxxxx"
    },

```

```
"Name": "My Cluster",
"ServiceRole": "EMR_DefaultRole",
"Tags": [],
"TerminationProtected": false,
"UnhealthyNodeReplacement": false,
"ReleaseLabel": "emr-5.2.0",
"NormalizedInstanceHours": 472,
"InstanceCollectionType": "INSTANCE_FLEET",
"InstanceFleets": [
  {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1487897212.74,
        "CreationDateTime": 1487896933.948
      },
      "State": "RUNNING",
      "StateChangeReason": {
        "Message": ""
      }
    },
    "ProvisionedSpotCapacity": 1,
    "Name": "MASTER",
    "InstanceFleetType": "MASTER",
    "LaunchSpecifications": {
      "SpotSpecification": {
        "TimeoutDurationMinutes": 60,
        "TimeoutAction": "TERMINATE_CLUSTER"
      }
    },
    "TargetSpotCapacity": 1,
    "ProvisionedOnDemandCapacity": 0,
    "InstanceTypeSpecifications": [
      {
        "BidPrice": "0.5",
        "InstanceType": "m3.xlarge",
        "WeightedCapacity": 1
      }
    ],
    "Id": "if-xxxxxxx",
    "TargetOnDemandCapacity": 0
  }
],
"Applications": [
  {
```



```

        "Version": "2.7.3",
        "Name": "Hadoop"
    }
],
"ScaleDownBehavior": "TERMINATE_AT_INSTANCE_HOUR",
"VisibleToAllUsers": true,
"BootstrapActions": [],
"MasterPublicDnsName": "ec2-xxx-xx-xxx-xx.compute-1.amazonaws.com",
"AutoTerminate": false,
"Id": "j-xxxxx",
"Configurations": []
}
}

```

For ami based uniform instance group cluster:

```

{
  "Cluster": {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1399400564.432,
        "CreationDateTime": 1399400268.62
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting for steps to run"
      }
    },
    "Ec2InstanceAttributes": {
      "IamInstanceProfile": "EMR_EC2_DefaultRole",
      "Ec2AvailabilityZone": "us-east-1c"
    },
    "Name": "My Cluster",
    "Tags": [],
    "TerminationProtected": true,
    "UnhealthyNodeReplacement": true,
    "RunningAmiVersion": "2.5.4",
    "InstanceGroups": [
      {
        "RequestedInstanceCount": 1,
        "Status": {
          "Timeline": {
            "ReadyDateTime": 1399400558.848,
            "CreationDateTime": 1399400268.621
          }
        }
      }
    ]
  }
}

```

```
    },
    "State": "RUNNING",
    "StateChangeReason": {
      "Message": ""
    }
  },
  "Name": "Master instance group",
  "InstanceGroupType": "MASTER",
  "InstanceType": "m1.small",
  "Id": "ig-ABCD",
  "Market": "ON_DEMAND",
  "RunningInstanceCount": 1
},
{
  "RequestedInstanceCount": 2,
  "Status": {
    "Timeline": {
      "ReadyDateTime": 1399400564.439,
      "CreationDateTime": 1399400268.621
    },
    "State": "RUNNING",
    "StateChangeReason": {
      "Message": ""
    }
  },
  "Name": "Core instance group",
  "InstanceGroupType": "CORE",
  "InstanceType": "m1.small",
  "Id": "ig-DEF",
  "Market": "ON_DEMAND",
  "RunningInstanceCount": 2
}
],
"Applications": [
  {
    "Version": "1.0.3",
    "Name": "hadoop"
  }
],
"BootstrapActions": [],
"VisibleToAllUsers": false,
"RequestedAmiVersion": "2.4.2",
"LogUri": "s3://myLogUri/",
"AutoTerminate": false,
```

```
    "Id": "j-XXXXXXXX"
  }
}
```

- Untuk detail API, lihat [DescribeCluster](#) di Referensi AWS CLI Perintah.

## describe-step

Contoh kode berikut menunjukkan cara menggunakan `describe-step`.

### AWS CLI

Perintah berikut menjelaskan langkah dengan ID langkah `s-3LZC0QUT43AM` dalam cluster dengan ID cluster `j-3SD91U2E1L2QX`:

```
aws emr describe-step --cluster-id j-3SD91U2E1L2QX --step-id s-3LZC0QUT43AM
```

### Output:

```
{
  "Step": {
    "Status": {
      "Timeline": {
        "EndDateTime": 1433200470.481,
        "CreationDateTime": 1433199926.597,
        "StartDateTime": 1433200404.959
      },
      "State": "COMPLETED",
      "StateChangeReason": {}
    },
    "Config": {
      "Args": [
        "s3://us-west-2.elasticmapreduce/libs/hive/hive-script",
        "--base-path",
        "s3://us-west-2.elasticmapreduce/libs/hive/",
        "--install-hive",
        "--hive-versions",
        "0.13.1"
      ],
      "Jar": "s3://us-west-2.elasticmapreduce/libs/script-runner/script-runner.jar",
      "Properties": {}
    }
  }
}
```

```
    },
    "Id": "s-3LZC0QUT43AM",
    "ActionOnFailure": "TERMINATE_CLUSTER",
    "Name": "Setup hive"
  }
}
```

- Untuk detail API, lihat [DescribeStep](#) di Referensi AWS CLI Perintah.

## get

Contoh kode berikut menunjukkan cara menggunakan `get`.

### AWS CLI

Berikut ini mengunduh `hadoop-examples.jar` arsip dari instance master di cluster dengan ID `clusterj-3SD91U2E1L2QX`:

```
aws emr get --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem --src /
home/hadoop-examples.jar --dest ~
```

- Untuk detail API, lihat [Dapatkan](#) Referensi AWS CLI Perintah.

## list-clusters

Contoh kode berikut menunjukkan cara menggunakan `list-clusters`.

### AWS CLI

Perintah berikut mencantumkan semua cluster EMR aktif di wilayah saat ini:

```
aws emr list-clusters --active
```

Output:

```
{
  "Clusters": [
    {
      "Status": {
        "Timeline": {
          "ReadyDateTime": 1433200405.353,
```

```

        "CreationDateTime": 1433199926.596
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting after step completed"
      }
    },
    "NormalizedInstanceHours": 6,
    "Id": "j-3SD91U2E1L2QX",
    "Name": "my-cluster"
  }
]
}

```

- Untuk detail API, lihat [ListClusters](#) di Referensi AWS CLI Perintah.

## list-instance-fleets

Contoh kode berikut menunjukkan cara menggunakan `list-instance-fleets`.

### AWS CLI

Untuk mendapatkan detail konfigurasi armada instance dalam sebuah cluster

Contoh ini mencantumkan rincian armada instance di cluster yang ditentukan.

Perintah:

```
list-instance-fleets --cluster-id 'j-12ABCDEFGH134JK'
```

Output:

```

{
  "InstanceFleets": [
    {
      "Status": {
        "Timeline": {
          "ReadyDateTime": 1488759094.637,
          "CreationDateTime": 1488758719.817
        },
        "State": "RUNNING",
        "StateChangeReason": {
          "Message": ""
        }
      }
    }
  ]
}

```

```
    }
  },
  "ProvisionedSpotCapacity": 6,
  "Name": "CORE",
  "InstanceFleetType": "CORE",
  "LaunchSpecifications": {
    "SpotSpecification": {
      "TimeoutDurationMinutes": 60,
      "TimeoutAction": "TERMINATE_CLUSTER"
    }
  },
  "ProvisionedOnDemandCapacity": 2,
  "InstanceTypeSpecifications": [
    {
      "BidPrice": "0.5",
      "InstanceType": "m3.xlarge",
      "WeightedCapacity": 2
    }
  ],
  "Id": "if-1ABC2DEFGHIJ3"
},
{
  "Status": {
    "Timeline": {
      "ReadyDateTime": 1488759058.598,
      "CreationDateTime": 1488758719.811
    },
    "State": "RUNNING",
    "StateChangeReason": {
      "Message": ""
    }
  },
  "ProvisionedSpotCapacity": 0,
  "Name": "MASTER",
  "InstanceFleetType": "MASTER",
  "ProvisionedOnDemandCapacity": 1,
  "InstanceTypeSpecifications": [
    {
      "BidPriceAsPercentageOfOnDemandPrice": 100.0,
      "InstanceType": "m3.xlarge",
      "WeightedCapacity": 1
    }
  ],
  "Id": "if-2ABC4DEFGHIJ4"
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [ListInstanceFleets](#) di Referensi AWS CLI Perintah.

## list-instances

Contoh kode berikut menunjukkan cara menggunakan `list-instances`.

### AWS CLI

Perintah berikut mencantumkan semua instance dalam cluster dengan ID `j-3C6XNQ39VR9WL` cluster:

```
aws emr list-instances --cluster-id j-3C6XNQ39VR9WL
```

### Output:

```
For a uniform instance group based cluster  
{  
  "Instances": [  
    {  
      "Status": {  
        "Timeline": {  
          "ReadyDateTime": 1433200400.03,  
          "CreationDateTime": 1433199960.152  
        },  
        "State": "RUNNING",  
        "StateChangeReason": {}  
      },  
      "Ec2InstanceId": "i-f19ecfee",  
      "PublicDnsName": "ec2-52-52-41-150.us-west-2.compute.amazonaws.com",  
      "PrivateDnsName": "ip-172-21-11-216.us-west-2.compute.internal",  
      "PublicIpAddress": "52.52.41.150",  
      "Id": "ci-3NNHQUQ2TWB6Y",  
      "PrivateIpAddress": "172.21.11.216"  
    },  
    {  
      "Status": {  
        "Timeline": {  
          "ReadyDateTime": 1433200400.031,
```

```

        "CreationDateTime": 1433199949.102
      },
      "State": "RUNNING",
      "StateChangeReason": {}
    },
    "Ec2InstanceId": "i-1feee4c2",
    "PublicDnsName": "ec2-52-63-246-32.us-west-2.compute.amazonaws.com",
    "PrivateDnsName": "ip-172-31-24-130.us-west-2.compute.internal",
    "PublicIpAddress": "52.63.246.32",
    "Id": "ci-GAOCMKNKDCV7",
    "PrivateIpAddress": "172.21.11.215"
  },
  {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1433200400.031,
        "CreationDateTime": 1433199949.102
      },
      "State": "RUNNING",
      "StateChangeReason": {}
    },
    "Ec2InstanceId": "i-15cfeee3",
    "PublicDnsName": "ec2-52-25-246-63.us-west-2.compute.amazonaws.com",
    "PrivateDnsName": "ip-172-31-24-129.us-west-2.compute.internal",
    "PublicIpAddress": "52.25.246.63",
    "Id": "ci-2W3TDFFB47UAD",
    "PrivateIpAddress": "172.21.11.214"
  }
]
}

```

For a fleet based cluster:

```

{
  "Instances": [
    {
      "Status": {
        "Timeline": {
          "ReadyDateTime": 1487810810.878,
          "CreationDateTime": 1487810588.367,
          "EndDateTime": 1488022990.924
        },
        "State": "TERMINATED",
        "StateChangeReason": {

```



```
        "Message": "Instance was terminated."
      }
    },
    "Ec2InstanceId": "i-xxxxx",
    "InstanceFleetId": "if-xxxxx",
    "EbsVolumes": [],
    "PublicDnsName": "ec2-xx-xxx-xxx-xxx.compute-1.amazonaws.com",
    "InstanceType": "m3.xlarge",
    "PrivateDnsName": "ip-xx-xx-xxx-xx.ec2.internal",
    "Market": "SPOT",
    "PublicIpAddress": "xx.xx.xxx.xxx",
    "Id": "ci-xxxxx",
    "PrivateIpAddress": "10.47.191.80"
  }
]
}
```

- Untuk detail API, lihat [ListInstances](#) di Referensi AWS CLI Perintah.

## list-security-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-security-configurations`.

### AWS CLI

Untuk membuat daftar konfigurasi keamanan di wilayah saat ini

Perintah:

```
aws emr list-security-configurations
```

Output:

```
{
  "SecurityConfigurations": [
    {
      "CreationDateTime": 1473889697.417,
      "Name": "MySecurityConfig-1"
    },
    {
      "CreationDateTime": 1473889697.417,
      "Name": "MySecurityConfig-2"
    }
  ]
}
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [ListSecurityConfigurations](#) di Referensi AWS CLI Perintah.

## **list-steps**

Contoh kode berikut menunjukkan cara menggunakan `list-steps`.

### AWS CLI

Perintah berikut mencantumkan semua langkah dalam cluster dengan ID cluster `j-3SD91U2E1L2QX`:

```
aws emr list-steps --cluster-id j-3SD91U2E1L2QX
```

- Untuk detail API, lihat [ListSteps](#) di Referensi AWS CLI Perintah.

## **modify-cluster-attributes**

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-attributes`.

### AWS CLI

Perintah berikut menetapkan visibilitas cluster EMR dengan `j-301CDNY0J5XM4` ID untuk semua pengguna:

```
aws emr modify-cluster-attributes --cluster-id j-301CDNY0J5XM4 --visible-to-all-users
```

- Untuk detail API, lihat [ModifyClusterAttributes](#) di Referensi AWS CLI Perintah.

## **modify-instance-fleet**

Contoh kode berikut menunjukkan cara menggunakan `modify-instance-fleet`.

### AWS CLI

Untuk mengubah kapasitas target armada instance

Contoh ini mengubah kapasitas target On-Demand dan Spot menjadi 1 untuk armada instans yang ditentukan.

Perintah:

```
aws emr modify-instance-fleet --cluster-id 'j-12ABCDEFGH134JK' --instance-fleet InstanceFleetId='if-2ABC4DEFGHIJ4',TargetOnDemandCapacity=1,TargetSpotCapacity=1
```

- Untuk detail API, lihat [ModifyInstanceFleet](#) di Referensi AWS CLI Perintah.

## put

Contoh kode berikut menunjukkan cara menggunakan `put`.

### AWS CLI

Perintah berikut mengunggah file bernama `healthcheck.sh` ke instance master di cluster dengan ID `j-3SD91U2E1L2QX` cluster:

```
aws emr put --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem --src ~/scripts/healthcheck.sh --dest /home/hadoop/bin/healthcheck.sh
```

- Untuk detail API, lihat [Masukkan](#) Referensi AWS CLI Perintah.

## remove-tags

Contoh kode berikut menunjukkan cara menggunakan `remove-tags`.

### AWS CLI

Perintah berikut menghapus tag dengan kunci `prod` dari cluster dengan ID `clusterj-3SD91U2E1L2QX`:

```
aws emr remove-tags --resource-id j-3SD91U2E1L2QX --tag-keys prod
```

- Untuk detail API, lihat [RemoveTags](#) di Referensi AWS CLI Perintah.

## schedule-hbase-backup

Contoh kode berikut menunjukkan cara menggunakan `schedule-hbase-backup`.

## AWS CLI

Catatan: Perintah ini hanya dapat digunakan dengan HBase pada AMI versi 2.x dan 3.x

1. Untuk menjadwalkan cadangan HBase penuh >>>>>  
06ab6d6e13564b5733d75abaf3b599f93cf39a23

Perintah:

```
aws emr schedule-hbase-backup --cluster-id j-XXXXXXYY --type full --dir
s3://myBucket/backup --interval 10 --unit hours --start-time
2014-04-21T05:26:10Z --consistent
```

Output:

```
None
```

2. Untuk menjadwalkan pencadangan HBase tambahan

Perintah:

```
aws emr schedule-hbase-backup --cluster-id j-XXXXXXYY --type incremental
--dir s3://myBucket/backup --interval 30 --unit minutes --start-time
2014-04-21T05:26:10Z --consistent
```

Output:

```
None
```

- Untuk detail API, lihat [ScheduleHbaseBackup](#) di Referensi AWS CLI Perintah.

## socks

Contoh kode berikut menunjukkan cara menggunakan socks.

## AWS CLI

Perintah berikut membuka koneksi socks dengan instance master di cluster dengan ID cluster `j-3SD91U2E1L2QX`:

```
aws emr socks --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem
```

Opsi key pair file mengambil jalur lokal ke file kunci pribadi.

- Untuk detail API, lihat [Socks](#) in AWS CLI Command Reference.

## ssh

Contoh kode berikut menunjukkan cara menggunakan ssh.

### AWS CLI

Perintah berikut membuka koneksi ssh dengan instance master di cluster dengan ID j-3SD91U2E1L2QX cluster:

```
aws emr ssh --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem
```

Opsi key pair file mengambil jalur lokal ke file kunci pribadi.

Output:

```
ssh -o StrictHostKeyChecking=no -o ServerAliveInterval=10 -i /home/local/user/.ssh/mykey.pem hadoop@ec2-52-52-41-150.us-west-2.compute.amazonaws.com
Warning: Permanently added 'ec2-52-52-41-150.us-west-2.compute.amazonaws.com,52.52.41.150' (ECDSA) to the list of known hosts.
Last login: Mon Jun  1 23:15:38 2015

  _|  _|_ )
  _| (    /  Amazon Linux AMI
  _|\__|__|

https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/
26 package(s) needed for security, out of 39 available
Run "sudo yum update" to apply all updates.

-----

Welcome to Amazon Elastic MapReduce running Hadoop and Amazon Linux.

Hadoop is installed in /home/hadoop. Log files are in /mnt/var/log/hadoop. Check
/mnt/var/log/hadoop/steps for diagnosing step failures.

The Hadoop UI can be accessed via the following commands:

ResourceManager    lynx http://ip-172-21-11-216:9026/
```

```
NameNode          lynx http://ip-172-21-11-216:9101/
```

```
-----  
[hadoop@ip-172-31-16-216 ~]$
```

- Untuk detail API, lihat [Ssh](#) di Referensi AWS CLI Perintah.

## Amazon EMR pada contoh EKS menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan EMR AWS Command Line Interface dengan Amazon di EKS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **update-role-trust-policy**

Contoh kode berikut menunjukkan cara menggunakan `update-role-trust-policy`.

AWS CLI

Untuk memperbarui kebijakan kepercayaan Peran IAM yang akan digunakan dengan Amazon EMR di EKS

Perintah contoh ini memperbarui kebijakan kepercayaan dari peran bernama `example_iam_role` sedemikian rupa sehingga dapat digunakan dengan Amazon EMR di EKS dengan namespace `example_namespace` dari cluster EKS bernama `example_cluster`.

## Perintah:

```
aws emr-containers update-role-trust-policy \  
  --cluster example_cluster \  
  --namespace example_namespace \  
  --role-name example_iam_role
```

## Output:

If the trust policy has already been updated, then the output will be:  
Trust policy statement already exists for role example\_iam\_role. No changes were made!

If the trust policy has not been updated yet, then the output will be:  
Successfully updated trust policy of role example\_iam\_role.

- Untuk detail API, lihat [UpdateRoleTrustPolicy](#) di Referensi AWS CLI Perintah.

## EventBridge contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with EventBridge.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **delete-rule**

Contoh kode berikut menunjukkan cara menggunakan `delete-rule`.

## AWS CLI

Untuk menghapus aturan CloudWatch Acara

Contoh ini menghapus aturan bernama InstanceStateChanges EC2:

```
aws events delete-rule --name "EC2InstanceStateChanges"
```

- Untuk detail API, lihat [DeleteRule](#) di Referensi AWS CLI Perintah.

## describe-rule

Contoh kode berikut menunjukkan cara menggunakan `describe-rule`.

## AWS CLI

Untuk menampilkan informasi tentang aturan CloudWatch Acara

Contoh ini menampilkan informasi tentang aturan bernama DailyLambdaFunction:

```
aws events describe-rule --name "DailyLambdaFunction"
```

- Untuk detail API, lihat [DescribeRule](#) di Referensi AWS CLI Perintah.

## disable-rule

Contoh kode berikut menunjukkan cara menggunakan `disable-rule`.

## AWS CLI

Untuk menonaktifkan aturan CloudWatch Acara

Contoh ini menonaktifkan aturan bernama DailyLambdaFunction. Aturannya tidak dihapus:

```
aws events disable-rule --name "DailyLambdaFunction"
```

- Untuk detail API, lihat [DisableRule](#) di Referensi AWS CLI Perintah.

## enable-rule

Contoh kode berikut menunjukkan cara menggunakan `enable-rule`.



## AWS CLI

Untuk mengaktifkan aturan CloudWatch Acara

Contoh ini memungkinkan aturan bernama `DailyLambdaFunction`, yang sebelumnya telah dinonaktifkan:

```
aws events enable-rule --name "DailyLambdaFunction"
```

- Untuk detail API, lihat [EnableRule](#) di Referensi AWS CLI Perintah.

## `list-rule-names-by-target`

Contoh kode berikut menunjukkan cara menggunakan `list-rule-names-by-target`.

## AWS CLI

Untuk menampilkan semua aturan yang memiliki target tertentu

Contoh ini menampilkan semua aturan yang memiliki fungsi Lambda bernama `"MyFunctionName"` sebagai target:

```
aws events list-rule-names-by-target --target-arn "arn:aws:lambda:us-east-1:123456789012:function:MyFunctionName"
```

- Untuk detail API, lihat [ListRuleNamesByTarget](#) di Referensi AWS CLI Perintah.

## `list-rules`

Contoh kode berikut menunjukkan cara menggunakan `list-rules`.

## AWS CLI

Untuk menampilkan daftar semua aturan CloudWatch Acara

Contoh ini menampilkan semua aturan CloudWatch Peristiwa di wilayah:

```
aws events list-rules
```

Untuk menampilkan daftar aturan CloudWatch Peristiwa yang dimulai dengan string tertentu.

Contoh ini menampilkan semua aturan CloudWatch Acara di wilayah yang memiliki nama yang dimulai dengan "Harian":

```
aws events list-rules --name-prefix "Daily"
```

- Untuk detail API, lihat [ListRules](#) di Referensi AWS CLI Perintah.

## list-targets-by-rule

Contoh kode berikut menunjukkan cara menggunakan `list-targets-by-rule`.

### AWS CLI

Untuk menampilkan semua target untuk aturan CloudWatch Acara

Contoh ini menampilkan semua target dari aturan bernama `DailyLambdaFunction`:

```
aws events list-targets-by-rule --rule "DailyLambdaFunction"
```

- Untuk detail API, lihat [ListTargetsByRule](#) di Referensi AWS CLI Perintah.

## put-events

Contoh kode berikut menunjukkan cara menggunakan `put-events`.

### AWS CLI

Untuk mengirim acara khusus ke CloudWatch Acara

Contoh ini mengirimkan acara kustom ke CloudWatch Events. Acara ini terkandung dalam file `putevents.json`:

```
aws events put-events --entries file://putevents.json
```

Berikut adalah isi dari file `putevents.json`:

```
[
  {
    "Source": "com.mycompany.myapp",
    "Detail": "{ \"key1\": \"value1\", \"key2\": \"value2\" }",
    "Resources": [
```

```

    "resource1",
    "resource2"
  ],
  "DetailType": "myDetailType"
},
{
  "Source": "com.mycompany.myapp",
  "Detail": "{ \"key1\": \"value3\", \"key2\": \"value4\" }",
  "Resources": [
    "resource1",
    "resource2"
  ],
  "DetailType": "myDetailType"
}
]

```

- Untuk detail API, lihat [PutEvents](#) di Referensi AWS CLI Perintah.

## put-rule

Contoh kode berikut menunjukkan cara menggunakan `put-rule`.

### AWS CLI

Untuk membuat aturan CloudWatch Acara

Contoh ini membuat aturan yang dipicu setiap hari pada pukul 9:00 pagi (UTC). Jika Anda menggunakan `put-target` untuk menambahkan fungsi Lambda sebagai target aturan ini, Anda dapat menjalankan fungsi Lambda setiap hari pada waktu yang ditentukan:

```
aws events put-rule --name "DailyLambdaFunction" --schedule-expression "cron(0 9 * * ? *)"
```

Contoh ini membuat aturan yang memicu ketika instans EC2 di wilayah mengubah status:

```
aws events put-rule --name "EC2InstanceStateChanges" --event-pattern "{\"source\": [\"aws.ec2\"], \"detail-type\": [\"EC2 Instance State-change Notification\"]}" --role-arn "arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

Contoh ini membuat aturan yang memicu ketika instans EC2 di wilayah tersebut dihentikan atau dihentikan:

```
aws events put-rule --name "EC2InstanceStateChangeStopOrTerminate" --event-pattern "{\"source\":[\"aws.ec2\"],\"detail-type\":[\"EC2 Instance State-change Notification\"],\"detail\":{\"state\":[\"stopped\",\"terminated\"]}}" --role-arn "arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

- Untuk detail API, lihat [PutRule](#) di Referensi AWS CLI Perintah.

## put-targets

Contoh kode berikut menunjukkan cara menggunakan `put-targets`.

### AWS CLI

Untuk menambahkan target untuk aturan CloudWatch Acara

Contoh ini menambahkan fungsi Lambda sebagai target aturan:

```
aws events put-targets --rule DailyLambdaFunction --targets "Id"="1", "Arn"="arn:aws:lambda:us-east-1:123456789012:function:MyFunctionName"
```

Contoh ini menetapkan aliran Amazon Kinesis sebagai target, sehingga peristiwa yang ditangkap oleh aturan ini diteruskan ke aliran:

```
aws events put-targets --rule EC2InstanceStateChanges --targets "Id"="1", "Arn"="arn:aws:kinesis:us-east-1:123456789012:stream/MyStream", "RoleArn"="arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

Contoh ini menetapkan dua aliran Amazon Kinesis sebagai target untuk satu aturan:

```
aws events put-targets --rule DailyLambdaFunction --targets "Id"="Target1", "Arn"="arn:aws:kinesis:us-east-1:379642911888:stream/MyStream1", "RoleArn"="arn:aws:iam::379642911888:role/MyRoleToAccessLambda" "Id"="Target2", "Arn"="arn:aws:kinesis:us-east-1:379642911888:stream/MyStream2", "RoleArn"="arn:aws:iam::379642911888:role/MyRoleToAccessLambda"
```

- Untuk detail API, lihat [PutTargets](#) di Referensi AWS CLI Perintah.

## remove-targets

Contoh kode berikut menunjukkan cara menggunakan `remove-targets`.

## AWS CLI

Untuk menghapus target untuk suatu acara

Contoh ini menghapus aliran Amazon Kinesis bernama MyStream 1 dari target aturan. DailyLambdaFunction Saat DailyLambdaFunction dibuat, aliran ini ditetapkan sebagai target dengan ID Target1:

```
aws events remove-targets --rule "DailyLambdaFunction" --ids "Target1"
```

- Untuk detail API, lihat [RemoveTargets](#) di Referensi AWS CLI Perintah.

## test-event-pattern

Contoh kode berikut menunjukkan cara menggunakan `test-event-pattern`.

### AWS CLI

Untuk memeriksa apakah pola acara cocok dengan peristiwa tertentu

Contoh ini menguji apakah pola “source:com.mycompany.myapp” cocok dengan peristiwa yang ditentukan. Dalam contoh ini, outputnya akan menjadi “true”:

```
aws events test-event-pattern --event-pattern "{\"source\": [\"com.mycompany.myapp\"]}" --event "{\"id\": \"1\", \"source\": \"com.mycompany.myapp\", \"detail-type\": \"myDetailType\", \"account\": \"123456789012\", \"region\": \"us-east-1\", \"time\": \"2017-04-11T20:11:04Z\"}"
```

- Untuk detail API, lihat [TestEventPattern](#) di Referensi AWS CLI Perintah.

## Contoh Firewall Manager menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Firewall Manager.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `associate-admin-account`.

AWS CLI

Untuk mengatur akun administrator Firewall Manager

`associate-admin-account` Contoh berikut menetapkan akun administrator untuk Firewall Manager.

```
aws fms associate-admin-account \  
  --admin-account 123456789012
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengatur Akun Administrator Manajer AWS Firewall](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [AssociateAdminAccount](#) di Referensi AWS CLI Perintah.

### **delete-notification-channel**

Contoh kode berikut menunjukkan cara menggunakan `delete-notification-channel`.

AWS CLI

Untuk menghapus informasi topik SNS untuk log Firewall Manager

`delete-notification-channel` Contoh berikut menghapus informasi topik SNS.

```
aws fms delete-notification-channel
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi Notifikasi Amazon SNS dan CloudWatch Alarm Amazon](#) di AWS WAF, Firewall Manager, dan Panduan AWS Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DeleteNotificationChannel](#) di Referensi AWS CLI Perintah.

## **delete-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-policy`.

### AWS CLI

Untuk menghapus kebijakan Firewall Manager

`delete-policy` Contoh berikut menghapus kebijakan dengan ID yang ditentukan, beserta semua sumber dayanya.

```
aws fms delete-policy \  
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --delete-all-policy-resources
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Kebijakan AWS Firewall Manager](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## **disassociate-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-admin-account`.

### AWS CLI

Untuk menghapus akun administrator Firewall Manager

`disassociate-admin-account` Contoh berikut menghapus asosiasi akun administrator saat ini dari Firewall Manager.

```
aws fms disassociate-admin-account
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengatur Akun Administrator Manajer AWS Firewall](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DisassociateAdminAccount](#) di Referensi AWS CLI Perintah.

## get-admin-account

Contoh kode berikut menunjukkan cara menggunakan `get-admin-account`.

### AWS CLI

Untuk mengambil akun administrator Firewall Manager

`get-admin-account` Contoh berikut mengambil akun administrator.

```
aws fms get-admin-account
```

Output:

```
{
  "AdminAccount": "123456789012",
  "RoleStatus": "READY"
}
```

Untuk informasi selengkapnya, lihat [Prasyarat AWS Firewall Manager](#) di WAF AWS, Firewall AWS Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetAdminAccount](#) di Referensi AWS CLI Perintah.

## get-compliance-detail

Contoh kode berikut menunjukkan cara menggunakan `get-compliance-detail`.

### AWS CLI

Untuk mengambil informasi kepatuhan untuk akun



`get-compliance-detail` Contoh berikut mengambil informasi kepatuhan untuk kebijakan dan akun anggota yang ditentukan.

```
aws fms get-compliance-detail \  
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --member-account 123456789012
```

Output:

```
{  
  "PolicyComplianceDetail": {  
    "EvaluationLimitExceeded": false,  
    "IssueInfoMap": {},  
    "MemberAccount": "123456789012",  
    "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "PolicyOwner": "123456789012",  
    "Violators": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Melihat Kepatuhan Sumber Daya dengan Kebijakan](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [GetComplianceDetail](#) di Referensi AWS CLI Perintah.

## `get-notification-channel`

Contoh kode berikut menunjukkan cara menggunakan `get-notification-channel`.

AWS CLI

Untuk mengambil informasi topik SNS untuk log Firewall Manager

`get-notification-channel` Contoh berikut mengambil informasi topik SNS.

```
aws fms get-notification-channel
```

Output:

```
{  
  "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:us-west-2-fms",  
  "SnsRoleName": "arn:aws:iam::123456789012:role/aws-service-role/  
fms.amazonaws.com/AWSServiceRoleForFMS"
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Notifikasi Amazon SNS dan CloudWatch Alarm Amazon](#) di AWS WAF, Firewall Manager, dan Panduan AWS Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [GetNotificationChannel](#) di Referensi AWS CLI Perintah.

## get-policy

Contoh kode berikut menunjukkan cara menggunakan `get-policy`.

### AWS CLI

Untuk mengambil kebijakan Firewall Manager

`get-policy` Contoh berikut mengambil kebijakan dengan ID yang ditentukan.

```
aws fms get-policy \
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
  "Policy": {
    "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "PolicyName": "test",
    "PolicyUpdateToken": "1:p+2RpKR4wPFx7mcrL1U0QQ==",
    "SecurityServicePolicyData": {
      "Type": "SECURITY_GROUPS_COMMON",
      "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_COMMON\",
\\revertManualSecurityGroupChanges\":true,\\exclusiveResourceSecurityGroupManagement
\":false,\\securityGroups\":[{\"id\":\"sg-045c43ccc9724e63e\"}]}"
    },
    "ResourceType": "AWS::EC2::Instance",
    "ResourceTags": [],
    "ExcludeResourceTags": false,
    "RemediationEnabled": false
  },
  "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/d1ac59b8-938e-42b3-
b2e0-7c620422ddc2"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kebijakan AWS Firewall Manager](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetPolicy](#) di Referensi AWS CLI Perintah.

## list-compliance-status

Contoh kode berikut menunjukkan cara menggunakan `list-compliance-status`.

### AWS CLI

Untuk mengambil informasi kepatuhan kebijakan untuk akun anggota

`list-compliance-status` Contoh berikut mengambil informasi kepatuhan akun anggota untuk kebijakan yang ditentukan.

```
aws fms list-compliance-status \  
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

### Output:

```
{  
  "PolicyComplianceStatusList": [  
    {  
      "PolicyOwner": "123456789012",  
      "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "PolicyName": "test",  
      "MemberAccount": "123456789012",  
      "EvaluationResults": [  
        {  
          "ComplianceStatus": "COMPLIANT",  
          "ViolatorCount": 0,  
          "EvaluationLimitExceeded": false  
        },  
        {  
          "ComplianceStatus": "NON_COMPLIANT",  
          "ViolatorCount": 2,  
          "EvaluationLimitExceeded": false  
        }  
      ],  
      "LastUpdated": 1576283774.0,  
      "IssueInfoMap": {}  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat Kepatuhan Sumber Daya dengan Kebijakan](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [ListComplianceStatus](#) di Referensi AWS CLI Perintah.

## **list-member-accounts**

Contoh kode berikut menunjukkan cara menggunakan `list-member-accounts`.

### AWS CLI

Untuk mengambil akun anggota di organisasi

`list-member-accounts` Contoh berikut mencantumkan semua akun anggota yang ada di organisasi administrator Firewall Manager.

```
aws fms list-member-accounts
```

Output:

```
{  
  "MemberAccounts": [  
    "222222222222",  
    "333333333333",  
    "444444444444"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [AWS Firewall Manager](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [ListMemberAccounts](#) di Referensi AWS CLI Perintah.

## **list-policies**

Contoh kode berikut menunjukkan cara menggunakan `list-policies`.

## AWS CLI

Untuk mengambil semua kebijakan Firewall Manager

`list-policies` Contoh berikut mengambil daftar kebijakan untuk akun. Dalam contoh ini, output dibatasi hingga dua hasil per permintaan. Setiap panggilan mengembalikan sebuah `NextToken` yang dapat digunakan sebagai nilai untuk `--starting-token` parameter dalam `list-policies` panggilan berikutnya untuk mendapatkan set hasil berikutnya untuk daftar.

```
aws fms list-policies \  
  --max-items 2
```

Output:

```
{  
  "PolicyList": [  
    {  
      "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "PolicyName": "test",  
      "ResourceType": "AWS::EC2::Instance",  
      "SecurityServiceType": "SECURITY_GROUPS_COMMON",  
      "RemediationEnabled": false  
    },  
    {  
      "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "PolicyId": "457c9b21-fc94-406c-ae63-21217395ba72",  
      "PolicyName": "test",  
      "ResourceType": "AWS::EC2::Instance",  
      "SecurityServiceType": "SECURITY_GROUPS_COMMON",  
      "RemediationEnabled": false  
    }  
  ],  
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kebijakan AWS Firewall Manager](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [ListPolicies](#) di Referensi AWS CLI Perintah.

## put-notification-channel

Contoh kode berikut menunjukkan cara menggunakan `put-notification-channel`.

### AWS CLI

Untuk mengatur informasi topik SNS untuk log Firewall Manager

`put-notification-channel` Contoh berikut menetapkan informasi topik SNS.

```
aws fms put-notification-channel \  
  --sns-topic-arn arn:aws:sns:us-west-2:123456789012:us-west-2-fms \  
  --sns-role-name arn:aws:iam::123456789012:role/aws-service-role/  
fms.amazonaws.com/AWSServiceRoleForFMS
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi Notifikasi Amazon SNS dan CloudWatch Alarm Amazon](#) di AWS WAF, Firewall Manager, dan Panduan AWS Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [PutNotificationChannel](#) di Referensi AWS CLI Perintah.

## put-policy

Contoh kode berikut menunjukkan cara menggunakan `put-policy`.

### AWS CLI

Untuk membuat kebijakan Firewall Manager

`put-policy` Contoh berikut membuat kebijakan grup keamanan Firewall Manager.

```
aws fms put-policy \  
  --cli-input-json file://policy.json
```

Isi dari `policy.json`:

```
{  
  "Policy": {  
    "PolicyName": "test",  
    "SecurityServicePolicyData": {
```

```

        "Type": "SECURITY_GROUPS_USAGE_AUDIT",
        "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_USAGE_AUDIT\",
\\\"deleteUnusedSecurityGroups\\\":false,\\\"coalesceRedundantSecurityGroups\\\":true}"
    },
    "ResourceType": "AWS::EC2::SecurityGroup",
    "ResourceTags": [],
    "ExcludeResourceTags": false,
    "RemediationEnabled": false
},
"TagList": [
  {
    "Key": "foo",
    "Value": "foo"
  }
]
}

```

### Output:

```

{
  "Policy": {
    "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "PolicyName": "test",
    "PolicyUpdateToken": "1:X9QGexP7HASDlsFp+G31Iw==",
    "SecurityServicePolicyData": {
      "Type": "SECURITY_GROUPS_USAGE_AUDIT",
      "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_USAGE_AUDIT\",
\\\"deleteUnusedSecurityGroups\\\":false,\\\"coalesceRedundantSecurityGroups\\\":true,
\\\"optionalDelayForUnusedInMinutes\\\":null}"
    },
    "ResourceType": "AWS::EC2::SecurityGroup",
    "ResourceTags": [],
    "ExcludeResourceTags": false,
    "RemediationEnabled": false
  },
  "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Kebijakan AWS Firewall Manager](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [PutPolicy](#) di Referensi AWS CLI Perintah.

## AWS FIS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS FIS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-experiment-template**

Contoh kode berikut menunjukkan cara menggunakan `create-experiment-template`.

AWS CLI

Untuk membuat template percobaan

`create-experiment-template` Contoh berikut membuat template percobaan di akun AWS FIS Anda.

```
aws fis create-experiment-template \  
  --cli-input-json file://myfile.json
```

Isi dari `myfile.json`:

```
{  
  "description": "experimentTemplate",  
  "stopConditions": [  
    {
```



```

        "source": "aws:cloudwatch:alarm",
        "value": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:alarmName"
    }
],
"targets": {
    "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",
        "resourceArns": [
            "arn:aws:ec2:us-west-2:123456789012:instance/i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
    }
},
"actions": {
    "reboot": {
        "actionId": "aws:ec2:reboot-instances",
        "description": "reboot",
        "parameters": {},
        "targets": {
            "Instances": "Instances-Target-1"
        }
    }
},
"roleArn": "arn:aws:iam::123456789012:role/myRole"
}

```

### Output:

```

{
  "experimentTemplate": {
    "id": "ABCDE1fgHIJKLmNop",
    "description": "experimentTemplate",
    "targets": {
      "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",
        "resourceArns": [
          "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
      }
    },
    "actions": {

```

```
    "reboot": {
      "actionId": "aws:ec2:reboot-instances",
      "description": "reboot",
      "parameters": {},
      "targets": {
        "Instances": "Instances-Target-1"
      }
    },
    "stopConditions": [
      {
        "source": "aws:cloudwatch:alarm",
        "value": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:alarmName"
      }
    ],
    "creationTime": 1616434850.659,
    "lastUpdateTime": 1616434850.659,
    "roleArn": "arn:aws:iam::123456789012:role/myRole",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat template eksperimen](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [CreateExperimentTemplate](#) di Referensi AWS CLI Perintah.

## delete-experiment-template

Contoh kode berikut menunjukkan cara menggunakan `delete-experiment-template`.

### AWS CLI

Untuk menghapus template percobaan

`delete-experiment-template` Contoh berikut menghapus template percobaan yang ditentukan.

```
aws fis delete-experiment-template \
  --id ABCDE1fgHIJKLmNop
```

Output:

```
{
  "experimentTemplate": {
    "id": "ABCDE1fgHIJkLmNop",
    "description": "myExperimentTemplate",
    "targets": {
      "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",
        "resourceArns": [
          "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
      }
    },
    "actions": {
      "testaction": {
        "actionId": "aws:ec2:stop-instances",
        "parameters": {},
        "targets": {
          "Instances": "Instances-Target-1"
        }
      }
    },
    "stopConditions": [
      {
        "source": "none"
      }
    ],
    "creationTime": 1616017191.124,
    "lastUpdateTime": 1616017859.607,
    "roleArn": "arn:aws:iam::123456789012:role/FISRole"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus templat eksperimen](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [DeleteExperimentTemplate](#) di Referensi AWS CLI Perintah.

## get-action

Contoh kode berikut menunjukkan cara menggunakan `get-action`.

## AWS CLI

Untuk mendapatkan detail tindakan

`get-action` Contoh berikut mendapatkan rincian dari tindakan yang ditentukan.

```
aws fis get-action \  
  --id aws:ec2:stop-instances
```

Output:

```
{  
  "action": {  
    "id": "aws:ec2:stop-instances",  
    "description": "Stop the specified EC2 instances.",  
    "parameters": {  
      "startInstancesAfterDuration": {  
        "description": "The time to wait before restarting the instances  
(ISO 8601 duration).",  
        "required": false  
      }  
    },  
    "targets": {  
      "Instances": {  
        "resourceType": "aws:ec2:instance"  
      }  
    },  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Tindakan](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [GetAction](#) di Referensi AWS CLI Perintah.

## **get-experiment-template**

Contoh kode berikut menunjukkan cara menggunakan `get-experiment-template`.

## AWS CLI

Untuk mendapatkan detail template eksperimen

`get-experiment-template` Contoh berikut mendapatkan rincian template percobaan yang ditentukan.

```
aws fis get-experiment-template \  
  --id ABCDE1fgHIJkLmNop
```

Output:

```
{  
  "experimentTemplate": {  
    "id": "ABCDE1fgHIJkLmNop",  
    "description": "myExperimentTemplate",  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
    "actions": {  
      "testaction": {  
        "actionId": "aws:ec2:stop-instances",  
        "parameters": {},  
        "targets": {  
          "Instances": "Instances-Target-1"  
        }  
      }  
    },  
    "stopConditions": [  
      {  
        "source": "none"  
      }  
    ],  
    "creationTime": 1616017191.124,  
    "lastUpdateTime": 1616017331.51,  
    "roleArn": "arn:aws:iam::123456789012:role/FISRole",  
    "tags": {  
      "key": "value"  
    }  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Templat eksperimen](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [GetExperimentTemplate](#) di Referensi AWS CLI Perintah.

## get-experiment

Contoh kode berikut menunjukkan cara menggunakan `get-experiment`.

### AWS CLI

Untuk mendapatkan detail eksperimen

`get-experiment` Contoh berikut mendapatkan rincian percobaan yang ditentukan.

```
aws fis get-experiment \  
  --id ABC12DeFGhI3jKLMNOP
```

Output:

```
{  
  "experiment": {  
    "id": "ABC12DeFGhI3jKLMNOP",  
    "experimentTemplateId": "ABCDE1fgHIJkLmNop",  
    "roleArn": "arn:aws:iam::123456789012:role/myRole",  
    "state": {  
      "status": "completed",  
      "reason": "Experiment completed."  
    },  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
    "actions": {  
      "reboot": {
```

```
    "actionId": "aws:ec2:reboot-instances",
    "parameters": {},
    "targets": {
      "Instances": "Instances-Target-1"
    },
    "state": {
      "status": "completed",
      "reason": "Action was completed."
    }
  },
  "stopConditions": [
    {
      "source": "none"
    }
  ],
  "creationTime": 1616432509.662,
  "startTime": 1616432509.962,
  "endTime": 1616432522.307,
  "tags": {}
}
}
```

Untuk informasi selengkapnya, lihat [Eksperimen untuk AWS FIS](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [GetExperiment](#) di Referensi AWS CLI Perintah.

## list-actions

Contoh kode berikut menunjukkan cara menggunakan `list-actions`.

### AWS CLI

Untuk membuat daftar tindakan

`list-actions` Contoh berikut mencantumkan tindakan yang tersedia.

```
aws fis list-actions
```

Output:

```
{
```

```
"actions": [  
  {  
    "id": "aws:ec2:reboot-instances",  
    "description": "Reboot the specified EC2 instances.",  
    "targets": {  
      "Instances": {  
        "resourceType": "aws:ec2:instance"  
      }  
    },  
    "tags": {}  
  },  
  {  
    "id": "aws:ec2:stop-instances",  
    "description": "Stop the specified EC2 instances.",  
    "targets": {  
      "Instances": {  
        "resourceType": "aws:ec2:instance"  
      }  
    },  
    "tags": {}  
  },  
  {  
    "id": "aws:ec2:terminate-instances",  
    "description": "Terminate the specified EC2 instances.",  
    "targets": {  
      "Instances": {  
        "resourceType": "aws:ec2:instance"  
      }  
    },  
    "tags": {}  
  },  
  {  
    "id": "aws:ecs:drain-container-instances",  
    "description": "Drain percentage of underlying EC2 instances on an ECS  
cluster.",  
    "targets": {  
      "Clusters": {  
        "resourceType": "aws:ecs:cluster"  
      }  
    },  
    "tags": {}  
  },  
  {  
    "id": "aws:eks:terminate-nodegroup-instances",
```



```
    "description": "Terminates a percentage of the underlying EC2 instances
in an EKS cluster.",
    "targets": {
        "Nodegroups": {
            "resourceType": "aws:eks:nodegroup"
        }
    },
    "tags": {}
},
{
    "id": "aws:fis:inject-api-internal-error",
    "description": "Cause an AWS service to return internal error responses
for specific callers and operations.",
    "targets": {
        "Roles": {
            "resourceType": "aws:iam:role"
        }
    },
    "tags": {}
},
{
    "id": "aws:fis:inject-api-throttle-error",
    "description": "Cause an AWS service to return throttled responses for
specific callers and operations.",
    "targets": {
        "Roles": {
            "resourceType": "aws:iam:role"
        }
    },
    "tags": {}
},
{
    "id": "aws:fis:inject-api-unavailable-error",
    "description": "Cause an AWS service to return unavailable error
responses for specific callers and operations.",
    "targets": {
        "Roles": {
            "resourceType": "aws:iam:role"
        }
    },
    "tags": {}
},
{
    "id": "aws:fis:wait",
```

```
    "description": "Wait for the specified duration. Stop condition
monitoring will continue during this time.",
    "tags": {}
  },
  {
    "id": "aws:rds:failover-db-cluster",
    "description": "Failover a DB Cluster to one of the replicas.",
    "targets": {
      "Clusters": {
        "resourceType": "aws:rds:cluster"
      }
    },
    "tags": {}
  },
  {
    "id": "aws:rds:reboot-db-instances",
    "description": "Reboot the specified DB instances.",
    "targets": {
      "DBInstances": {
        "resourceType": "aws:rds:db"
      }
    },
    "tags": {}
  },
  {
    "id": "aws:ssm:send-command",
    "description": "Run the specified SSM document.",
    "targets": {
      "Instances": {
        "resourceType": "aws:ec2:instance"
      }
    },
    "tags": {}
  }
]
}
```

Untuk informasi selengkapnya, lihat [Tindakan](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [ListActions](#) di Referensi AWS CLI Perintah.

## list-experiment-templates

Contoh kode berikut menunjukkan cara menggunakan `list-experiment-templates`.

### AWS CLI

Untuk membuat daftar templat eksperimen

`list-experiment-templates` Contoh berikut mencantumkan templat eksperimen di AWS akun Anda.

```
aws fis list-experiment-templates
```

Output:

```
{
  "experimentTemplates": [
    {
      "id": "ABCDE1fgHIJkLmNop",
      "description": "myExperimentTemplate",
      "creationTime": 1616017191.124,
      "lastUpdateTime": 1616017191.124,
      "tags": {
        "key": "value"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Templat eksperimen](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [ListExperimentTemplates](#) di Referensi AWS CLI Perintah.

## list-experiments

Contoh kode berikut menunjukkan cara menggunakan `list-experiments`.

### AWS CLI

Untuk membuat daftar eksperimen

`list-experiments` Contoh berikut mencantumkan eksperimen di AWS akun Anda.

```
aws fis list-experiments
```

Output:

```
{
  "experiments": [
    {
      "id": "ABCdeF1GHiJkLM23N0",
      "experimentTemplateId": "ABCDE1fgHIJkLmNop",
      "state": {
        "status": "running",
        "reason": "Experiment is running."
      },
      "creationTime": 1616017341.197,
      "tags": {
        "key": "value"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Eksperimen](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [ListExperiments](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk sumber daya yang ditentukan.

```
aws fis list-tags-for-resource \
  --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
```

**Output:**

```
{
  "tags": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS FIS Anda](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

**start-experiment**

Contoh kode berikut menunjukkan cara menggunakan `start-experiment`.

**AWS CLI**

Untuk memulai percobaan

`start-experiment` Contoh berikut memulai percobaan yang ditentukan.

```
aws fis start-experiment \
  --experiment-template-id ABCDE1fgHIJkLmNop
```

**Output:**

```
{
  "experiment": {
    "id": "ABC12DeFGhI3jKLMNOP",
    "experimentTemplateId": "ABCDE1fgHIJkLmNop",
    "roleArn": "arn:aws:iam::123456789012:role/myRole",
    "state": {
      "status": "initiating",
      "reason": "Experiment is initiating."
    },
    "targets": {
      "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",

```

```
        "resourceArns": [
            "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
    }
},
"actions": {
    "reboot": {
        "actionId": "aws:ec2:reboot-instances",
        "parameters": {},
        "targets": {
            "Instances": "Instances-Target-1"
        },
        "state": {
            "status": "pending",
            "reason": "Initial state"
        }
    }
},
"stopConditions": [
    {
        "source": "none"
    }
],
"creationTime": 1616432464.025,
"startTime": 1616432464.374,
"tags": {}
}
}
```

Untuk informasi selengkapnya, lihat [Eksperimen untuk AWS FIS](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [StartExperiment](#) di Referensi AWS CLI Perintah.

## stop-experiment

Contoh kode berikut menunjukkan cara menggunakan `stop-experiment`.

### AWS CLI

Untuk menghentikan percobaan

`stop-experiment` Contoh berikut menghentikan percobaan yang ditentukan agar tidak berjalan.

```
aws fis stop-experiment \  
  --id ABC12DeFGhI3jKLMNOP
```

Output:

```
{  
  "experiment": {  
    "id": "ABC12DeFGhI3jKLMNOP",  
    "experimentTemplateId": "ABCDE1fgHIJkLmNop",  
    "roleArn": "arn:aws:iam::123456789012:role/myRole",  
    "state": {  
      "status": "stopping",  
      "reason": "Stopping Experiment."  
    },  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
    "actions": {  
      "reboot": {  
        "actionId": "aws:ec2:reboot-instances",  
        "parameters": {},  
        "targets": {  
          "Instances": "Instances-Target-1"  
        },  
        "startAfter": [  
          "wait"  
        ],  
        "state": {  
          "status": "pending",  
          "reason": "Initial state."  
        }  
      },  
      "wait": {  
        "actionId": "aws:fis:wait",
```

```
        "parameters": {
            "duration": "PT5M"
        },
        "state": {
            "status": "running",
            "reason": ""
        }
    }
},
"stopConditions": [
    {
        "source": "none"
    }
],
"creationTime": 1616432680.927,
"startTime": 1616432681.177,
"tags": {}
}
}
```

Untuk informasi selengkapnya, lihat [Eksperimen untuk AWS FIS](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [StopExperiment](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menandai sumber daya yang ditentukan.

```
aws fis tag-resource \  
  --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP \  
 \  
  --tags key1=value1,key2=value2
```

Perintah ini tidak menghasilkan output.



Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS FIS Anda](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag sumber daya

`untag-resource` Contoh berikut menghapus tag dari sumber daya yang ditentukan.

```
aws fis untag-resource \  
  --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS FIS Anda](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-experiment-template**

Contoh kode berikut menunjukkan cara menggunakan `update-experiment-template`.

### AWS CLI

Untuk memperbarui template eksperimen

`update-experiment-template` Contoh berikut memperbarui deskripsi template percobaan yang ditentukan.

```
aws fis update-experiment-template \  
  --id ABCDE1fgHIJkLmNop \  
  ---description myExperimentTemplate
```

Output:

```
{
  "experimentTemplate": {
    "id": "ABCDE1fgHIJkLmNop",
    "description": "myExperimentTemplate",
    "targets": {
      "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",
        "resourceArns": [
          "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
      }
    },
    "actions": {
      "testaction": {
        "actionId": "aws:ec2:stop-instances",
        "parameters": {},
        "targets": {
          "Instances": "Instances-Target-1"
        }
      }
    },
    "stopConditions": [
      {
        "source": "none"
      }
    ],
    "creationTime": 1616017191.124,
    "lastUpdateTime": 1616017859.607,
    "roleArn": "arn:aws:iam::123456789012:role/FISRole",
    "tags": {
      "key": "value"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui templat eksperimen](#) di Panduan Pengguna Simulator Injeksi AWS Kesalahan.

- Untuk detail API, lihat [UpdateExperimentTemplate](#) di Referensi AWS CLI Perintah.

## GameLift Contoh Amazon menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon GameLift.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-build**

Contoh kode berikut menunjukkan cara menggunakan `create-build`.

AWS CLI

Contoh1: Untuk membuat game build dari file di bucket S3

`create-build` Contoh berikut membuat sumber daya build game kustom. Ini menggunakan file zip yang disimpan di lokasi S3 di AWS akun yang Anda kontrol. Contoh ini mengasumsikan bahwa Anda telah membuat peran IAM yang memberikan GameLift izin Amazon untuk mengakses lokasi S3. Karena permintaan tidak menentukan sistem operasi, sumber daya build baru default ke `WINDOWS_2012`.

```
aws gamelift create-build \  
  --storage-location file://storage-loc.json \  
  --name MegaFrogRaceServer.NA \  
  --build-version 12345.678
```

Isi dari `storage-loc.json`:

```
{
  "Bucket": "MegaFrogRaceServer_NA_build_files"
  "Key": "MegaFrogRaceServer_build_123.zip"
  "RoleArn": "arn:aws:iam::123456789012:role/gamelift"
}
```

Output:

```
{
  "Build": {
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreationTime": 1496708916.18,
    "Name": "MegaFrogRaceServer.NA",
    "OperatingSystem": "WINDOWS_2012",
    "SizeOnDisk": 479303,
    "Status": "INITIALIZED",
    "Version": "12345.678"
  },
  "StorageLocation": {
    "Bucket": "MegaFrogRaceServer_NA_build_files",
    "Key": "MegaFrogRaceServer_build_123.zip"
  }
}
```

Contoh2: Untuk membuat sumber daya pembuatan game untuk mengunggah file secara manual GameLift

`create-build`Contoh berikut membuat sumber daya build baru. Ini juga mendapat lokasi penyimpanan dan kredensial sementara yang memungkinkan Anda mengunggah build game secara manual ke GameLift lokasi di Amazon S3. Setelah berhasil mengunggah build, GameLift layanan memvalidasi build dan memperbarui status build baru.

```
aws gamelift create-build \
  --name MegaFrogRaceServer.NA \
  --build-version 12345.678 \
  --operating-system AMAZON_LINUX
```

Output:

```
{
  "Build": {
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreationTime": 1496708916.18,
    "Name": "MegaFrogRaceServer.NA",
    "OperatingSystem": "AMAZON_LINUX",
    "SizeOnDisk": 0,
    "Status": "INITIALIZED",
    "Version": "12345.678"
  },
  "StorageLocation": {
    "Bucket": "gamelift-builds-us-west-2",
    "Key": "123456789012/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  },
  "UploadCredentials": {
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "SessionToken": "AgoGb3JpZ2luENZ...EXAMPLETOKEN=="
  }
}
```

Untuk informasi selengkapnya, lihat [Mengunggah Pembuatan Server Kustom ke GameLift](#) dalam Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [CreateBuild](#) di Referensi AWS CLI Perintah.

## create-fleet

Contoh kode berikut menunjukkan cara menggunakan `create-fleet`.

### AWS CLI

Contoh 1: Untuk membuat armada Linux dasar

`create-fleet` Contoh berikut membuat armada instans Linux on-demand yang dikonfigurasi minimal untuk meng-host build server kustom. Anda dapat menyelesaikan konfigurasi dengan menggunakan `update-fleet`.

```
aws gamelift create-fleet \
  --name MegaFrogRaceServer.NA.v2 \
```

```
--description 'Hosts for v2 North America' \
--build-id build-1111aaaa-22bb-33cc-44dd-5555eeee66ff \
--certificate-configuration 'CertificateType=GENERATED' \
--ec2-instance-type c4.large \
--fleet-type ON_DEMAND \
--runtime-configuration 'ServerProcesses=[{LaunchPath=/local/game/release-na/
MegaFrogRace_Server.exe,ConcurrentExecutions=1}]'
```

### Output:

```
{
  "FleetAttributes": {
    "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    },
    "CreationTime": 1496365885.44,
    "Description": "Hosts for v2 North America",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetType": "ON_DEMAND",
    "InstanceType": "c4.large",
    "MetricGroups": ["default"],
    "Name": "MegaFrogRace.NA.v2",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "AMAZON_LINUX",
    "ServerLaunchPath": "/local/game/release-na/MegaFrogRace_Server.exe",
    "Status": "NEW"
  }
}
```

### Contoh 2: Untuk membuat armada Windows dasar

`create-fleet` Contoh berikut membuat armada instance Windows spot yang dikonfigurasi minimal untuk meng-host build server kustom. Anda dapat menyelesaikan konfigurasi dengan menggunakan `update-fleet`.

```
aws gamelift create-fleet \
--name MegaFrogRace.NA.v2 \
--description 'Hosts for v2 North America' \
--build-id build-2222aaaa-33bb-44cc-55dd-6666eeee77ff \
```

```
--certificate-configuration 'CertificateType=GENERATED' \
--ec2-instance-type c4.large \
--fleet-type SPOT \
--runtime-configuration 'ServerProcesses=[{LaunchPath=C:\game
\Bin64.Release.Dedicated\MegaFrogRace_Server.exe,ConcurrentExecutions=1}]'
```

### Output:

```
{
  "FleetAttributes": {
    "BuildId": "build-2222aaaa-33bb-44cc-55dd-6666eeee77ff",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    },
    "CreationTime": 1496365885.44,
    "Description": "Hosts for v2 North America",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetType": "SPOT",
    "InstanceType": "c4.large",
    "MetricGroups": ["default"],
    "Name": "MegaFrogRace.NA.v2",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "WINDOWS_2012",
    "ServerLaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
    "Status": "NEW"
  }
}
```

### Contoh 3: Untuk membuat armada yang sepenuhnya dikonfigurasi

`create-fleet` Contoh berikut membuat armada instance Spot Windows untuk build server kustom, dengan pengaturan konfigurasi yang paling umum digunakan disediakan.

```
aws gamelift create-fleet \
--name MegaFrogRace.NA.v2 \
--description 'Hosts for v2 North America' \
--build-id build-2222aaaa-33bb-44cc-55dd-6666eeee77ff \
--certificate-configuration 'CertificateType=GENERATED' \
--ec2-instance-type c4.large \
```

```

--ec2-inbound-permissions
'FromPort=33435,ToPort=33435,IpRange=10.24.34.0/23,Protocol=UDP' \
--fleet-type SPOT \
--new-game-session-protection-policy FullProtection \
--runtime-configuration file://runtime-config.json \
--metric-groups default \
--instance-role-arn 'arn:aws:iam::444455556666:role/GameLiftS3Access'

```

Isi dari runtime-config.json:

```

GameSessionActivationTimeoutSeconds=300,
MaxConcurrentGameSessionActivations=2,
ServerProcesses=[
  {LaunchPath=C:\game\Bin64.Release.Dedicated\MegaFrogRace_Server.exe,Parameters=-
debug,ConcurrentExecutions=1},
  {LaunchPath=C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe,ConcurrentExecutions=1}]

```

Output:

```

{
  "FleetAttributes": {
    "InstanceRoleArn": "arn:aws:iam::444455556666:role/GameLiftS3Access",
    "Status": "NEW",
    "InstanceType": "c4.large",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "Description": "Hosts for v2 North America",
    "FleetType": "SPOT",
    "OperatingSystem": "WINDOWS_2012",
    "Name": "MegaFrogRace.NA.v2",
    "CreationTime": 1569309011.11,
    "MetricGroups": [
      "default"
    ],
    "BuildId": "build-2222aaaa-33bb-44cc-55dd-6666eeee77ff",
    "ServerLaunchParameters": "abc",
    "ServerLaunchPath": "C:\\game\\Bin64.Release.Dedicated\\
\MegaFrogRace_Server.exe",
    "NewGameSessionProtectionPolicy": "FullProtection",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    }
  }
}

```



```

    }
  }
}

```

#### Contoh 4: Untuk membuat armada Server Realtime

`create-fleet` Contoh berikut membuat armada instance Spot dengan skrip konfigurasi Realtime yang telah diunggah ke Amazon. GameLift Semua server Realtime dikerahkan ke mesin Linux. Untuk keperluan contoh ini, asumsikan bahwa skrip Realtime yang diunggah mencakup beberapa file skrip, dengan `Init()` fungsi yang terletak di file skrip yang disebut `MainScript.js` Seperti yang ditunjukkan, file ini diidentifikasi sebagai skrip peluncuran dalam konfigurasi runtime.

```

aws gamelift create-fleet \
  --name MegaFrogRace.NA.realtime \
  --description 'Mega Frog Race Realtime fleet' \
  --script-id script-1111aaaa-22bb-33cc-44dd-5555eeee66ff \
  --ec2-instance-type c4.large \
  --fleet-type SPOT \
  --certificate-configuration 'CertificateType=GENERATED' --runtime-configuration
'ServerProcesses=[{LaunchPath=/local/game/MainScript.js,Parameters=+map
Winter444,ConcurrentExecutions=5}]'

```

#### Output:

```

{
  "FleetAttributes": {
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "Status": "NEW",
    "CreationTime": 1569310745.212,
    "InstanceType": "c4.large",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    },
    "Name": "MegaFrogRace.NA.realtime",
    "ScriptId": "script-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetType": "SPOT",
    "MetricGroups": [
      "default"
    ],
  },
}

```

```
    "Description": "Mega Frog Race Realtime fleet",
    "OperatingSystem": "AMAZON_LINUX"
  }
}
```

- Untuk detail API, lihat [CreateFleet](#) di Referensi AWS CLI Perintah.

## create-game-session-queue

Contoh kode berikut menunjukkan cara menggunakan `create-game-session-queue`.

### AWS CLI

Contoh1: Untuk mengatur antrian sesi permainan yang diurutkan

`create-game-session-queue` Contoh berikut membuat antrian sesi permainan baru dengan tujuan di dua wilayah. Ini juga mengkonfigurasi antrian sehingga sesi permainan meminta waktu habis setelah menunggu 10 menit untuk penempatan. Karena tidak ada kebijakan latensi yang ditentukan, GameLift upaya untuk menempatkan semua sesi permainan dengan tujuan pertama yang terdaftar.

```
aws gamelift create-game-session-queue \
  --name MegaFrogRaceServer-NA \
  --destinations file://destinations.json \
  --timeout-in-seconds 600
```

Isi dari `destinations.json`:

```
{
  "Destinations": [
    {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" },
    {"DestinationArn": "arn:aws:gamelift:us-west-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222" }
  ]
}
```

Output:

```
{
```

```

"GameSessionQueues": [
  {
    "Name": "MegaFrogRaceServer-NA",
    "GameSessionQueueArn": "arn:aws:gamelift:us-west-2:123456789012:gamesessionqueue/MegaFrogRaceServer-NA",
    "TimeoutInSeconds": 600,
    "Destinations": [
      {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"},
      {"DestinationArn": "arn:aws:gamelift:us-west-1::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"}
    ]
  }
]
}

```

Contoh2: Untuk mengatur antrean sesi game dengan kebijakan latensi pemain

`create-game-session-queue` Contoh berikut membuat antrean sesi permainan baru dengan kebijakan latensi dua pemain. Kebijakan pertama menetapkan batas latensi 100 ms yang diberlakukan selama menit pertama upaya penempatan sesi game. Kebijakan kedua menaikkan batas latensi menjadi 200 ms hingga waktu permintaan penempatan habis pada 3 menit.

```

aws gamelift create-game-session-queue \
  --name MegaFrogRaceServer-NA \
  --destinations file://destinations.json \
  --player-latency-policies file://latency-policies.json \
  --timeout-in-seconds 180

```

Isi dari `destinations.json`:

```

{
  "Destinations": [
    { "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" },
    { "DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222" }
  ]
}

```

Isi dari `latency-policies.json`:

```
{
  "PlayerLatencyPolicies": [
    {"MaximumIndividualPlayerLatencyMilliseconds": 200},
    {"MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds":
60}
  ]
}
```

Output:

```
{
  "GameSessionQueue": {
    "Name": "MegaFrogRaceServer-NA",
    "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:111122223333:gamesessionqueue/MegaFrogRaceServer-NA",
    "TimeoutInSeconds": 600,
    "PlayerLatencyPolicies": [
      {
        "MaximumIndividualPlayerLatencyMilliseconds": 100,
        "PolicyDurationSeconds": 60
      },
      {
        "MaximumIndividualPlayerLatencyMilliseconds": 200
      }
    ]
    "Destinations": [
      {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"},
      {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"}
    ],
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Antrian](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [CreateGameSessionQueue](#) di Referensi AWS CLI Perintah.

## delete-build

Contoh kode berikut menunjukkan cara menggunakan `delete-build`.

## AWS CLI

Untuk menghapus build game kustom

`delete-build` Contoh berikut menghapus build dari GameLift akun Amazon Anda. Setelah build dihapus, Anda tidak dapat menggunakannya untuk membuat armada baru. Operasi ini tidak dapat dibatalkan.

```
aws gamelift delete-build \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBuild](#) di Referensi AWS CLI Perintah.

## **delete-fleet**

Contoh kode berikut menunjukkan cara menggunakan `delete-fleet`.

## AWS CLI

Untuk menghapus armada yang tidak lagi digunakan

`delete-fleet` Contoh berikut menghapus armada yang telah diperkecil menjadi nol instance. Jika kapasitas armada lebih besar dari nol, permintaan gagal dengan kesalahan HTTP 400.

```
aws gamelift delete-fleet \  
  --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [GameLift Mengelola Armada](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DeleteFleet](#) di Referensi AWS CLI Perintah.

## **delete-game-session-queue**

Contoh kode berikut menunjukkan cara menggunakan `delete-game-session-queue`.

## AWS CLI

Untuk menghapus antrean sesi permainan

`delete-game-session-queue` Contoh berikut menghapus antrian sesi permainan tertentu.

```
aws gamelift delete-game-session-queue \  
  --name MegaFrogRace-NA
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteGameSessionQueue](#) di Referensi AWS CLI Perintah.

## describe-build

Contoh kode berikut menunjukkan cara menggunakan `describe-build`.

### AWS CLI

Untuk mendapatkan informasi tentang pembuatan game khusus

`describe-build` Contoh berikut mengambil properti untuk sumber daya build server game.

```
aws gamelift describe-build \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "Build": {  
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "CreationTime": 1496708916.18,  
    "Name": "My_Game_Server_Build_One",  
    "OperatingSystem": "AMAZON_LINUX",  
    "SizeOnDisk": 1304924,  
    "Status": "READY",  
    "Version": "12345.678"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengunggah Pembuatan Server Kustom ke GameLift](#) dalam Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeBuild](#) di Referensi AWS CLI Perintah.

## describe-ec2-instance-limits

Contoh kode berikut menunjukkan cara menggunakan `describe-ec2-instance-limits`.

### AWS CLI

Untuk mengambil batas layanan untuk jenis instans EC2

`describe-ec2-instance-limits` Contoh berikut menampilkan instans maksimum yang diizinkan dan instans saat ini yang digunakan untuk jenis instans EC2 yang ditentukan di Wilayah saat ini. Hasilnya menunjukkan bahwa hanya lima dari dua puluh contoh yang diizinkan yang digunakan.

```
aws gamelift describe-ec2-instance-limits \  
  --ec2-instance-type m5.large
```

Output:

```
{  
  "EC2InstanceLimits": [  
    {  
      "EC2InstanceType": "m5.large",  
      "CurrentInstances": 5,  
      "InstanceLimit": 20  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Memilih Sumber Daya Komputasi](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeEc2 InstanceLimits](#) di Referensi AWS CLI Perintah.

## describe-fleet-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-attributes`.

### AWS CLI

Contoh1: Untuk melihat atribut untuk daftar armada

`describe-fleet-attributes` Contoh berikut mengambil atribut armada untuk dua armada tertentu. Seperti yang ditunjukkan, armada yang diminta disebar dengan build yang sama, satu untuk instans On-Demand dan satu untuk instance Spot, dengan beberapa perbedaan konfigurasi kecil.

```
aws gamelift describe-fleet-attributes \
  --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
  "FleetAttributes": [
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
      "FleetType": "ON_DEMAND",
      "InstanceType": "c4.large",
      "Description": "On-demand hosts for v2 North America",
      "Name": "MegaFrogRaceServer.NA.v2-od",
      "CreationTime": 1568836191.995,
      "Status": "ACTIVE",
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
      "ServerLaunchPath": "C:\\\\game\\MegaFrogRace_Server.exe",
      "ServerLaunchParameters": "+gamelift_start_server",
      "NewGameSessionProtectionPolicy": "NoProtection",
      "OperatingSystem": "WINDOWS_2012",
      "MetricGroups": [
        "default"
      ],
      "CertificateConfiguration": {
        "CertificateType": "DISABLED"
      }
    },
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
      "FleetType": "SPOT",
```



```

    "InstanceType": "c4.large",
    "Description": "On-demand hosts for v2 North America",
    "Name": "MegaFrogRaceServer.NA.v2-spot",
    "CreationTime": 1568838275.379,
    "Status": "ACTIVATING",
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
    "ServerLaunchPath": "C:\\\\game\\\\MegaFrogRace_Server.exe",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "WINDOWS_2012",
    "MetricGroups": [
        "default"
    ],
    "CertificateConfiguration": {
        "CertificateType": "GENERATED"
    }
}
]
}

```

Contoh2: Untuk meminta atribut untuk semua armada

Berikut ini `describe-fleet-attributes` mengembalikan atribut armada untuk semua armada dengan status apa pun. Contoh ini menggambarkan penggunaan parameter `pagination` untuk mengembalikan satu armada pada satu waktu.

```

aws gamelift describe-fleet-attributes \
  --limit 1

```

Output:

```

{
  "FleetAttributes": [
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
      "FleetType": "SPOT",
      "InstanceType": "c4.large",
      "Description": "On-demand hosts for v2 North America",
      "Name": "MegaFrogRaceServer.NA.v2-spot",
      "CreationTime": 1568838275.379,

```

```

        "Status": "ACTIVATING",
        "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
        "ServerLaunchPath": "C:\\game\\MegaFrogRace_Server.exe",
        "NewGameSessionProtectionPolicy": "NoProtection",
        "OperatingSystem": "WINDOWS_2012",
        "MetricGroups": [
            "default"
        ],
        "CertificateConfiguration": {
            "CertificateType": "GENERATED"
        }
    }
],
    "NextToken":
    "eyJhd3NBWY2NvdW50SWQI0nsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjpw7InMiOiJidWlsZC01NWYxZTZmMS1"
}

```

Outputnya mencakup NextToken nilai yang dapat Anda gunakan saat Anda memanggil perintah untuk kedua kalinya. Berikan nilai ke `--next-token` parameter untuk menentukan di mana untuk mengambil output. Perintah berikut mengembalikan hasil kedua dalam output.

```

aws gamelift describe-fleet-attributes \
  --limit 1 \
  --next-token
eyJhd3NBWY2NvdW50SWQI0nsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjpw7InMiOiJidWlsZC01NWYxZTZmMS1

```

Ulangi sampai respon tidak menyertakan NextToken nilai.

Untuk informasi selengkapnya, lihat [Menyiapkan GameLift Armada](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeFleetAttributes](#) di Referensi AWS CLI Perintah.

## describe-fleet-capacity

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-capacity`.

### AWS CLI

Untuk melihat status kapasitas untuk daftar armada

`describe-fleet-capacity` Contoh berikut mengambil kapasitas saat ini untuk dua armada tertentu.

```
aws gamelift describe-fleet-capacity \
  --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
  EXAMPLE11111 fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
  "FleetCapacity": [
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "InstanceType": "c5.large",
      "InstanceCounts": {
        "DESIRED": 10,
        "MINIMUM": 1,
        "MAXIMUM": 20,
        "PENDING": 0,
        "ACTIVE": 10,
        "IDLE": 3,
        "TERMINATING": 0
      }
    },
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "InstanceType": "c5.large",
      "InstanceCounts": {
        "DESIRED": 13,
        "MINIMUM": 1,
        "MAXIMUM": 20,
        "PENDING": 0,
        "ACTIVE": 15,
        "IDLE": 2,
        "TERMINATING": 2
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [GameLift Metrik untuk Armada](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeFleetCapacity](#) di Referensi AWS CLI Perintah.

## describe-fleet-events

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-events`.

### AWS CLI

Untuk meminta acara untuk rentang waktu tertentu

`describe-fleet-events` Contoh berikut menampilkan detail semua peristiwa terkait armada yang terjadi selama rentang waktu yang ditentukan.

```
aws gamelift describe-fleet-events \
  --fleet-id arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 \
  --start-time 1579647600 \
  --end-time 1579649400 \
  --limit 5
```

### Output:

```
{
  "Events": [
    {
      "EventId": "a37b6892-5d07-4d3b-8b47-80244ecf66b9",
      "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "EventCode": "FLEET_STATE_ACTIVE",
      "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to ACTIVE",
      "EventTime": 1579649342.191
    },
    {
      "EventId": "67da4ec9-92a3-4d95-886a-5d6772c24063",
      "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "EventCode": "FLEET_STATE_ACTIVATING",
      "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to ACTIVATING",
      "EventTime": 1579649321.427
    },
  ],
}
```

```

    {
      "EventId": "23813a46-a9e6-4a53-8847-f12e6a8381ac",
      "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "EventCode": "FLEET_STATE_BUILDING",
      "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to BUILDING",
      "EventTime": 1579649321.243
    },
    {
      "EventId": "3bf217d0-1d44-42f9-9202-433ed475d2e8",
      "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "EventCode": "FLEET_STATE_VALIDATING",
      "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to VALIDATING",
      "EventTime": 1579649197.449
    },
    {
      "EventId": "2ecd0130-5986-44eb-99a7-62df27741084",
      "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "EventCode": "FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND",
      "Message": "Failed to find a valid path",
      "EventTime": 1569319075.839,
      "PreSignedLogUrl": "https://gamelift-event-logs-prod-
us-west-2.s3.us-west-2.amazonaws.com/logs/fleet-83422059-8329-42a2-
a4d6-c4444386a6f8/events/2ecd0130-5986-44eb-99a7-62df27741084/
FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND.txt?X-Amz-Security-
Token=IQoJb3JpZ2luX2VjEB8aCXVzLXdlc3Q0MiJHMEUCIHV5K%2FLPx8h310D
%2FAvx0%2FZxsDy5XA3cJ0wPdu3T0eBa%2FAiEA1yovokcZYy%2FV4CWW6126aFyiSH0
%2Bxz%2FBMAhEHYHMQNcqkQMImp%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F
%2FARAAGgw3NDEwNjE10TIXNzEiDI8rsZtzLzlwEDQhXSrlAtl5Ae
%2Fgo6FCIzqxPbXfb0nSvFYqeDlriZarEpKqKrUt8mXQv9iqHResqCph9AKo491wgSYTT2QoSxnrD7%2FUgv
%2BZm2pVuczvUkUA0fcx6s0GxpjIAzdIE%2F5P%2FB7B9M%2BVZ
%2F9KF82hbJi0HTE6Y7BjKsEgFCvk4UXILhfjtan9iQl8%2F21ZTurAcJbm7Y5tuLF9SWSK3%2BEa7VX0cCK4D401sMj
%2FIaXoHkNvg0RVTa0hIqdvpdQlSBNdqTXbjHTu6fETE9Y9Ky%2BiJK5KiUG
%2F59GjCpDcvS1FqKeLUEmKT7wysGmvjMc2n%2Fr
%2F9VxQfte7w9srXwLLAQuwhiXAAyI5ICM5JvzjzQwTqD4CHTVKUUDwL
%2BRZzbuuqkJObZml02CkRGp%2B74RTAzLbWptVqZTIIfzctiCTmWxb
%2FmKyELRySvLrwNj%2BGJ7%2BCrN0RC%2FjlgfLYIZyAqjPgAu5HjgX
%2BM7jCo9M7wBTInAXK0FQuf9dvA84SuwX0JFp17LYGjrHMKv0qC3GfbTMrZ6kzeNV9awKCpXB2Gnx9z2KvI1JdqirWV
%2F9C6%2B4jIZPME3jXmZcEHqqw5uvAVF7aeIavtUZU8pXPDIWT0YE4p3Kriy2AA7ziCRKtVfjv839InyLk8LUjsioWk
%2BYUq8%2FDTL1Lxqj1S%2Fi04TI0Wo7ilAo%2FKKWWF4guuNDexj8E00ynSp1yImB
%2BZf2Fua3044W4eEXAMPLE33333&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170621T231808Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-

```

```
Credential=AKIAIOSFODNN7EXAMPLE%2F20170621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
    }
  ],
  "NextToken":
    "eyJhd3NBWY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS"
}
```

Untuk informasi selengkapnya, lihat [Mendebug Masalah GameLift Armada](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeFleetEvents](#) di Referensi AWS CLI Perintah.

## describe-fleet-port-settings

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-port-settings`.

### AWS CLI

Untuk melihat izin koneksi masuk untuk armada

`describe-fleet-port-settings` Contoh berikut mengambil pengaturan koneksi untuk armada tertentu.

```
aws gamelift describe-fleet-port-settings \
  --fleet-id arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
  EXAMPLE11111
```

Output:

```
{
  "InboundPermissions": [
    {
      "FromPort": 33400,
      "ToPort": 33500,
      "IpRange": "0.0.0.0/0",
      "Protocol": "UDP"
    },
    {
      "FromPort": 1900,
      "ToPort": 2000,
```

```
        "IpRange": "0.0.0.0/0",
        "Protocol": "TCP"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan GameLift Armada](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeFleetPortSettings](#) di Referensi AWS CLI Perintah.

## describe-fleet-utilization

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet-utilization`.

### AWS CLI

Contoh1: Untuk melihat data penggunaan untuk daftar armada

`describe-fleet-utilization` Contoh berikut mengambil informasi penggunaan saat ini untuk satu armada tertentu.

```
aws gamelift describe-fleet-utilization \
  --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
  EXAMPLE11111
```

Output:

```
{
  "FleetUtilization": [
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ActiveServerProcessCount": 100,
      "ActiveGameSessionCount": 62,
      "CurrentPlayerSessionCount": 329,
      "MaximumPlayerSessionCount": 1000
    }
  ]
}
```

Contoh2: Untuk meminta data penggunaan untuk semua armada

Berikut ini `describe-fleet-utilization` mengembalikan data penggunaan armada untuk semua armada dengan status apa pun. Contoh ini menggunakan parameter `pagination` untuk mengembalikan data untuk dua armada sekaligus.

```
aws gamelift describe-fleet-utilization \
  --limit 2
```

Output:

```
{
  "FleetUtilization": [
    {
      "FleetId": "fleet-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
      "ActiveServerProcessCount": 100,
      "ActiveGameSessionCount": 13,
      "CurrentPlayerSessionCount": 98,
      "MaximumPlayerSessionCount": 1000
    },
    {
      "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
      "ActiveServerProcessCount": 100,
      "ActiveGameSessionCount": 62,
      "CurrentPlayerSessionCount": 329,
      "MaximumPlayerSessionCount": 1000
    }
  ],
  "NextToken":
  "eyJhd3NBWY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS1"
}
```

Panggil perintah untuk kedua kalinya, `NextToken` berikan nilai sebagai argumen ke `--next-token` parameter untuk melihat dua hasil berikutnya.

```
aws gamelift describe-fleet-utilization \
  --limit 2 \
  --next-token
eyJhd3NBWY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS1
```

Ulangi sampai respon tidak lagi menyertakan `NextToken` nilai dalam output.

Untuk informasi selengkapnya, lihat [GameLift Metrik untuk Armada](#) di Panduan GameLift Pengembang Amazon.



- Untuk detail API, lihat [DescribeFleetUtilization](#) di Referensi AWS CLI Perintah.

## describe-game-session-queues

Contoh kode berikut menunjukkan cara menggunakan `describe-game-session-queues`.

### AWS CLI

Untuk melihat antrian sesi permainan

`describe-game-session-queues` Contoh berikut mengambil properti untuk dua antrian tertentu.

```
aws gamelift describe-game-session-queues \
  --names MegaFrogRace-NA MegaFrogRace-EU
```

Output:

```
{
  "GameSessionQueues": [{
    "Destinations": [{
      "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    {
      "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
    }
  ],
  "Name": "MegaFrogRace-NA",
  "TimeoutInSeconds": 600,
  "GameSessionQueueArn": "arn:aws:gamelift:us-west-2::gamesessionqueue/
MegaFrogRace-NA",
  "PlayerLatencyPolicies": [{
    "MaximumIndividualPlayerLatencyMilliseconds": 200
  },
  {
    "MaximumIndividualPlayerLatencyMilliseconds": 100,
    "PolicyDurationSeconds": 60
  }
  ],
  "FilterConfiguration": {
```

```

        "AllowedLocations": ["us-west-2", "ap-south-1", "us-east-1"]
    },
    "PriorityConfiguration": {
        "PriorityOrder": ["LOCATION", "FLEET_TYPE", "DESTINATION"],
        "LocationOrder": ["us-west-2", "ap-south-1", "us-east-1"]
    }
},
{
    "Destinations": [{
        "DestinationArn": "arn:aws:gamelift:eu-west-3::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
    }],
    "Name": "MegaFrogRace-EU",
    "TimeoutInSeconds": 600,
    "GameSessionQueueArn": "arn:aws:gamelift:us-west-2::gamesessionqueue/
MegaFrogRace-EU"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menggunakan Antrian Multi-Wilayah di Panduan Pengembang Amazon GameLift](#) .

- Untuk detail API, lihat [DescribeGameSessionQueues](#) di Referensi AWS CLI Perintah.

## describe-runtime-configuration

Contoh kode berikut menunjukkan cara menggunakan `describe-runtime-configuration`.

### AWS CLI

Untuk meminta konfigurasi runtime untuk armada

`describe-runtime-configuration` Contoh berikut mengambil rincian tentang konfigurasi runtime saat ini untuk armada tertentu.

```
aws gamelift describe-runtime-configuration \
  --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
```

```

    "RuntimeConfiguration": {
      "ServerProcesses": [
        {
          "LaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
          "Parameters": "+gamelift_start_server",
          "ConcurrentExecutions": 3
        },
        {
          "LaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
          "Parameters": "+gamelift_start_server +debug",
          "ConcurrentExecutions": 1
        }
      ],
      "MaxConcurrentGameSessionActivations": 2147483647,
      "GameSessionActivationTimeoutSeconds": 300
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Menjalankan Beberapa Proses pada Armada](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [DescribeRuntimeConfiguration](#) di Referensi AWS CLI Perintah.

## list-builds

Contoh kode berikut menunjukkan cara menggunakan `list-builds`.

### AWS CLI

Contoh1: Untuk mendapatkan daftar build game khusus

`list-builds` Contoh berikut mengambil properti untuk semua build server game di Region saat ini. Permintaan sampel menggambarkan cara menggunakan parameter pagination, `Limit` dan `NextToken`, untuk mengambil hasil dalam set sekuensial. Perintah pertama mengambil dua build pertama. Karena ada lebih dari dua yang tersedia, responsnya mencakup a `NextToken` untuk menunjukkan bahwa lebih banyak hasil yang tersedia.

```

aws gamelift list-builds \
  --limit 2

```

**Output:**

```
{
  "Builds": [
    {
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "CreationTime": 1495664528.723,
      "Name": "My_Game_Server_Build_One",
      "OperatingSystem": "WINDOWS_2012",
      "SizeOnDisk": 8567781,
      "Status": "READY",
      "Version": "12345.678"
    },
    {
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "CreationTime": 1495528748.555,
      "Name": "My_Game_Server_Build_Two",
      "OperatingSystem": "AMAZON_LINUX_2",
      "SizeOnDisk": 8567781,
      "Status": "FAILED",
      "Version": "23456.789"
    }
  ],
  "NextToken":
  "eyJhd3NBZjY2NvdW50SWQlbnRlcyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS1"
}
```

Anda kemudian dapat memanggil perintah lagi dengan `--next-token` parameter sebagai berikut untuk melihat dua build berikutnya.

```
aws gamelift list-builds \
  --limit 2
  --next-token
  eyJhd3NBZjY2NvdW50SWQlbnRlcyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS1
```

Ulangi sampai respon tidak menyertakan `NextToken` nilai.

**Contoh2:** Untuk mendapatkan daftar build game khusus dalam status kegagalan

`list-builds` Contoh berikut mengambil properti untuk semua build server game di wilayah saat ini yang saat ini memiliki status GAGAL.

```
aws gamelift list-builds \  
  --status FAILED
```

Output:

```
{  
  "Builds": [  
    {  
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "CreationTime": 1495528748.555,  
      "Name": "My_Game_Server_Build_Two",  
      "OperatingSystem": "AMAZON_LINUX_2",  
      "SizeOnDisk": 8567781,  
      "Status": "FAILED",  
      "Version": "23456.789"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListBuilds](#) di Referensi AWS CLI Perintah.

## list-fleets

Contoh kode berikut menunjukkan cara menggunakan `list-fleets`.

### AWS CLI

Contoh1: Untuk mendapatkan daftar semua armada di suatu Wilayah

`list-fleets` Contoh berikut menampilkan ID armada dari semua armada di Wilayah saat ini.

Contoh ini menggunakan parameter `pagination` untuk mengambil dua ID armada sekaligus.

Respons mencakup `next-token` atribut, yang menunjukkan bahwa ada lebih banyak hasil untuk diambil.

```
aws gamelift list-fleets \  
  --limit 2
```

**Output:**

```
{
  "FleetIds": [
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
  ],
  "NextToken":
  "eyJhd3NBW52NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZEIkJp7InMiOiJidWlsZC01NWYxZTZmMS"
}
```

Anda dapat meneruskan NextToken nilai dari respons sebelumnya di perintah berikutnya, seperti yang ditunjukkan di sini untuk mendapatkan dua hasil berikutnya.

```
aws gamelift list-fleets \
  --limit 2 \
  --next-token
eyJhd3NBW52NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZEIkJp7InMiOiJidWlsZC00NDRLZjQxZS1
```

Contoh2: Untuk mendapatkan daftar semua armada di Wilayah dengan build atau skrip tertentu

`list-builds` Contoh berikut mengambil ID armada yang digunakan dengan build game yang ditentukan. Jika Anda bekerja dengan Server Realtime, Anda dapat memberikan ID skrip sebagai pengganti ID build. Karena contoh ini tidak menentukan parameter batas, hasilnya dapat mencakup hingga 16 ID armada.

```
aws gamelift list-fleets \
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

**Output:**

```
{
  "FleetIds": [
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE44444"
  ]
}
```

- Untuk detail API, lihat [ListFleets](#) di Referensi AWS CLI Perintah.

## request-upload-credentials

Contoh kode berikut menunjukkan cara menggunakan `request-upload-credentials`.

### AWS CLI

Untuk menyegarkan kredensial akses untuk mengunggah build

`create-build` Contoh berikut memperoleh kredensial akses baru yang valid untuk mengunggah file GameLift build ke lokasi Amazon S3. Kredensial memiliki rentang hidup yang terbatas. Anda mendapatkan ID build dari respons `CreateBuild` permintaan asli.

```
aws gamelift request-upload-credentials \  
--build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "StorageLocation": {  
    "Bucket": "gamelift-builds-us-west-2",  
    "Key": "123456789012/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
  },  
  "UploadCredentials": {  
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",  
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",  
    "SessionToken": "AgoGb3JpZ21uENZ...EXAMPLETOKEN=="  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengunggah Pembuatan Server Kustom ke GameLift](#) dalam Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [RequestUploadCredentials](#) di Referensi AWS CLI Perintah.

## start-fleet-actions

Contoh kode berikut menunjukkan cara menggunakan `start-fleet-actions`.

### AWS CLI

Untuk memulai ulang aktivitas penskalaan otomatis armada

`start-fleet-actions` Contoh berikut melanjutkan penggunaan semua kebijakan penskalaan yang didefinisikan untuk armada tertentu tetapi dihentikan dengan memanggil `stop-fleet-actions`. Setelah memulai, kebijakan penskalaan segera mulai melacak metrik masing-masing.

```
aws gamelift start-fleet-actions \  
  --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --actions AUTO_SCALING
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [StartFleetActions](#) di Referensi AWS CLI Perintah.

## **stop-fleet-actions**

Contoh kode berikut menunjukkan cara menggunakan `stop-fleet-actions`.

### AWS CLI

Untuk menghentikan aktivitas penskalaan otomatis armada

`stop-fleet-actions` Contoh berikut menghentikan penggunaan semua kebijakan penskalaan yang ditentukan untuk armada yang ditentukan. Setelah kebijakan ditangguhkan, kapasitas armada tetap dalam jumlah instans aktif yang sama kecuali Anda menyesuainya secara manual.

```
aws gamelift start-fleet-actions \  
  --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --actions AUTO_SCALING
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [StopFleetActions](#) di Referensi AWS CLI Perintah.

## **update-build**

Contoh kode berikut menunjukkan cara menggunakan `update-build`.

### AWS CLI

Untuk memperbarui build game kustom



`update-build` Contoh berikut mengubah nama dan informasi versi yang dikaitkan dengan sumber daya build tertentu. Objek build yang dikembalikan memverifikasi bahwa perubahan berhasil dilakukan.

```
aws gamelift update-build \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --name MegaFrogRaceServer.NA.east \  
  --build-version 12345.east
```

Output:

```
{  
  "Build": {  
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "CreationTime": 1496708916.18,  
    "Name": "MegaFrogRaceServer.NA.east",  
    "OperatingSystem": "AMAZON_LINUX_2",  
    "SizeOnDisk": 1304924,  
    "Status": "READY",  
    "Version": "12345.east"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui File Build Anda](#) di Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [UpdateBuild](#) di Referensi AWS CLI Perintah.

## **update-game-session-queue**

Contoh kode berikut menunjukkan cara menggunakan `update-game-session-queue`.

### AWS CLI

Untuk memperbarui konfigurasi antrian sesi permainan

`update-game-session-queue` Contoh berikut menambahkan tujuan baru dan memperbarui kebijakan latensi pemain untuk antrean sesi game yang ada.

```
aws gamelift update-game-session-queue \  
  --game-session-queue-name game-session-queue
```

```
--name MegaFrogRace-NA \  
--destinations file://destinations.json \  
--player-latency-policies file://latency-policies.json
```

Isi dari destinations.json:

```
{  
  "Destinations": [  
    {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/  
fleet-1a2b3c4d-5e6f-7a8b-9c0d-1e2f3a4b5c6d"},  
    {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/  
fleet-5c6d3c4d-5e6f-7a8b-9c0d-1e2f3a4b5a2b"},  
    {"DestinationArn": "arn:aws:gamelift:us-east-1::alias/  
alias-11aa22bb-3c4d-5e6f-000a-1111aaaa22bb"}  
  ]  
}
```

Isi dari latency-policies.json:

```
{  
  "PlayerLatencyPolicies": [  
    {"MaximumIndividualPlayerLatencyMilliseconds": 200},  
    {"MaximumIndividualPlayerLatencyMilliseconds": 150, "PolicyDurationSeconds":  
120},  
    {"MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds":  
120}  
  ]  
}
```

Output:

```
{  
  "GameSessionQueue": {  
    "Destinations": [  
      {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/  
fleet-1a2b3c4d-5e6f-7a8b-9c0d-1e2f3a4b5c6d"},  
      {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/  
fleet-5c6d3c4d-5e6f-7a8b-9c0d-1e2f3a4b5a2b"},  
      {"DestinationArn": "arn:aws:gamelift:us-east-1::alias/  
alias-11aa22bb-3c4d-5e6f-000a-1111aaaa22bb"}  
    ],  
  }  
}
```

```

    "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:111122223333:gamesessionqueue/MegaFrogRace-NA",
    "Name": "MegaFrogRace-NA",
    "TimeoutInSeconds": 600,
    "PlayerLatencyPolicies": [
      {"MaximumIndividualPlayerLatencyMilliseconds": 200},
      {"MaximumIndividualPlayerLatencyMilliseconds": 150,
"PolicyDurationSeconds": 120},
      {"MaximumIndividualPlayerLatencyMilliseconds": 100,
"PolicyDurationSeconds": 120}
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Menggunakan Antrian Multi-Wilayah di Panduan Pengembang Amazon GameLift](#).

- Untuk detail API, lihat [UpdateGameSessionQueue](#) di Referensi AWS CLI Perintah.

## upload-build

Contoh kode berikut menunjukkan cara menggunakan `upload-build`.

### AWS CLI

Contoh1: Untuk mengunggah build server game Linux

`upload-build` Contoh berikut mengunggah file build server game Linux dari direktori file ke GameLift layanan dan membuat sumber daya build.

```

aws gamelift upload-build \
  --name MegaFrogRaceServer.NA \
  --build-version 2.0.1 \
  --build-root ~/MegaFrogRace_Server/release-na \
  --operating-system AMAZON_LINUX_2
  --server-sdk-version 4.0.2

```

### Output:

```

Uploading ~/MegaFrogRace_Server/release-na: 16.0 KiB / 74.6 KiB (21.45%)
Uploading ~/MegaFrogRace_Server/release-na: 32.0 KiB / 74.6 KiB (42.89%)
Uploading ~/MegaFrogRace_Server/release-na: 48.0 KiB / 74.6 KiB (64.34%)
Uploading ~/MegaFrogRace_Server/release-na: 64.0 KiB / 74.6 KiB (85.79%)

```

```
Uploading ~/MegaFrogRace_Server/release-na: 74.6 KiB / 74.6 KiB (100.00%)
Successfully uploaded ~/MegaFrogRace_Server/release-na to AWS GameLift
Build ID: build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Contoh2: Untuk mengunggah build server game Windows

upload-build Contoh berikut mengunggah file build server game Windows dari direktori ke GameLift layanan dan membuat catatan build.

```
aws gamelift upload-build \
  --name MegaFrogRaceServer.NA \
  --build-version 2.0.1 \
  --build-root C:\MegaFrogRace_Server\release-na \
  --operating-system WINDOWS_2012
  --server-sdk-version 4.0.2
```

Output:

```
Uploading C:\MegaFrogRace_Server\release-na: 16.0 KiB / 74.6 KiB (21.45%)
Uploading C:\MegaFrogRace_Server\release-na: 32.0 KiB / 74.6 KiB (42.89%)
Uploading C:\MegaFrogRace_Server\release-na: 48.0 KiB / 74.6 KiB (64.34%)
Uploading C:\MegaFrogRace_Server\release-na: 64.0 KiB / 74.6 KiB (85.79%)
Uploading C:\MegaFrogRace_Server\release-na: 74.6 KiB / 74.6 KiB (100.00%)
Successfully uploaded C:\MegaFrogRace_Server\release-na to AWS GameLift
Build ID: build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Untuk informasi selengkapnya, lihat [Mengunggah Pembuatan Server Kustom ke GameLift](#) dalam Panduan GameLift Pengembang Amazon.

- Untuk detail API, lihat [UploadBuild](#) di Referensi AWS CLI Perintah.

## Contoh Global Accelerator menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Global Accelerator.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **add-custom-routing-endpoints**

Contoh kode berikut menunjukkan cara menggunakan `add-custom-routing-endpoints`.

## AWS CLI

Untuk menambahkan titik akhir subnet VPC ke grup titik akhir untuk akselerator perutean kustom

`add-custom-routing-endpoints` Contoh berikut menambahkan titik akhir subnet VPC ke grup titik akhir untuk akselerator perutean kustom.

```
aws globalaccelerator add-custom-routing-endpoints \
  --endpoint-group-arn
  arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
  abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd \
  --endpoint-configurations "EndpointId=subnet-1234567890abcdef0"
```

## Output:

```
{
  "EndpointDescriptions": [
    {
      "EndpointId": "subnet-1234567890abcdef0"
    }
  ],
  "EndpointGroupArn": "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
  abcd-1234-abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd"
}
```

Untuk informasi selengkapnya, lihat [titik akhir subnet VPC untuk akselerator perutean kustom di Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).AWS

- Untuk detail API, lihat [AddCustomRoutingEndpoints](#) di Referensi AWS CLI Perintah.

## advertise-byoip-cidr

Contoh kode berikut menunjukkan cara menggunakan `advertise-byoip-cidr`.

### AWS CLI

Untuk mengiklankan rentang alamat

`advertise-byoip-cidr` Contoh berikut meminta AWS untuk mengiklankan rentang alamat yang telah Anda sediakan untuk digunakan dengan sumber daya Anda. AWS

```
aws globalaccelerator advertise-byoip-cidr \  
  --cidr 198.51.100.0/24
```

Output:

```
{  
  "ByoipCidr": {  
    "Cidr": "198.51.100.0/24",  
    "State": "PENDING_ADVERTISING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membawa Alamat IP Anda Sendiri di AWS Global Accelerator di Panduan](#) Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [AdvertiseByoipCidr](#) di Referensi AWS CLI Perintah.

## allow-custom-routing-traffic

Contoh kode berikut menunjukkan cara menggunakan `allow-custom-routing-traffic`.

### AWS CLI

Untuk mengizinkan lalu lintas ke tujuan instans Amazon EC2 tertentu di subnet VPC untuk akselerator perutean khusus

`allow-custom-routing-traffic` Contoh berikut menentukan bahwa lalu lintas diizinkan ke alamat IP instans (tujuan) Amazon EC2 tertentu dan port untuk titik akhir subnet VPC dalam akselerator perutean khusus dapat menerima lalu lintas.

```
aws globalaccelerator allow-custom-routing-traffic \  
  \
```

```

--endpoint-group-arn
arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab88888example \
--endpoint-id subnet-abcd123example \
--destination-addresses "172.31.200.6" "172.31.200.7" \
--destination-ports 80 81

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [titik akhir subnet VPC untuk akselerator perutean kustom di Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).AWS

- Untuk detail API, lihat [AllowCustomRoutingTraffic](#) di Referensi AWS CLI Perintah.

## create-accelerator

Contoh kode berikut menunjukkan cara menggunakan `create-accelerator`.

### AWS CLI

Untuk membuat akselerator

`create-accelerator` Contoh berikut membuat akselerator dengan dua tag dengan dua alamat IP statis BYOIP. Anda harus menentukan US-West-2 (Oregon) Wilayah untuk membuat atau memperbarui akselerator.

```

aws globalaccelerator create-accelerator \
--name ExampleAccelerator \
--tags Key="Name",Value="Example Name" Key="Project",Value="Example Project" \
--ip-addresses 192.0.2.250 198.51.100.52

```

Output:

```

{
  "Accelerator": {
    "AcceleratorArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [

```

```

    {
      "IpAddresses": [
        "192.0.2.250",
        "198.51.100.52"
      ],
      "IpFamily": "IPv4"
    }
  ],
  "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
  "CreatedTime": 1542394847.0,
  "LastModifiedTime": 1542394847.0
}
}

```

Untuk informasi selengkapnya, lihat [Akselerator di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [CreateAccelerator](#) di Referensi AWS CLI Perintah.

## create-custom-routing-accelerator

Contoh kode berikut menunjukkan cara menggunakan `create-custom-routing-accelerator`.

### AWS CLI

Untuk membuat akselerator perutean kustom

`create-custom-routing-accelerator` Contoh berikut membuat akselerator routing kustom dengan tag `Name` dan `Project`

```

aws globalaccelerator create-custom-routing-accelerator \
  --name ExampleCustomRoutingAccelerator \
  --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project" \
  --ip-addresses 192.0.2.250 198.51.100.52

```

Output:

```

{
  "Accelerator": {
    "AcceleratorArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
      abcd-1234abcdefgh",
    "IpAddressType": "IPV4",

```



```

    "Name": "ExampleCustomRoutingAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
          "192.0.2.250",
          "198.51.100.52"
        ],
        "IpFamily": "IPv4"
      }
    ],
    "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
    "CreatedTime": 1542394847.0,
    "LastModifiedTime": 1542394847.0
  }
}

```

Untuk informasi selengkapnya, lihat [Akselerator perutean khusus di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [CreateCustomRoutingAccelerator](#) di Referensi AWS CLI Perintah.

## create-custom-routing-endpoint-group

Contoh kode berikut menunjukkan cara menggunakan `create-custom-routing-endpoint-group`.

### AWS CLI

Untuk membuat grup endpoint untuk akselerator perutean kustom

`create-custom-routing-endpoint-group` Contoh berikut membuat grup endpoint untuk akselerator routing kustom.

```

aws globalaccelerator create-custom-routing-endpoint-group \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --endpoint-group-region us-east-2 \
  --destination-configurations "FromPort=80,ToPort=81,Protocols=TCP,UDP"

```

Output:

```
{
  "EndpointGroup": {
    "EndpointGroupArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
      abcd-1234abcdefg/0123vxyz/endpoint-group/4321abcd",
    "EndpointGroupRegion": "us-east-2",
    "DestinationDescriptions": [
      {
        "FromPort": 80,
        "ToPort": 81,
        "Protocols": [
          "TCP",
          "UDP"
        ]
      }
    ],
    "EndpointDescriptions": []
  }
}
```

Untuk informasi selengkapnya, lihat [Grup titik akhir untuk akselerator perutean kustom di AWS Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [CreateCustomRoutingEndpointGroup](#) di Referensi AWS CLI Perintah.

## **create-custom-routing-listener**

Contoh kode berikut menunjukkan cara menggunakan `create-custom-routing-listener`.

### AWS CLI

Untuk membuat pendengar untuk akselerator perutean kustom

`create-custom-routing-listener` Contoh berikut membuat pendengar dengan rentang port dari 5000 hingga 10000 untuk akselerator perutean kustom.

```
aws globalaccelerator create-custom-routing-listener \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
  abcd-1234-abcd-1234abcdefg \
  --port-ranges FromPort=5000,ToPort=10000
```

Output:

```
{
  "Listener": {
    "PortRange": [
      "FromPort": 5000,
      "ToPort": 10000
    ],
    "ListenerArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz"
  }
}
```

Untuk informasi selengkapnya, lihat [Pendengar untuk akselerator perutean kustom di Global Accelerator di Panduan AWS Pengembang AWS Akselerator Global](#).

- Untuk detail API, lihat [CreateCustomRoutingListener](#) di Referensi AWS CLI Perintah.

## create-endpoint-group

Contoh kode berikut menunjukkan cara menggunakan `create-endpoint-group`.

### AWS CLI

Untuk membuat grup endpoint

`create-endpoint-group` Contoh berikut membuat grup endpoint dengan satu endpoint.

```
aws globalaccelerator create-endpoint-group \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --endpoint-group-region us-east-1 \
  --endpoint-configurations EndpointId=i-1234567890abcdef0,Weight=128
```

Output:

```
{
  "EndpointGroup": {
    "TrafficDialPercentage": 100.0,
    "EndpointDescriptions": [
      {
        "Weight": 128,
        "EndpointId": "i-1234567890abcdef0"
      }
    ]
  }
}
```

```

    }
  ],
  "EndpointGroupArn":
  "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
  "EndpointGroupRegion": "us-east-1"
}
}

```

Untuk informasi selengkapnya, lihat [Grup titik akhir di Akselerator AWS Global di Panduan Pengembang](#) Akselerator AWS Global.

- Untuk detail API, lihat [CreateEndpointGroup](#) di Referensi AWS CLI Perintah.

## create-listener

Contoh kode berikut menunjukkan cara menggunakan `create-listener`.

### AWS CLI

Untuk membuat pendengar

`create-listener` Contoh berikut membuat listener dengan dua port.

```

aws globalaccelerator create-listener \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --port-ranges FromPort=80,ToPort=80 FromPort=81,ToPort=81 \
  --protocol TCP

```

Output:

```

{
  "Listener": {
    "PortRanges": [
      {
        "ToPort": 80,
        "FromPort": 80
      },
      {
        "ToPort": 81,
        "FromPort": 81
      }
    ]
  }
}

```

```

    }
  ],
  "ClientAffinity": "NONE",
  "Protocol": "TCP",
  "ListenerArn":
  "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz"
}
}

```

Untuk informasi selengkapnya, lihat [Pendengar di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [CreateListener](#) di Referensi AWS CLI Perintah.

## deny-custom-routing-traffic

Contoh kode berikut menunjukkan cara menggunakan `deny-custom-routing-traffic`.

### AWS CLI

Untuk menentukan alamat tujuan yang tidak dapat menerima lalu lintas dalam akselerator perutean kustom

`deny-custom-routing-traffic` Contoh berikut menentukan alamat tujuan atau alamat di titik akhir subnet yang tidak dapat menerima lalu lintas untuk akselerator perutean kustom. Untuk menentukan lebih dari satu alamat tujuan, pisahkan alamat dengan spasi. Tidak ada jawaban untuk `deny-custom-routing-traffic` panggilan yang berhasil.

```

aws globalaccelerator deny-custom-routing-traffic \
  --endpoint-group-arn
  "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab8888example" \
  --endpoint-id "subnet-abcd123example" \
  --destination-addresses "198.51.100.52"

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [titik akhir subnet VPC untuk akselerator perutean kustom di Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).AWS

- Untuk detail API, lihat [DenyCustomRoutingTraffic](#) di Referensi AWS CLI Perintah.

## deprovision-byoip-cidr

Contoh kode berikut menunjukkan cara menggunakan `deprovision-byoip-cidr`.

### AWS CLI

Untuk menghentikan penyediaan rentang alamat

`deprovision-byoip-cidr` Contoh berikut merilis rentang alamat tertentu yang Anda sediakan untuk digunakan dengan sumber daya Anda AWS .

```
aws globalaccelerator deprovision-byoip-cidr \  
  --cidr "198.51.100.0/24"
```

Output:

```
{  
  "ByoipCidr": {  
    "Cidr": "198.51.100.0/24",  
    "State": "PENDING_DEPROVISIONING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membawa alamat IP Anda sendiri di AWS Global Accelerator di Panduan](#) Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [DeprovisionByoipCidr](#) di Referensi AWS CLI Perintah.

## describe-accelerator-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-accelerator-attributes`.

### AWS CLI

Untuk menggambarkan atribut akselerator

`describe-accelerator-attributes` Contoh berikut mengambil rincian atribut untuk akselerator.

```
aws globalaccelerator describe-accelerator-attributes \  
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefg
```

**Output:**

```
{
  "AcceleratorAttributes": {
    "FlowLogsEnabled": true
    "FlowLogsS3Bucket": flowlogs-abc
    "FlowLogsS3Prefix": bucketprefix-abc
  }
}
```

Untuk informasi selengkapnya, lihat [Akselerator di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [DescribeAcceleratorAttributes](#) di Referensi AWS CLI Perintah.

**describe-accelerator**

Contoh kode berikut menunjukkan cara menggunakan `describe-accelerator`.

**AWS CLI**

Untuk menggambarkan akselerator

`describe-accelerator` Contoh berikut mengambil rincian tentang akselerator yang ditentukan.

```
aws globalaccelerator describe-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

**Output:**

```
{
  "Accelerator": {
    "AcceleratorArn":
    "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
```

```

        "IpAddresses": [
            "192.0.2.250",
            "198.51.100.52"
        ],
        "IpFamily": "IPv4"
    }
],
"DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
"CreatedTime": 1542394847,
"LastModifiedTime": 1542395013
}
}

```

Untuk informasi selengkapnya, lihat [Akselerator di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [DescribeAccelerator](#) di Referensi AWS CLI Perintah.

## describe-custom-routing-accelerator-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-routing-accelerator-attributes`.

### AWS CLI

Untuk menggambarkan atribut akselerator perutean kustom

`describe-custom-routing-accelerator-attributes` Contoh berikut menjelaskan atribut untuk akselerator routing kustom.

```

aws globalaccelerator describe-custom-routing-accelerator-attributes \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh

```

Output:

```

{
  "AcceleratorAttributes": {
    "FlowLogsEnabled": false
  }
}

```



Untuk informasi selengkapnya, lihat [Akselerator perutean khusus di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [DescribeCustomRoutingAcceleratorAttributes](#) di Referensi AWS CLI Perintah.

## **describe-custom-routing-accelerator**

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-routing-accelerator`.

### AWS CLI

Untuk menggambarkan akselerator perutean khusus

`describe-custom-routing-accelerator` Contoh berikut mengambil rincian tentang akselerator routing kustom yang ditentukan.

```
aws globalaccelerator describe-custom-routing-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefg
```

### Output:

```
{
  "Accelerator": {
    "AcceleratorArn":
"arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefg",
    "IpAddressType": "IPV4",
    "Name": "ExampleCustomRoutingAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
          "192.0.2.250",
          "198.51.100.52"
        ],
        "IpFamily": "IPv4"
      }
    ],
    "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
  }
}
```

```
    "CreatedTime": 1542394847,  
    "LastModifiedTime": 1542395013  
  }  
}
```

Untuk informasi selengkapnya, lihat [Akselerator perutean khusus di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [DescribeCustomRoutingAccelerator](#) di Referensi AWS CLI Perintah.

## **describe-custom-routing-endpoint-group**

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-routing-endpoint-group`.

### AWS CLI

Untuk mendeskripsikan grup endpoint untuk akselerator perutean kustom

`describe-custom-routing-endpoint-group` Contoh berikut menjelaskan grup endpoint untuk akselerator routing kustom.

```
aws globalaccelerator describe-custom-routing-endpoint-group \  
  --endpoint-group-arn  
  arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
  abcd-1234abcdefgh/listener/6789vxyz/endpoint-group/ab8888example
```

### Output:

```
{  
  "EndpointGroup": {  
    "EndpointGroupArn":  
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
    abcd-1234abcdefgh/listener/6789vxyz/endpoint-group/ab8888example",  
    "EndpointGroupRegion": "us-east-2",  
    "DestinationDescriptions": [  
      {  
        "FromPort": 5000,  
        "ToPort": 10000,  
        "Protocols": [  
          "UDP"  
        ]  
      }  
    ]  
  }  
}
```

```

    ],
    "EndpointDescriptions": [
      {
        "EndpointId": "subnet-1234567890abcdef0"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Grup titik akhir untuk akselerator perutean kustom di AWS Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [DescribeCustomRoutingEndpointGroup](#) di Referensi AWS CLI Perintah.

## describe-custom-routing-listener

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-routing-listener`.

### AWS CLI

Untuk mendeskripsikan pendengar untuk akselerator perutean kustom

`describe-custom-routing-listener` Contoh berikut menjelaskan pendengar untuk akselerator perutean kustom.

```

aws globalaccelerator describe-custom-routing-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234

```

Output:

```

{
  "Listener": {
    "PortRanges": [
      "FromPort": 5000,
      "ToPort": 10000
    ],
    "ListenerArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234"
  }
}

```

Untuk informasi selengkapnya, lihat [Pendengar untuk akselerator perutean kustom di Global Accelerator di Panduan AWS Pengembang AWS Akselerator Global](#).

- Untuk detail API, lihat [DescribeCustomRoutingListener](#) di Referensi AWS CLI Perintah.

## describe-endpoint-group

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoint-group`.

### AWS CLI

Untuk menggambarkan grup titik akhir

`describe-endpoint-group` Contoh berikut mengambil detail tentang grup titik akhir dengan titik akhir berikut: instans Amazon EC2, ALB, dan NLB.

```
aws globalaccelerator describe-endpoint-group \
  --endpoint-group-arn
  arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-group/
ab8888example
```

Output:

```
{
  "EndpointGroup": {
    "TrafficDialPercentage": 100.0,
    "EndpointDescriptions": [
      {
        "Weight": 128,
        "EndpointId": "i-1234567890abcdef0"
      },
      {
        "Weight": 128,
        "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/app/ALBTesting/alb01234567890xyz"
      },
      {
        "Weight": 128,
        "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/net/NLBTesting/alb01234567890qrs"
      }
    ]
  }
}
```

```

    "EndpointGroupArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-
group/4321abcd-abcd-4321-abcd-4321abcdefg",
    "EndpointGroupRegion": "us-east-1"
  }
}

```

Untuk informasi selengkapnya, lihat [Grup titik akhir di Akselerator AWS Global di Panduan Pengembang](#) Akselerator AWS Global.

- Untuk detail API, lihat [DescribeEndpointGroup](#) di Referensi AWS CLI Perintah.

## describe-listener

Contoh kode berikut menunjukkan cara menggunakan `describe-listener`.

### AWS CLI

Untuk menggambarkan pendengar

`describe-listener` Contoh berikut menjelaskan pendengar.

```

aws globalaccelerator describe-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234

```

Output:

```

{
  "Listener": {
    "ListenerArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
    "PortRanges": [
      {
        "FromPort": 80,
        "ToPort": 80
      }
    ],
    "Protocol": "TCP",
    "ClientAffinity": "NONE"
  }
}

```





**Output:**

```
{
  "ByoipCidrs": [
    {
      "Cidr": "198.51.100.0/24",
      "State": "READY"
    }
    {
      "Cidr": "203.0.113.25/24",
      "State": "READY"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membawa alamat IP Anda sendiri di AWS Global Accelerator di Panduan](#) Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [ListByoipCidr](#) di Referensi AWS CLI Perintah.

**list-custom-routing-accelerators**

Contoh kode berikut menunjukkan cara menggunakan `list-custom-routing-accelerators`.

**AWS CLI**

Untuk membuat daftar akselerator perutean kustom

`list-custom-routing-accelerators` Contoh berikut mencantumkan akselerator perutean kustom di akun AWS .

```
aws globalaccelerator list-custom-routing-accelerators
```

**Output:**

```
{
  "Accelerators": [
    {
      "AcceleratorArn":
        "arn:aws:globalaccelerator::012345678901:accelerator/5555abcd-abcd-5555-
        abcd-5555EXAMPLE1",
      "Name": "TestCustomRoutingAccelerator",
```





## list-custom-routing-endpoint-groups

Contoh kode berikut menunjukkan cara menggunakan `list-custom-routing-endpoint-groups`.

### AWS CLI

Untuk mencantumkan grup titik akhir untuk pendengar dalam akselerator perutean kustom

`list-custom-routing-endpoint-groups` Contoh berikut mencantumkan grup titik akhir untuk pendengar dalam akselerator perutean kustom.

```
aws globalaccelerator list-custom-routing-endpoint-groups \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234
```

### Output:

```
{
  "EndpointGroups": [
    {
      "EndpointGroupArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab8888example",
      "EndpointGroupRegion": "eu-central-1",
      "DestinationDescriptions": [
        {
          "FromPort": 80,
          "ToPort": 80,
          "Protocols": [
            "TCP",
            "UDP"
          ]
        }
      ]
      "EndpointDescriptions": [
        {
          "EndpointId": "subnet-abcd123example"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Grup titik akhir untuk akselerator perutean kustom di AWS Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [ListCustomRoutingEndpointGroups](#) di Referensi AWS CLI Perintah.

## **list-custom-routing-listeners**

Contoh kode berikut menunjukkan cara menggunakan `list-custom-routing-listeners`.

### AWS CLI

Untuk membuat daftar pendengar untuk akselerator perutean kustom

`list-custom-routing-listeners` Contoh berikut mencantumkan pendengar untuk akselerator perutean kustom.

```
aws globalaccelerator list-custom-routing-listeners \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

### Output:

```
{
  "Listeners": [
    {
      "ListenerArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
      "PortRanges": [
        {
          "FromPort": 5000,
          "ToPort": 10000
        }
      ],
      "Protocol": "TCP"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Pendengar untuk akselerator perutean kustom di Global Accelerator di Panduan AWS Pengembang AWS Akselerator Global](#).

- Untuk detail API, lihat [ListCustomRoutingListeners](#) di Referensi AWS CLI Perintah.

## list-custom-routing-port-mappings-by-destination

Contoh kode berikut menunjukkan cara menggunakan `list-custom-routing-port-mappings-by-destination`.

### AWS CLI

Untuk membuat daftar pemetaan port untuk tujuan akselerator perutean kustom tertentu

`list-custom-routing-port-mappings-by-destination` Contoh berikut menyediakan pemetaan port untuk server EC2 tujuan tertentu (di alamat tujuan) untuk akselerator perutean kustom.

```
aws globalaccelerator list-custom-routing-port-mappings-by-destination \
  --endpoint-id subnet-abcd123example \
  --destination-address 198.51.100.52
```

### Output:

```
{
  "DestinationPortMappings": [
    {
      "AcceleratorArn":
        "arn:aws:globalaccelerator::402092451327:accelerator/24ea29b8-
        d750-4489-8919-3095f3c4b0a7",
      "AcceleratorSocketAddresses": [
        {
          "IpAddress": "192.0.2.250",
          "Port": 65514
        },
        {
          "IpAddress": "192.10.100.99",
          "Port": 65514
        }
      ],
      "EndpointGroupArn":
        "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
        abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab88888example",
      "EndpointId": "subnet-abcd123example",
      "EndpointGroupRegion": "us-west-2",
      "DestinationSocketAddress": {
        "IpAddress": "198.51.100.52",
```

```

        "Port": 80
      },
      "IpAddressType": "IPv4",
      "DestinationTrafficState": "ALLOW"
    }
  ]
}

```

Untuk informasi [selengkapnya](#), lihat [Cara kerja akselerator perutean kustom di AWS Global Accelerator](#) di Panduan Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [ListCustomRoutingPortMappingsByDestination](#) di Referensi AWS CLI Perintah.

## list-custom-routing-port-mappings

Contoh kode berikut menunjukkan cara menggunakan `list-custom-routing-port-mappings`.

### AWS CLI

Untuk membuat daftar pemetaan port dalam akselerator perutean kustom

`list-custom-routing-port-mappings` Contoh berikut menyediakan sebagian daftar pemetaan port dalam akselerator routing kustom.

```

aws globalaccelerator list-custom-routing-port-mappings \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefg

```

### Output:

```

{
  "PortMappings": [
    {
      "AcceleratorPort": 40480,
      "EndpointGroupArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefg/listener/0123vxyz/endpoint-group/098765zyxwvu",
      "EndpointId": "subnet-1234567890abcdef0",
      "DestinationSocketAddress": {
        "IpAddress": "192.0.2.250",
        "Port": 80
      }
    }
  ]
}

```

```

    },
    "Protocols": [
        "TCP",
        "UDP"
    ],
    "DestinationTrafficState": "ALLOW"
}
{
    "AcceleratorPort": 40481,
    "EndpointGroupArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
    "EndpointId": "subnet-1234567890abcdef0",
    "DestinationSocketAddress": {
        "IpAddress": "192.0.2.251",
        "Port": 80
    },
    "Protocols": [
        "TCP",
        "UDP"
    ],
    "DestinationTrafficState": "ALLOW"
}
]
}

```

Untuk informasi [selengkapnya, lihat Cara kerja akselerator perutean kustom di AWS Global Accelerator](#) di Panduan Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [ListCustomRoutingPortMappings](#) di Referensi AWS CLI Perintah.

## list-endpoint-groups

Contoh kode berikut menunjukkan cara menggunakan `list-endpoint-groups`.

### AWS CLI

Untuk membuat daftar grup titik akhir

`list-endpoint-groups` Contoh berikut mencantumkan grup endpoint untuk listener. Listener ini memiliki dua grup endpoint.

```
aws globalaccelerator --region us-west-2 list-endpoint-groups \
```

```
--listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234
```

Output:

```
{  
  "EndpointGroups": [  
    {  
      "EndpointGroupArn":  
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab88888example",  
      "EndpointGroupRegion": "eu-central-1",  
      "EndpointDescriptions": [],  
      "TrafficDialPercentage": 100.0,  
      "HealthCheckPort": 80,  
      "HealthCheckProtocol": "TCP",  
      "HealthCheckIntervalSeconds": 30,  
      "ThresholdCount": 3  
    }  
    {  
      "EndpointGroupArn":  
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab99999example",  
      "EndpointGroupRegion": "us-east-1",  
      "EndpointDescriptions": [],  
      "TrafficDialPercentage": 50.0,  
      "HealthCheckPort": 80,  
      "HealthCheckProtocol": "TCP",  
      "HealthCheckIntervalSeconds": 30,  
      "ThresholdCount": 3  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Grup Titik Akhir di Akselerator AWS Global di Panduan Pengembang](#) Akselerator AWS Global.

- Untuk detail API, lihat [ListEndpointGroups](#) di Referensi AWS CLI Perintah.

## list-listeners

Contoh kode berikut menunjukkan cara menggunakan `list-listeners`.

## AWS CLI

Untuk daftar pendengar

`list-listeners` Contoh berikut mencantumkan pendengar untuk akselerator.

```
aws globalaccelerator list-listeners \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
  "Listeners": [
    {
      "ListenerArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
      "PortRanges": [
        {
          "FromPort": 80,
          "ToPort": 80
        }
      ],
      "Protocol": "TCP",
      "ClientAffinity": "NONE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Pendengar di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [ListListeners](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk daftar tag untuk akselerator



`list-tags-for-resource` Contoh berikut mencantumkan tag untuk akselerator tertentu.

```
aws globalaccelerator list-tags-for-resource \  
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Project",  
      "Value": "A123456"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **provision-byoip-cidr**

Contoh kode berikut menunjukkan cara menggunakan `provision-byoip-cidr`.

AWS CLI

Untuk menyediakan rentang alamat

`provision-byoip-cidr` Contoh berikut menyediakan rentang alamat yang ditentukan untuk digunakan dengan AWS sumber daya Anda.

```
aws globalaccelerator provision-byoip-cidr \  
  --cidr 192.0.2.250/24 \  
  --cidr-authorization-context Message="$text_message",Signature="$signed_message"
```

Output:

```
{
```

```
"ByoipCidr": {
  "Cidr": "192.0.2.250/24",
  "State": "PENDING_PROVISIONING"
}
```

Untuk informasi selengkapnya, lihat [Membawa alamat IP Anda sendiri di AWS Global Accelerator di Panduan](#) Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [ProvisionByoipCidr](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai akselerator

`tag-resource` Contoh berikut menambahkan tag Nama dan Proyek ke akselerator, bersama dengan nilai yang sesuai untuk masing-masing.

```
aws globalaccelerator tag-resource \
  --resource-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Akselerator AWS Global di Panduan](#) Pengembang Akselerator AWS Global.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari akselerator

`untag-resource` Contoh berikut menghapus tag Nama dan Proyek dari akselerator.

```
aws globalaccelerator untag-resource \  
  --resource-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh \  
  --tag-keys Key="Name" Key="Project"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-accelerator-attributes**

Contoh kode berikut menunjukkan cara menggunakan `update-accelerator-attributes`.

### AWS CLI

Untuk memperbarui atribut akselerator

`update-accelerator-attributes` Contoh berikut memperbarui akselerator untuk mengaktifkan log aliran. Anda harus menentukan US-West-2 (Oregon) Wilayah untuk membuat atau memperbarui atribut akselerator.

```
aws globalaccelerator update-accelerator-attributes \  
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh \  
  --flow-logs-enabled \  
  --flow-logs-s3-bucket flowlogs-abc \  
  --flow-logs-s3-prefix bucketprefix-abc
```

Output:

```
{  
  "AcceleratorAttributes": {  
    "FlowLogsEnabled": true  
    "FlowLogsS3Bucket": flowlogs-abc  
    "FlowLogsS3Prefix": bucketprefix-abc  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Akselerator di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [UpdateAcceleratorAttributes](#) di Referensi AWS CLI Perintah.

## update-accelerator

Contoh kode berikut menunjukkan cara menggunakan `update-accelerator`.

### AWS CLI

Untuk memperbarui akselerator

`update-accelerator` Contoh berikut memodifikasi akselerator untuk mengubah nama akselerator menjadi `ExampleAcceleratorNew`. Anda harus menentukan `US-West-2` (Oregon) Wilayah untuk membuat atau memperbarui akselerator.

```
aws globalaccelerator update-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --name ExampleAcceleratorNew
```

### Output:

```
{
  "Accelerator": {
    "AcceleratorArn":
      "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleAcceleratorNew",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
          "192.0.2.250",
          "198.51.100.52"
        ],
        "IpFamily": "IPv4"
      }
    ]
  }
}
```

```
    }
  ],
  "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
  "CreatedTime": 1232394847,
  "LastModifiedTime": 1232395654
}
}
```

Untuk informasi selengkapnya, lihat [Akselerator di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [UpdateAccelerator](#) di Referensi AWS CLI Perintah.

## update-custom-routing-accelerator-attributes

Contoh kode berikut menunjukkan cara menggunakan `update-custom-routing-accelerator-attributes`.

### AWS CLI

Untuk memperbarui atribut akselerator perutean kustom

`update-custom-routing-accelerator-attributes` Contoh berikut memperbarui akselerator perutean kustom untuk mengaktifkan log aliran.

```
aws globalaccelerator update-custom-routing-accelerator-attributes \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --flow-logs-enabled \
  --flow-logs-s3-bucket flowlogs-abc \
  --flow-logs-s3-prefix bucketprefix-abc
```

Output:

```
{
  "AcceleratorAttributes": {
    "FlowLogsEnabled": true
    "FlowLogsS3Bucket": flowlogs-abc
    "FlowLogsS3Prefix": bucketprefix-abc
  }
}
```

Untuk informasi selengkapnya, lihat [Akselerator perutean khusus di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [UpdateCustomRoutingAcceleratorAttributes](#) di Referensi AWS CLI Perintah.

## update-custom-routing-accelerator

Contoh kode berikut menunjukkan cara menggunakan `update-custom-routing-accelerator`.

### AWS CLI

Untuk memperbarui akselerator perutean kustom

`update-custom-routing-accelerator` Contoh berikut memodifikasi akselerator routing kustom untuk mengubah nama akselerator.

```
aws globalaccelerator --region us-west-2 update-custom-routing-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --name ExampleCustomRoutingAcceleratorNew
```

Output:

```
{
  "Accelerator": {
    "AcceleratorArn":
    "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleCustomRoutingAcceleratorNew",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
          "192.0.2.250",
          "198.51.100.52"
        ],
        "IpFamily": "IPv4"
      }
    ],
    "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
```

```

    "CreatedTime": 1232394847,
    "LastModifiedTime": 1232395654
  }
}

```

Untuk informasi selengkapnya, lihat [Akselerator perutean khusus di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [UpdateCustomRoutingAccelerator](#) di Referensi AWS CLI Perintah.

## update-custom-routing-listener

Contoh kode berikut menunjukkan cara menggunakan `update-custom-routing-listener`.

### AWS CLI

Untuk memperbarui listener untuk akselerator perutean kustom

`update-custom-routing-listener` Contoh berikut memperbarui listener untuk mengubah rentang port.

```

aws globalaccelerator update-custom-routing-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --port-ranges FromPort=10000,ToPort=20000

```

### Output:

```

{
  "Listener": {
    "ListenerArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz
    "PortRanges": [
      {
        "FromPort": 10000,
        "ToPort": 20000
      }
    ],
    "Protocol": "TCP"
  }
}

```

Untuk informasi selengkapnya, lihat [Pendengar untuk akselerator perutean kustom di Global Accelerator di Panduan AWS Pengembang AWS Akselerator Global](#).

- Untuk detail API, lihat [UpdateCustomRoutingListener](#) di Referensi AWS CLI Perintah.

## update-endpoint-group

Contoh kode berikut menunjukkan cara menggunakan `update-endpoint-group`.

### AWS CLI

Untuk memperbarui grup titik akhir

`update-endpoint-group` Contoh berikut menambahkan tiga titik akhir ke grup endpoint: alamat IP Elastis, ALB, dan NLB.

```
aws globalaccelerator update-endpoint-group \
  --endpoint-group-arn
  arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-group/
ab8888example \
  --endpoint-configurations \
    EndpointId=eipalloc-eip01234567890abc,Weight=128 \
    EndpointId=arn:aws:elasticloadbalancing:us-east-1:000123456789:loadbalancer/
app/ALBTesting/alb01234567890xyz,Weight=128 \
    EndpointId=arn:aws:elasticloadbalancing:us-east-1:000123456789:loadbalancer/
net/NLBTesting/alb01234567890qrs,Weight=128
```

Output:

```
{
  "EndpointGroup": {
    "TrafficDialPercentage": 100,
    "EndpointDescriptions": [
      {
        "Weight": 128,
        "EndpointId": "eip01234567890abc"
      },
      {
        "Weight": 128,
        "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/app/ALBTesting/alb01234567890xyz"
      },
    ]
  }
}
```



```

    {
      "Weight": 128,
      "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/net/NLBTesting/alb01234567890qrs"
    }
  ],
  "EndpointGroupArn":
  "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-
group/4321abcd-abcd-4321-abcd-4321abcdefgh",
  "EndpointGroupRegion": "us-east-1"
}
}

```

Untuk informasi selengkapnya, lihat [Grup titik akhir di Akselerator AWS Global di Panduan Pengembang](#) Akselerator AWS Global.

- Untuk detail API, lihat [UpdateEndpointGroup](#) di Referensi AWS CLI Perintah.

## update-listener

Contoh kode berikut menunjukkan cara menggunakan `update-listener`.

### AWS CLI

Untuk memperbarui pendengar

`update-listener` Contoh berikut memperbarui listener untuk mengubah port ke 100.

```

aws globalaccelerator update-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --port-ranges FromPort=100,ToPort=100

```

Output:

```

{
  "Listener": {
    "ListenerArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz
    "PortRanges": [

```

```
    {
      "FromPort": 100,
      "ToPort": 100
    }
  ],
  "Protocol": "TCP",
  "ClientAffinity": "NONE"
}
```

Untuk informasi selengkapnya, lihat [Pendengar di Akselerator AWS Global di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [UpdateListener](#) di Referensi AWS CLI Perintah.

## **withdraw-byoip-cidr**

Contoh kode berikut menunjukkan cara menggunakan `withdraw-byoip-cidr`.

### AWS CLI

Untuk menarik rentang alamat

`withdraw-byoip-cidr` Contoh berikut menarik rentang alamat dari AWS Global Accelerator yang sebelumnya Anda iklankan untuk digunakan dengan sumber daya Anda. AWS

```
aws globalaccelerator withdraw-byoip-cidr \
  --cidr 192.0.2.250/24
```

Output:

```
{
  "ByoipCidr": {
    "Cidr": "192.0.2.250/24",
    "State": "PENDING_WITHDRAWING"
  }
}
```

Untuk informasi selengkapnya, lihat [Membawa alamat IP Anda sendiri di AWS Global Accelerator di Panduan Pengembang Akselerator AWS Global](#).

- Untuk detail API, lihat [WithdrawByoipCidr](#) di Referensi AWS CLI Perintah.

## AWS Glue contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Glue.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-stop-job-run**

Contoh kode berikut menunjukkan cara menggunakan `batch-stop-job-run`.

AWS CLI

Untuk menghentikan pekerjaan berjalan

`batch-stop-job-run` Contoh berikut menghentikan pekerjaan berjalan.

```
aws glue batch-stop-job-run \  
  --job-name "my-testing-job" \  
  --job-run-id jr_852f1de1f29fb62e0ba4166c33970803935d87f14f96cfdee5089d5274a61d3f
```

Output:

```
{  
  "SuccessfulSubmissions": [  
    {  
      "JobName": "my-testing-job",  
      "JobRunId":  
        "jr_852f1de1f29fb62e0ba4166c33970803935d87f14f96cfdee5089d5274a61d3f"    }  
  ]  
}
```

```

    }
  ],
  "Errors": [],
  "ResponseMetadata": {
    "RequestId": "66bd6b90-01db-44ab-95b9-6aeff0e73d88",
    "HTTPStatusCode": 200,
    "HTTPHeaders": {
      "date": "Fri, 16 Oct 2020 20:54:51 GMT",
      "content-type": "application/x-amz-json-1.1",
      "content-length": "148",
      "connection": "keep-alive",
      "x-amzn-requestid": "66bd6b90-01db-44ab-95b9-6aeff0e73d88"
    },
    "RetryAttempts": 0
  }
}

```

Untuk informasi selengkapnya, lihat [Job Runs](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [BatchStopJobRundi](#) Referensi AWS CLI Perintah.

## create-connection

Contoh kode berikut menunjukkan cara menggunakan `create-connection`.

### AWS CLI

Untuk membuat koneksi untuk penyimpanan data AWS Glue

`create-connection` Contoh berikut membuat koneksi di Katalog Data AWS Glue yang menyediakan informasi koneksi untuk penyimpanan data Kafka.

```

aws glue create-connection \
  --connection-input '{ \
    "Name":"conn-kafka-custom", \
    "Description":"kafka connection with ssl to custom kafka", \
    "ConnectionType":"KAFKA", \
    "ConnectionProperties":{ \
      "KAFKA_BOOTSTRAP_SERVERS":"<Kafka-broker-server-url>:<SSL-Port>", \
      "KAFKA_SSL_ENABLED":"true", \
      "KAFKA_CUSTOM_CERT": "s3://bucket/prefix/cert-file.pem" \
    }, \
    "PhysicalConnectionRequirements":{ \

```

```
"SubnetId":"subnet-1234", \  
  "SecurityGroupIdList":["sg-1234"], \  
  "AvailabilityZone":"us-east-1a"} \  
}' \  
--region us-east-1 \  
--endpoint https://glue.us-east-1.amazonaws.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendefinisikan Koneksi di Katalog Data AWS Glue](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [CreateConnection](#) di Referensi AWS CLI Perintah.

## create-database

Contoh kode berikut menunjukkan cara menggunakan `create-database`.

### AWS CLI

Untuk membuat database

`create-database` Contoh berikut membuat database di AWS Glue Data Catalog.

```
aws glue create-database \  
  --database-input "{\"Name\":\"tempdb\"}" \  
  --profile my_profile \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendefinisikan Database di Katalog Data Anda](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [CreateDatabase](#) di Referensi AWS CLI Perintah.

## create-job

Contoh kode berikut menunjukkan cara menggunakan `create-job`.

### AWS CLI

Untuk membuat pekerjaan untuk mengubah data

`create-job` berikut membuat pekerjaan streaming yang menjalankan skrip yang disimpan di S3.

```
aws glue create-job \  
  --name my-testing-job \  
  --role AWSGlueServiceRoleDefault \  
  --command '{ \  
    "Name": "gluestreaming", \  
    "ScriptLocation": "s3://DOC-EXAMPLE-BUCKET/folder/" \  
  }' \  
  --region us-east-1 \  
  --output json \  
  --default-arguments '{ \  
    "--job-language":"scala", \  
    "--class":"GlueApp" \  
  }' \  
  --profile my-profile \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Isi dari `test_script.scala`:

```
import com.amazonaws.services.glue.ChoiceOption  
import com.amazonaws.services.glue.GlueContext  
import com.amazonaws.services.glue.MappingSpec  
import com.amazonaws.services.glue.ResolveSpec  
import com.amazonaws.services.glue.errors.CallSite  
import com.amazonaws.services.glue.util.GlueArgParser  
import com.amazonaws.services.glue.util.Job  
import com.amazonaws.services.glue.util.JsonOptions  
import org.apache.spark.SparkContext  
import scala.collection.JavaConverters._  
  
object GlueApp {  
  def main(sysArgs: Array[String]) {  
    val spark: SparkContext = new SparkContext()  
    val glueContext: GlueContext = new GlueContext(spark)  
    // @params: [JOB_NAME]  
    val args = GlueArgParser.getResolvedOptions(sysArgs,  
Seq("JOB_NAME").toArray)  
    Job.init(args("JOB_NAME"), glueContext, args.asJava)  
    // @type: DataSource  
    // @args: [database = "tempdb", table_name = "s3-source", transformation_ctx  
= "datasource0"]
```

```

    // @return: datasource0
    // @inputs: []
    val datasource0 = glueContext.getCatalogSource(database = "tempdb",
tableName = "s3-source", redshiftTmpDir = "", transformationContext =
"datasource0").getDynamicFrame()
    // @type: ApplyMapping
    // @args: [mapping = [("sensorid", "int", "sensorid", "int"),
("currenttemperature", "int", "currenttemperature", "int"), ("status", "string",
"status", "string")], transformation_ctx = "applymapping1"]
    // @return: applymapping1
    // @inputs: [frame = datasource0]
    val applymapping1 = datasource0.applyMapping(mappings = Seq(("sensorid",
"int", "sensorid", "int"), ("currenttemperature", "int", "currenttemperature",
"int"), ("status", "string", "status", "string")), caseSensitive = false,
transformationContext = "applymapping1")
    // @type: SelectFields
    // @args: [paths = ["sensorid", "currenttemperature", "status"],
transformation_ctx = "selectfields2"]
    // @return: selectfields2
    // @inputs: [frame = applymapping1]
    val selectfields2 = applymapping1.selectFields(paths = Seq("sensorid",
"currenttemperature", "status"), transformationContext = "selectfields2")
    // @type: ResolveChoice
    // @args: [choice = "MATCH_CATALOG", database = "tempdb", table_name = "my-
s3-sink", transformation_ctx = "resolvechoice3"]
    // @return: resolvechoice3
    // @inputs: [frame = selectfields2]
    val resolvechoice3 = selectfields2.resolveChoice(choiceOption =
Some(ChoiceOption("MATCH_CATALOG")), database = Some("tempdb"), tableName =
Some("my-s3-sink"), transformationContext = "resolvechoice3")
    // @type: DataSink
    // @args: [database = "tempdb", table_name = "my-s3-sink",
transformation_ctx = "datasink4"]
    // @return: datasink4
    // @inputs: [frame = resolvechoice3]
    val datasink4 = glueContext.getCatalogSink(database = "tempdb",
tableName = "my-s3-sink", redshiftTmpDir = "", transformationContext =
"datasink4").writeDynamicFrame(resolvechoice3)
    Job.commit()
}
}

```

Output:

```
{
  "Name": "my-testing-job"
}
```

Untuk informasi selengkapnya, lihat [Menulis Pekerjaan di AWS Glue](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [CreateJob](#) di Referensi AWS CLI Perintah.

## create-table

Contoh kode berikut menunjukkan cara menggunakan `create-table`.

### AWS CLI

Contoh 1: Untuk membuat tabel untuk aliran data Kinesis

`create-table` Contoh berikut membuat tabel dalam AWS Glue Data Catalog yang menjelaskan aliran data Kinesis.

```
aws glue create-table \
  --database-name tempdb \
  --table-input '{"Name":"test-kinesis-input", "StorageDescriptor":{ \
    "Columns":[ \
      {"Name":"sensorid", "Type":"int"}, \
      {"Name":"currenttemperature", "Type":"int"}, \
      {"Name":"status", "Type":"string"} \
    ], \
    "Location":"my-testing-stream", \
    "Parameters":{ \
      "typeOfData":"kinesis", "streamName":"my-testing-stream", \
      "kinesisUrl":"https://kinesis.us-east-1.amazonaws.com" \
    }, \
    "SerdeInfo":{ \
      "SerializationLibrary":"org.openx.data.jsonserde.JsonSerDe"} \
  }, \
  "Parameters":{ \
    "classification":"json"} \
}' \
--profile my-profile \
--endpoint https://glue.us-east-1.amazonaws.com
```



Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendefinisikan Tabel di Katalog Data AWS Glue](#) di Panduan Pengembang AWS Glue.

Contoh 2: Untuk membuat tabel untuk penyimpanan data Kafka

`create-table` Contoh berikut membuat tabel di AWS Glue Data Catalog yang menjelaskan penyimpanan data Kafka.

```
aws glue create-table \  
  --database-name tempdb \  
  --table-input '{"Name":"test-kafka-input", "StorageDescriptor":{ \  
    "Columns":[ \  
      {"Name":"sensorid", "Type":"int"}, \  
      {"Name":"currenttemperature", "Type":"int"}, \  
      {"Name":"status", "Type":"string"} \  
    ], \  
    "Location":"glue-topic", \  
    "Parameters":{ \  
      "typeOfData":"kafka","topicName":"glue-topic", \  
      "connectionName":"my-kafka-connection" \  
    }, \  
    "SerdeInfo":{ \  
      "SerializationLibrary":"org.apache.hadoop.hive.serde2.OpenCSVSerde"} \  
  } \  
  }, \  
  "Parameters":{ \  
    "separatorChar":"," } \  
}' \  
  --profile my-profile \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendefinisikan Tabel di Katalog Data AWS Glue](#) di Panduan Pengembang AWS Glue.

Contoh 3: Untuk membuat tabel untuk penyimpanan data AWS S3

`create-table` Contoh berikut membuat tabel di AWS Glue Data Catalog yang menjelaskan penyimpanan data AWS Simple Storage Service (AWS S3).

```
aws glue create-table \  
  --database-name tempdb \  
  --table-input '{"Name":"s3-output", "StorageDescriptor":{ \  
    "Columns":[ \  
      {"Name":"s1", "Type":"string"}, \  
      {"Name":"s2", "Type":"int"}, \  
      {"Name":"s3", "Type":"string"} \  
    ], \  
    "Location":"s3://bucket-path/", \  
    "SerdeInfo":{ \  
      "SerializationLibrary":"org.openx.data.jsonserde.JsonSerDe"} \  
    }, \  
    "Parameters":{ \  
      "classification":"json"} \  
  }' \  
  --profile my-profile \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendefinisikan Tabel di Katalog Data AWS Glue](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [CreateTable](#) di Referensi AWS CLI Perintah.

## delete-job

Contoh kode berikut menunjukkan cara menggunakan `delete-job`.

### AWS CLI

Untuk menghapus pekerjaan

`delete-job` Contoh berikut menghapus pekerjaan yang tidak lagi diperlukan.

```
aws glue delete-job \  
  --job-name my-testing-job
```

Output:

```
{  
  "JobName": "my-testing-job"
```

```
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Pekerjaan di AWS Glue Console](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [DeleteJob](#) di Referensi AWS CLI Perintah.

## get-databases

Contoh kode berikut menunjukkan cara menggunakan `get-databases`.

### AWS CLI

Untuk membuat daftar definisi dari beberapa atau semua database dalam Katalog Data AWS Glue

`get-databases` Contoh berikut mengembalikan informasi tentang database dalam Katalog Data.

```
aws glue get-databases
```

Output:

```
{
  "DatabaseList": [
    {
      "Name": "default",
      "Description": "Default Hive database",
      "LocationUri": "file:/spark-warehouse",
      "CreateTime": 1602084052.0,
      "CreateTableDefaultPermissions": [
        {
          "Principal": {
            "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
          },
          "Permissions": [
            "ALL"
          ]
        }
      ],
      "CatalogId": "111122223333"
    },
    {
      "Name": "flights-db",
```

```
    "CreateTime": 1587072847.0,
    "CreateTableDefaultPermissions": [
      {
        "Principal": {
          "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
        },
        "Permissions": [
          "ALL"
        ]
      }
    ],
    "CatalogId": "111122223333"
  },
  {
    "Name": "legislators",
    "CreateTime": 1601415625.0,
    "CreateTableDefaultPermissions": [
      {
        "Principal": {
          "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
        },
        "Permissions": [
          "ALL"
        ]
      }
    ],
    "CatalogId": "111122223333"
  },
  {
    "Name": "tempdb",
    "CreateTime": 1601498566.0,
    "CreateTableDefaultPermissions": [
      {
        "Principal": {
          "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
        },
        "Permissions": [
          "ALL"
        ]
      }
    ],
    "CatalogId": "111122223333"
  }
]
```

```
}
```

Untuk informasi selengkapnya, lihat [Mendefinisikan Database di Katalog Data Anda](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [GetDatabases](#) di Referensi AWS CLI Perintah.

## get-job-run

Contoh kode berikut menunjukkan cara menggunakan `get-job-run`.

### AWS CLI

Untuk mendapatkan informasi tentang menjalankan pekerjaan

`get-job-run` Contoh berikut mengambil informasi tentang menjalankan pekerjaan.

```
aws glue get-job-run \
  --job-name "Combine legislators data" \
  --run-id jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e
```

### Output:

```
{
  "JobRun": {
    "Id": "jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e",
    "Attempt": 0,
    "JobName": "Combine legislators data",
    "StartedOn": 1602873931.255,
    "LastModifiedOn": 1602874075.985,
    "CompletedOn": 1602874075.985,
    "JobRunState": "SUCCEEDED",
    "Arguments": {
      "--enable-continuous-cloudwatch-log": "true",
      "--enable-metrics": "",
      "--enable-spark-ui": "true",
      "--job-bookmark-option": "job-bookmark-enable",
      "--spark-event-logs-path": "s3://aws-glue-assets-111122223333-us-east-1/
sparkHistoryLogs/"
    },
    "PredecessorRuns": [],
    "AllocatedCapacity": 10,
    "ExecutionTime": 117,
  }
}
```

```
    "Timeout": 2880,  
    "MaxCapacity": 10.0,  
    "WorkerType": "G.1X",  
    "NumberOfWorkers": 10,  
    "LogGroupName": "/aws-glue/jobs",  
    "GlueVersion": "2.0"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Job Runs](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [GetJobRun](#) di Referensi AWS CLI Perintah.

## get-job-runs

Contoh kode berikut menunjukkan cara menggunakan `get-job-runs`.

### AWS CLI

Untuk mendapatkan informasi tentang semua pekerjaan yang dijalankan untuk suatu pekerjaan

`get-job-runs` Contoh berikut mengambil informasi tentang pekerjaan berjalan untuk suatu pekerjaan.

```
aws glue get-job-runs \  
  --job-name "my-testing-job"
```

Output:

```
{  
  "JobRuns": [  
    {  
      "Id":  
"jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e",  
      "Attempt": 0,  
      "JobName": "my-testing-job",  
      "StartedOn": 1602873931.255,  
      "LastModifiedOn": 1602874075.985,  
      "CompletedOn": 1602874075.985,  
      "JobRunState": "SUCCEEDED",  
      "Arguments": {  
        "--enable-continuous-cloudwatch-log": "true",  
        "--enable-metrics": "",  
      }  
    }  
  ]  
}
```

```

        "--enable-spark-ui": "true",
        "--job-bookmark-option": "job-bookmark-enable",
        "--spark-event-logs-path": "s3://aws-glue-assets-111122223333-us-
east-1/sparkHistoryLogs/"
    },
    "PredecessorRuns": [],
    "AllocatedCapacity": 10,
    "ExecutionTime": 117,
    "Timeout": 2880,
    "MaxCapacity": 10.0,
    "WorkerType": "G.1X",
    "NumberOfWorkers": 10,
    "LogGroupName": "/aws-glue/jobs",
    "GlueVersion": "2.0"
},
{
    "Id":
"jr_03cc19ddb11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_2",
    "Attempt": 2,
    "PreviousRunId":
"jr_03cc19ddb11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_1",
    "JobName": "my-testing-job",
    "StartedOn": 1602811168.496,
    "LastModifiedOn": 1602811282.39,
    "CompletedOn": 1602811282.39,
    "JobRunState": "FAILED",
    "ErrorMessage": "An error occurred while calling
o122.pyWriteDynamicFrame.
        Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
AccessDenied;
        Request ID: 021AAB703DB20A2D;
        S3 Extended Request ID: teZk24Y09TkXzBvMPG502L5VJBhe9DJuWA9/
TXtuG0qfByajkfL/Tlqt5JBGdEGpigAqzdMDM/U=)",
    "PredecessorRuns": [],
    "AllocatedCapacity": 10,
    "ExecutionTime": 110,
    "Timeout": 2880,
    "MaxCapacity": 10.0,
    "WorkerType": "G.1X",
    "NumberOfWorkers": 10,
    "LogGroupName": "/aws-glue/jobs",
    "GlueVersion": "2.0"
},
{

```

```

      "Id":
      "jr_03cc19ddab11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_1",
      "Attempt": 1,
      "PreviousRunId":
      "jr_03cc19ddab11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f",
      "JobName": "my-testing-job",
      "StartedOn": 1602811020.518,
      "LastModifiedOn": 1602811138.364,
      "CompletedOn": 1602811138.364,
      "JobRunState": "FAILED",
      "ErrorMessage": "An error occurred while calling
o122.pyWriteDynamicFrame.
          Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
AccessDenied;
          Request ID: 2671D37856AE7ABB;
          S3 Extended Request ID: RLJCJw20brV
+PpC6Gp0RahyF2fp9f1B5SSb2bTGPnUSPVizLXR11PN3QZ1db+v1o9qRVktNYbW8=)",
      "PredecessorRuns": [],
      "AllocatedCapacity": 10,
      "ExecutionTime": 113,
      "Timeout": 2880,
      "MaxCapacity": 10.0,
      "WorkerType": "G.1X",
      "NumberOfWorkers": 10,
      "LogGroupName": "/aws-glue/jobs",
      "GlueVersion": "2.0"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Job Runs](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [GetJobRuns](#) di Referensi AWS CLI Perintah.

## get-job

Contoh kode berikut menunjukkan cara menggunakan `get-job`.

### AWS CLI

Untuk mengambil informasi tentang pekerjaan

`get-job` Contoh berikut mengambil informasi tentang pekerjaan.



```
aws glue get-job \  
  --job-name my-testing-job
```

### Output:

```
{  
  "Job": {  
    "Name": "my-testing-job",  
    "Role": "Glue_DefaultRole",  
    "CreatedOn": 1602805698.167,  
    "LastModifiedOn": 1602805698.167,  
    "ExecutionProperty": {  
      "MaxConcurrentRuns": 1  
    },  
    "Command": {  
      "Name": "gluestreaming",  
      "ScriptLocation": "s3://janetst-bucket-01/Scripts/test_script.scala",  
      "PythonVersion": "2"  
    },  
    "DefaultArguments": {  
      "--class": "GlueApp",  
      "--job-language": "scala"  
    },  
    "MaxRetries": 0,  
    "AllocatedCapacity": 10,  
    "MaxCapacity": 10.0,  
    "GlueVersion": "1.0"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Pekerjaan](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [GetJob](#) di Referensi AWS CLI Perintah.

## get-plan

Contoh kode berikut menunjukkan cara menggunakan `get-plan`.

### AWS CLI

Untuk mendapatkan kode yang dihasilkan untuk memetakan data dari tabel sumber ke tabel target

Berikut ini `get-plan` mengambil kode yang dihasilkan untuk memetakan kolom dari sumber data ke target data.

```
aws glue get-plan --mapping '[ \
  { \
    "SourcePath":"sensorid", \
    "SourceTable":"anything", \
    "SourceType":"int", \
    "TargetPath":"sensorid", \
    "TargetTable":"anything", \
    "TargetType":"int" \
  }, \
  { \
    "SourcePath":"currenttemperature", \
    "SourceTable":"anything", \
    "SourceType":"int", \
    "TargetPath":"currenttemperature", \
    "TargetTable":"anything", \
    "TargetType":"int" \
  }, \
  { \
    "SourcePath":"status", \
    "SourceTable":"anything", \
    "SourceType":"string", \
    "TargetPath":"status", \
    "TargetTable":"anything", \
    "TargetType":"string" \
  }]' \
--source '{ \
  "DatabaseName":"tempdb", \
  "TableName":"s3-source" \
}' \
--sinks '[ \
  { \
    "DatabaseName":"tempdb", \
    "TableName":"my-s3-sink" \
  }]' \
--language "scala" \
--endpoint https://glue.us-east-1.amazonaws.com \
--output "text"
```

Output:

```
import com.amazonaws.services.glue.ChoiceOption
import com.amazonaws.services.glue.GlueContext
import com.amazonaws.services.glue.MappingSpec
import com.amazonaws.services.glue.ResolveSpec
import com.amazonaws.services.glue.errors.CallSite
import com.amazonaws.services.glue.util.GlueArgParser
import com.amazonaws.services.glue.util.Job
import com.amazonaws.services.glue.util.JsonOptions
import org.apache.spark.SparkContext
import scala.collection.JavaConverters._

object GlueApp {
  def main(sysArgs: Array[String]) {
    val spark: SparkContext = new SparkContext()
    val glueContext: GlueContext = new GlueContext(spark)
    // @params: [JOB_NAME]
    val args = GlueArgParser.getResolvedOptions(sysArgs, Seq("JOB_NAME").toArray)
    Job.init(args("JOB_NAME"), glueContext, args.asJava)
    // @type: DataSource
    // @args: [database = "tempdb", table_name = "s3-source", transformation_ctx =
"datasource0"]
    // @return: datasource0
    // @inputs: []
    val datasource0 = glueContext.getCatalogSource(database = "tempdb",
tableName = "s3-source", redshiftTmpDir = "", transformationContext =
"datasource0").getDynamicFrame()
    // @type: ApplyMapping
    // @args: [mapping = [("sensorid", "int", "sensorid", "int"),
("currenttemperature", "int", "currenttemperature", "int"), ("status", "string",
"status", "string")], transformation_ctx = "applymapping1"]
    // @return: applymapping1
    // @inputs: [frame = datasource0]
    val applymapping1 = datasource0.applyMapping(mappings = Seq(("sensorid",
"int", "sensorid", "int"), ("currenttemperature", "int", "currenttemperature",
"int"), ("status", "string", "status", "string")), caseSensitive = false,
transformationContext = "applymapping1")
    // @type: SelectFields
    // @args: [paths = ["sensorid", "currenttemperature", "status"],
transformation_ctx = "selectfields2"]
    // @return: selectfields2
    // @inputs: [frame = applymapping1]
    val selectfields2 = applymapping1.selectFields(paths = Seq("sensorid",
"currenttemperature", "status"), transformationContext = "selectfields2")
```

```

// @type: ResolveChoice
// @args: [choice = "MATCH_CATALOG", database = "tempdb", table_name = "my-s3-
sink", transformation_ctx = "resolvechoice3"]
// @return: resolvechoice3
// @inputs: [frame = selectfields2]
val resolvechoice3 = selectfields2.resolveChoice(choiceOption =
Some(ChoiceOption("MATCH_CATALOG")), database = Some("tempdb"), tableName =
Some("my-s3-sink"), transformationContext = "resolvechoice3")
// @type: DataSink
// @args: [database = "tempdb", table_name = "my-s3-sink", transformation_ctx =
"datasink4"]
// @return: datasink4
// @inputs: [frame = resolvechoice3]
val datasink4 = glueContext.getCatalogSink(database = "tempdb",
tableName = "my-s3-sink", redshiftTmpDir = "", transformationContext =
"datasink4").writeDynamicFrame(resolvechoice3)
Job.commit()
}
}

```

Untuk informasi selengkapnya, lihat [Mengedit Skrip di AWS Glue](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [GetPlan](#) di Referensi AWS CLI Perintah.

## get-tables

Contoh kode berikut menunjukkan cara menggunakan `get-tables`.

### AWS CLI

Untuk daftar definisi dari beberapa atau semua tabel dalam database yang ditentukan

`get-tables` Contoh berikut mengembalikan informasi tentang tabel dalam database tertentu.

```
aws glue get-tables --database-name 'tempdb'
```

Output:

```
{
  "TableList": [
    {
      "Name": "my-s3-sink",
```

```
"DatabaseName": "tempdb",
"CreateTime": 1602730539.0,
"UpdateTime": 1602730539.0,
"Retention": 0,
"StorageDescriptor": {
  "Columns": [
    {
      "Name": "sensorid",
      "Type": "int"
    },
    {
      "Name": "currenttemperature",
      "Type": "int"
    },
    {
      "Name": "status",
      "Type": "string"
    }
  ],
  "Location": "s3://janetst-bucket-01/test-s3-output/",
  "Compressed": false,
  "NumberOfBuckets": 0,
  "SerdeInfo": {
    "SerializationLibrary": "org.openx.data.jsonserde.JsonSerDe"
  },
  "SortColumns": [],
  "StoredAsSubDirectories": false
},
"Parameters": {
  "classification": "json"
},
"CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
"IsRegisteredWithLakeFormation": false,
"CatalogId": "007436865787"
},
{
  "Name": "s3-source",
  "DatabaseName": "tempdb",
  "CreateTime": 1602730658.0,
  "UpdateTime": 1602730658.0,
  "Retention": 0,
  "StorageDescriptor": {
    "Columns": [
      {
```

```
        "Name": "sensorid",
        "Type": "int"
    },
    {
        "Name": "currenttemperature",
        "Type": "int"
    },
    {
        "Name": "status",
        "Type": "string"
    }
],
"Location": "s3://janetst-bucket-01/",
"Compressed": false,
"NumberOfBuckets": 0,
"SortColumns": [],
"StoredAsSubDirectories": false
},
"Parameters": {
    "classification": "json"
},
"CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
"IsRegisteredWithLakeFormation": false,
"CatalogId": "007436865787"
},
{
    "Name": "test-kinesis-input",
    "DatabaseName": "tempdb",
    "CreateTime": 1601507001.0,
    "UpdateTime": 1601507001.0,
    "Retention": 0,
    "StorageDescriptor": {
        "Columns": [
            {
                "Name": "sensorid",
                "Type": "int"
            },
            {
                "Name": "currenttemperature",
                "Type": "int"
            },
            {
                "Name": "status",
                "Type": "string"
            }
        ]
    }
}
```

```

        }
    ],
    "Location": "my-testing-stream",
    "Compressed": false,
    "NumberOfBuckets": 0,
    "SerdeInfo": {
        "SerializationLibrary": "org.openx.data.jsonserde.JsonSerDe"
    },
    "SortColumns": [],
    "Parameters": {
        "kinesisUrl": "https://kinesis.us-east-1.amazonaws.com",
        "streamName": "my-testing-stream",
        "typeOfData": "kinesis"
    },
    "StoredAsSubDirectories": false
},
"Parameters": {
    "classification": "json"
},
"CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
"IsRegisteredWithLakeFormation": false,
"CatalogId": "007436865787"
}
]
}

```

Untuk informasi selengkapnya, lihat [Mendefinisikan Tabel di Katalog Data AWS Glue](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [GetTables](#) di Referensi AWS CLI Perintah.

## start-crawler

Contoh kode berikut menunjukkan cara menggunakan `start-crawler`.

### AWS CLI

Untuk memulai crawler

`start-crawler` Contoh berikut memulai crawler.

```
aws glue start-crawler --name my-crawler
```

**Output:**

```
None
```

Untuk informasi selengkapnya, lihat [Mendefinisikan Crawler di Panduan](#) Pengembang AWS Glue.

- Untuk detail API, lihat [StartCrawler](#) di Referensi AWS CLI Perintah.

**start-job-run**

Contoh kode berikut menunjukkan cara menggunakan `start-job-run`.

**AWS CLI**

Untuk mulai menjalankan pekerjaan

`start-job-run` Contoh berikut memulai pekerjaan.

```
aws glue start-job-run \  
  --job-name my-job
```

**Output:**

```
{  
  "JobRunId":  
  "jr_22208b1f44eb5376a60569d4b21dd20fcb8621e1a366b4e7b2494af764b82ded"  
}
```

Untuk informasi selengkapnya, lihat [Menulis Pekerjaan](#) di Panduan Pengembang AWS Glue.

- Untuk detail API, lihat [StartJobRun](#) di Referensi AWS CLI Perintah.

**GuardDuty contoh menggunakan AWS CLI**

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with GuardDuty.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.



Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **accept-invitation**

Contoh kode berikut menunjukkan cara menggunakan `accept-invitation`.

AWS CLI

Untuk menerima undangan untuk menjadi akun GuardDuty anggota di wilayah saat ini

`accept-invitation` Contoh berikut menunjukkan cara menerima undangan untuk menjadi akun GuardDuty anggota di wilayah saat ini.

```
aws guardduty accept-invitation \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --master-id 123456789111 \
  --invitation-id d6b94fb03a66ff665f7db8764example
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola GuardDuty Akun berdasarkan Undangan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [AcceptInvitation](#) di Referensi AWS CLI Perintah.

### **archive-findings**

Contoh kode berikut menunjukkan cara menggunakan `archive-findings`.

AWS CLI

Untuk mengarsipkan temuan di wilayah saat ini

Contoh ini menunjukkan cara mengarsipkan temuan di wilayah saat ini.

```
aws guardduty archive-findings \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --finding-ids d6b94fb03a66ff665f7db8764example 3eb970e0de00c16ec14e6910fexample
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola GuardDuty Akun berdasarkan Undangan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [ArchiveFindings](#) di Referensi AWS CLI Perintah.

## create-detector

Contoh kode berikut menunjukkan cara menggunakan `create-detector`.

### AWS CLI

Untuk mengaktifkan GuardDuty di wilayah saat ini

Contoh ini menunjukkan cara membuat detektor baru, yang memungkinkan GuardDuty, di wilayah saat ini. :

```
aws guardduty create-detector \  
  --enable
```

Output:

```
{  
  "DetectorId": "b6b992d6d2f48e64bc59180bfexample"  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Amazon GuardDuty](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreateDetector](#) di Referensi AWS CLI Perintah.

## create-filter

Contoh kode berikut menunjukkan cara menggunakan `create-filter`.

## AWS CLI

Untuk membuat filter baru untuk wilayah saat ini

Contoh ini membuat filter yang cocok dengan semua portscan temuan misalnya dibuat dari gambar tertentu. :

```
aws guardduty create-filter \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --action ARCHIVE \  
  --name myFilter \  
  --finding-criteria '{"Criterion": {"type": {"Eq": ["Recon:EC2/  
Portscan"]}, "resource.instanceDetails.imageId": {"Eq": ["ami-0a7a207083example"]}}}'
```

Output:

```
{  
  "Name": "myFilter"  
}
```

Untuk informasi selengkapnya, lihat [Memfilter temuan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreateFilter](#) di Referensi AWS CLI Perintah.

## create-ip-set

Contoh kode berikut menunjukkan cara menggunakan `create-ip-set`.

## AWS CLI

Untuk membuat set IP tepercaya

`create-ip-set` Contoh berikut membuat dan mengaktifkan set IP tepercaya di wilayah saat ini.

```
aws guardduty create-ip-set \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --name new-ip-set \  
  --format TXT \  
  --location s3://AWSDOC-EXAMPLE-BUCKET/customtrustlist.csv \  
  --activate
```

Output:

```
{
  "IpSetId": "d4b94fc952d6912b8f3060768example"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Daftar IP Tepercaya dan Daftar Ancaman](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreateIpSet](#) di Referensi AWS CLI Perintah.

## create-members

Contoh kode berikut menunjukkan cara menggunakan `create-members`.

### AWS CLI

Untuk mengaitkan anggota baru dengan akun GuardDuty master Anda di wilayah saat ini.

Contoh ini menunjukkan cara mengaitkan akun anggota yang akan dikelola oleh akun saat ini sebagai GuardDuty master.

```
aws guardduty create-members
  --detector-id b6b992d6d2f48e64bc59180bfexample \
  --account-details AccountId=111122223333,Email=first+member@example.com
  AccountId=111111111111 ,Email=another+member@example.com
```

Output:

```
{
  "UnprocessedAccounts": []
}
```

Untuk informasi selengkapnya, lihat [Mengelola beberapa akun](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreateMembers](#) di Referensi AWS CLI Perintah.

## create-publishing-destination

Contoh kode berikut menunjukkan cara menggunakan `create-publishing-destination`.

### AWS CLI

Untuk membuat tujuan penerbitan untuk mengeksport GuardDuty temuan di wilayah saat ini ke.

Contoh ini menunjukkan cara membuat tujuan penerbitan untuk GuardDuty temuan.

```
aws guardduty create-publishing-destination \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --destination-type S3 \  
  --destination-properties  
    DestinationArn=arn:aws:s3:::yourbucket,KmsKeyArn=arn:aws:kms:us-  
west-1:111122223333:key/84cee9c5-dea1-401a-ab6d-e1de7example
```

Output:

```
{  
  "DestinationId": "46b99823849e1bbc24dfbe3cexample"  
}
```

Untuk informasi selengkapnya, lihat [Mengeksplor temuan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreatePublishingDestination](#) di Referensi AWS CLI Perintah.

## create-sample-findings

Contoh kode berikut menunjukkan cara menggunakan `create-sample-findings`.

AWS CLI

Untuk membuat GuardDuty temuan sampel di wilayah saat ini.

Contoh ini menunjukkan cara membuat temuan sampel dari jenis yang disediakan.

```
aws guardduty create-sample-findings \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --finding-types UnauthorizedAccess:EC2/TorClient UnauthorizedAccess:EC2/TorRelay
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Contoh temuan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreateSampleFindings](#) di Referensi AWS CLI Perintah.

## create-threat-intel-set

Contoh kode berikut menunjukkan cara menggunakan `create-threat-intel-set`.

## AWS CLI

Untuk membuat ancaman baru intel ditetapkan di wilayah saat ini.

Contoh ini menunjukkan cara mengunggah intel ancaman yang disetel GuardDuty dan segera mengaktifkannya.

```
aws guardduty create-threat-intel-set \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --name myThreatSet \  
  --format TXT \  
  --location s3://EXAMPLEBUCKET/threatlist.csv \  
  --activate
```

Output:

```
{  
  "ThreatIntelSetId": "20b9a4691aeb33506b808878cexample"  
}
```

Untuk informasi selengkapnya, lihat [IP tepercaya dan daftar ancaman](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [CreateThreatIntelSet](#) di Referensi AWS CLI Perintah.

## decline-invitations

Contoh kode berikut menunjukkan cara menggunakan `decline-invitations`.

## AWS CLI

Untuk menolak undangan agar Guardduty dikelola oleh akun lain di wilayah saat ini.

Contoh ini menunjukkan cara menolak undangan keanggotaan.

```
aws guardduty decline-invitations \  
  --account-ids 111122223333
```

Output:

```
{
```

```
"UnprocessedAccounts": []  
}
```

Untuk informasi selengkapnya, lihat [Mengelola GuardDuty akun berdasarkan undangan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [DeclineInvitations](#) di Referensi AWS CLI Perintah.

## delete-detector

Contoh kode berikut menunjukkan cara menggunakan `delete-detector`.

### AWS CLI

Untuk menghapus detektor, dan menonaktifkan GuardDuty, di wilayah saat ini.

Contoh ini menunjukkan cara menghapus detektor, jika berhasil, ini akan menonaktifkan GuardDuty di wilayah yang terkait dengan detektor itu.

```
aws guardduty delete-detector \  
  --detector-id b6b992d6d2f48e64bc59180bfexample
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menangguhkan atau menonaktifkan GuardDuty di Panduan Pengguna](#). GuardDuty

- Untuk detail API, lihat [DeleteDetector](#) di Referensi AWS CLI Perintah.

## delete-filter

Contoh kode berikut menunjukkan cara menggunakan `delete-filter`.

### AWS CLI

Untuk menghapus filter yang ada di wilayah saat ini

Contoh ini menunjukkan cara membuat menghapus filter.

```
aws guardduty delete-filter \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --filter-name byebyeFilter
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memfilter temuan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [DeleteFilter](#) di Referensi AWS CLI Perintah.

## **disable-organization-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `disable-organization-admin-account`.

### AWS CLI

Untuk menghapus akun sebagai administrator yang didelegasikan untuk GuardDuty dalam organisasi Anda

Contoh ini menunjukkan cara menghapus akun sebagai administrator yang didelegasikan untuk GuardDuty.

```
aws guardduty disable-organization-admin-account \  
  --admin-account-id 111122223333
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun dengan AWS organisasi](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [DisableOrganizationAdminAccount](#) di Referensi AWS CLI Perintah.

## **disassociate-from-master-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-from-master-account`.

### AWS CLI

Untuk memisahkan diri dari akun master Anda saat ini di wilayah saat ini

`disassociate-from-master-account` Contoh berikut memisahkan akun Anda dari akun GuardDuty master saat ini di wilayah saat ini. AWS

```
aws guardduty disassociate-from-master-account \  
  --detector-id d4b040365221be2b54a6264dcexample
```



Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memahami Hubungan antara Akun GuardDuty Master dan Anggota](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [DisassociateFromMasterAccount](#) di Referensi AWS CLI Perintah.

## get-detector

Contoh kode berikut menunjukkan cara menggunakan `get-detector`.

### AWS CLI

Untuk mengambil detail detektor tertentu

`get-detector` Contoh berikut menampilkan rincian konfigurasi detektor yang ditentukan.

```
aws guardduty get-detector \
  --detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{
  "Status": "ENABLED",
  "ServiceRole": "arn:aws:iam::111122223333:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",
  "Tags": {},
  "FindingPublishingFrequency": "SIX_HOURS",
  "UpdatedAt": "2018-11-07T03:24:22.938Z",
  "CreatedAt": "2017-12-22T22:51:31.940Z"
}
```

Untuk informasi selengkapnya, lihat [Konsep dan Terminologi](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [GetDetector](#) di Referensi AWS CLI Perintah.

## get-findings

Contoh kode berikut menunjukkan cara menggunakan `get-findings`.

### AWS CLI

Contoh 1: Untuk mengambil rincian temuan tertentu

`get-findings` Contoh berikut mengambil rincian temuan JSON lengkap dari temuan yang ditentukan.

```
aws guardduty get-findings \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --finding-id 1ab92989eaf0e742df4a014d5example
```

Output:

```
{  
  "Findings": [  
    {  
      "Resource": {  
        "ResourceType": "AccessKey",  
        "AccessKeyDetails": {  
          "UserName": "testuser",  
          "UserType": "IAMUser",  
          "PrincipalId": "AIDACKCEVSQ6C2EXAMPLE",  
          "AccessKeyId": "ASIASZ4SI7REEEXAMPLE"  
        }  
      },  
      "Description": "APIs commonly used to discover the users, groups,  
policies and permissions in an account, was invoked by IAM principal testuser under  
unusual circumstances. Such activity is not typically seen from this principal.",  
      "Service": {  
        "Count": 5,  
        "Archived": false,  
        "ServiceName": "guardduty",  
        "EventFirstSeen": "2020-05-26T22:02:24Z",  
        "ResourceRole": "TARGET",  
        "EventLastSeen": "2020-05-26T22:33:55Z",  
        "DetectorId": "d4b040365221be2b54a6264dcexample",  
        "Action": {  
          "ActionType": "AWS_API_CALL",  
          "AwsApiCallAction": {  
            "RemoteIpDetails": {  
              "GeoLocation": {  
                "Lat": 51.5164,  
                "Lon": -0.093  
              },  
              "City": {  
                "CityName": "London"  
              }  
            }  
          }  
        }  
      }  
    ]  
  }
```

```

        "IpAddressV4": "52.94.36.7",
        "Organization": {
            "Org": "Amazon.com",
            "Isp": "Amazon.com",
            "Asn": "16509",
            "AsnOrg": "AMAZON-02"
        },
        "Country": {
            "CountryName": "United Kingdom"
        }
    },
    "Api": "ListPolicyVersions",
    "ServiceName": "iam.amazonaws.com",
    "CallerType": "Remote IP"
}
}
},
"Title": "Unusual user permission reconnaissance activity by testuser.",
"Type": "Recon:IAMUser/UserPermissions",
"Region": "us-east-1",
"Partition": "aws",
"Arn": "arn:aws:guardduty:us-east-1:111122223333:detector/
d4b040365221be2b54a6264dcexample/finding/1ab92989eaf0e742df4a014d5example",
"UpdatedAt": "2020-05-26T22:55:21.703Z",
"SchemaVersion": "2.0",
"Severity": 5,
"Id": "1ab92989eaf0e742df4a014d5example",
"CreatedAt": "2020-05-26T22:21:48.385Z",
"AccountId": "111122223333"
}
]
}

```

Untuk informasi selengkapnya, lihat [Temuan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [GetFindings](#) di Referensi AWS CLI Perintah.

## get-ip-set

Contoh kode berikut menunjukkan cara menggunakan `get-ip-set`.

### AWS CLI

Untuk daftar, dapatkan detail pada set IP tepercaya yang ditentukan

`get-ip-set` Contoh berikut menunjukkan status dan detail set IP tepercaya yang ditentukan.

```
aws guardduty get-ip-set \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --ip-set-id d4b94fc952d6912b8f3060768example
```

Output:

```
{  
  "Status": "ACTIVE",  
  "Location": "s3://AWSDOC-EXAMPLE-BUCKET.s3-us-west-2.amazonaws.com/  
customlist.csv",  
  "Tags": {},  
  "Format": "TXT",  
  "Name": "test-ip-set"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Daftar IP Tepercaya dan Daftar Ancaman](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [GetIpSet](#) di Referensi AWS CLI Perintah.

## get-master-account

Contoh kode berikut menunjukkan cara menggunakan `get-master-account`.

### AWS CLI

Untuk mengambil detail tentang akun master Anda di wilayah saat ini

`get-master-account` Contoh berikut menampilkan status dan detail akun master yang terkait dengan detektor Anda di wilayah saat ini.

```
aws guardduty get-master-account \  
  --detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{  
  "Master": {  
    "InvitationId": "04b94d9704854a73f94e061e8example",  
    "InvitedAt": "2020-06-09T22:23:04.970Z",
```

```
    "RelationshipStatus": "Enabled",  
    "AccountId": "123456789111"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memahami Hubungan antara Akun GuardDuty Master dan Anggota](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [GetMasterAccount](#) di Referensi AWS CLI Perintah.

## list-detectors

Contoh kode berikut menunjukkan cara menggunakan `list-detectors`.

### AWS CLI

Untuk daftar detektor yang tersedia di wilayah saat ini

`list-detectors` Contoh berikut mencantumkan detektor yang tersedia di AWS wilayah Anda saat ini.

```
aws guardduty list-detectors
```

Output:

```
{  
  "DetectorIds": [  
    "12abc34d567e8fa901bc2d34eexample"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Konsep dan Terminologi](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [ListDetectors](#) di Referensi AWS CLI Perintah.

## list-findings

Contoh kode berikut menunjukkan cara menggunakan `list-findings`.

### AWS CLI

Contoh 1: Untuk membuat daftar semua temuan untuk wilayah saat ini

`list-findings` Contoh berikut menampilkan daftar semua FindingIds untuk wilayah saat ini diurutkan berdasarkan tingkat keparahan dari tertinggi ke terendah.

```
aws guardduty list-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --sort-criteria '{"AttributeName": "severity", "OrderBy": "DESC"}'
```

Output:

```
{
  "FindingIds": [
    "04b8ab50fd29c64fc771b232dexample",
    "5ab8ab50fd21373735c826d3aexample",
    "90b93de7aba69107f05bbe60bexample",
    ...
  ]
}
```

Untuk informasi selengkapnya, lihat [Temuan](#) di Panduan GuardDuty Pengguna.

Contoh 2: Untuk membuat daftar temuan untuk wilayah saat ini yang cocok dengan kriteria temuan tertentu

`list-findings` Contoh berikut menampilkan daftar semua FindingIds yang cocok dengan jenis temuan tertentu.

```
aws guardduty list-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --finding-criteria '{"Criterion":{"type": {"Eq":["UnauthorizedAccess:EC2/SSHBruteForce"]}}}'
```

Output:

```
{
  "FindingIds": [
    "90b93de7aba69107f05bbe60bexample",
    "6eb9430d7023d30774d6f05e3example",
    "2eb91a2d060ac9a21963a5848example",
    "44b8ab50fd2b0039a9e48f570example",
    "9eb8ab4cd2b7e5b66ba4f5e96example",
    "e0b8ab3a38e9b0312cc390ceeexample"
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Temuan](#) di Panduan GuardDuty Pengguna.

Contoh 3: Untuk membuat daftar temuan untuk wilayah saat ini yang cocok dengan serangkaian kriteria temuan tertentu yang ditentukan dalam file JSON

`list-findings` Contoh berikut menampilkan daftar semua `FindingIds` yang tidak diarsipkan, dan melibatkan pengguna IAM bernama “testuser”, seperti yang ditentukan dalam file JSON.

```
aws guardduty list-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --finding-criteria file://myfile.json
```

Isi dari `myfile.json`:

```
{"Criterion": {
  "resource.accessKeyDetails.userName": {
    "Eq": [
      "testuser"
    ]
  },
  "service.archived": {
    "Eq": [
      "false"
    ]
  }
}
```

Output:

```
{
  "FindingIds": [
    "1ab92989eaf0e742df4a014d5example"
  ]
}
```

Untuk informasi selengkapnya, lihat [Temuan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [ListFindings](#) di Referensi AWS CLI Perintah.

## list-invitations

Contoh kode berikut menunjukkan cara menggunakan `list-invitations`.

### AWS CLI

Untuk mencantumkan detail undangan Anda untuk menjadi akun anggota di wilayah saat ini

`list-invitations` Contoh berikut mencantumkan detail dan status undangan Anda untuk menjadi akun GuardDuty anggota di wilayah saat ini.

```
aws guardduty list-invitations
```

Output:

```
{
  "Invitations": [
    {
      "InvitationId": "d6b94fb03a66ff665f7db8764example",
      "InvitedAt": "2020-06-10T17:56:38.221Z",
      "RelationshipStatus": "Invited",
      "AccountId": "123456789111"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola GuardDuty Akun berdasarkan Undangan](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [ListInvitations](#) di Referensi AWS CLI Perintah.

## list-ip-sets

Contoh kode berikut menunjukkan cara menggunakan `list-ip-sets`.

### AWS CLI

Untuk daftar set IP tepercaya di wilayah saat ini

`list-ip-sets` Contoh berikut mencantumkan set IP tepercaya di AWS wilayah Anda saat ini.

```
aws guardduty list-ip-sets \
```



```
--detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{
  "IpSetIds": [
    "d4b94fc952d6912b8f3060768example"
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Daftar IP Tepercaya dan Daftar Ancaman](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [ListIpSets](#) di Referensi AWS CLI Perintah.

## list-members

Contoh kode berikut menunjukkan cara menggunakan `list-members`.

### AWS CLI

Untuk daftar semua anggota di wilayah saat ini

`list-members` Contoh berikut mencantumkan semua akun anggota dan detailnya untuk wilayah saat ini.

```
aws guardduty list-members \
  --detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{
  "Members": [
    {
      "RelationshipStatus": "Enabled",
      "InvitedAt": "2020-06-09T22:49:00.910Z",
      "MasterId": "123456789111",
      "DetectorId": "7ab8b2f61b256c87f793f6a86example",
      "UpdatedAt": "2020-06-09T23:08:22.512Z",
      "Email": "your+member@example.com",
      "AccountId": "123456789222"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Memahami Hubungan antara Akun GuardDuty Master dan Anggota](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [ListMembers](#) di Referensi AWS CLI Perintah.

## update-ip-set

Contoh kode berikut menunjukkan cara menggunakan `update-ip-set`.

### AWS CLI

Untuk memperbarui set IP tepercaya

`update-ip-set` Contoh berikut menunjukkan cara memperbarui detail set IP tepercaya.

```
aws guardduty update-ip-set \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --ip-set-id d4b94fc952d6912b8f3060768example \
  --location https://AWSDOC-EXAMPLE-BUCKET.s3-us-west-2.amazonaws.com/
customtrustlist2.csv
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Daftar IP Tepercaya dan Daftar Ancaman](#) di Panduan GuardDuty Pengguna.

- Untuk detail API, lihat [UpdateIpSet](#) di Referensi AWS CLI Perintah.

## AWS Health contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Health.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-affected-entities**

Contoh kode berikut menunjukkan cara menggunakan `describe-affected-entities`.

AWS CLI

Untuk membuat daftar entitas yang terpengaruh oleh peristiwa AWS Kesehatan tertentu

`describe-affected-entities` Contoh berikut mencantumkan entitas yang terpengaruh oleh peristiwa AWS Kesehatan yang ditentukan. Acara ini adalah pemberitahuan penagihan untuk AWS akun tersebut.

```
aws health describe-affected-entities \
  --filter "eventArns=arn:aws:health:global::event/BILLING/
  AWS_BILLING_NOTIFICATION/AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-
  EXAMPLE11145" \
  --region us-east-1
```

Output:

```
{
  "entities": [
    {
      "entityArn": "arn:aws:health:global:123456789012:entity/
      EXAMPLEimSMoULmWHpb",
      "eventArn": "arn:aws:health:global::event/BILLING/
      AWS_BILLING_NOTIFICATION/AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-
      EXAMPLE11145",
      "entityValue": "AWS_ACCOUNT",
      "awsAccountId": "123456789012",
      "lastUpdatedTime": 1588356454.08
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Log peristiwa](#) di Panduan Pengguna AWS Kesehatan.

- Untuk detail API, lihat [DescribeAffectedEntities](#) di Referensi AWS CLI Perintah.

## describe-event-details

Contoh kode berikut menunjukkan cara menggunakan `describe-event-details`.

### AWS CLI

Untuk membuat daftar informasi tentang acara AWS Kesehatan

`describe-event-details` Contoh berikut mencantumkan informasi tentang acara AWS Kesehatan yang ditentukan.

```
aws health describe-event-details \
  --event-arns "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111" \
  --region us-east-1
```

Output:

```
{
  "successfulSet": [
    {
      "event": {
        "arn": "arn:aws:health:us-east-1::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
        "eventTypeCategory": "issue",
        "region": "us-east-1",
        "startTime": 1587462325.096,
        "endTime": 1587464204.774,
        "lastUpdatedTime": 1587464204.865,
        "statusCode": "closed"
      },
      "eventDescription": {
        "latestDescription": "[RESOLVED] Increased API Error Rates and
Latencies\n\n[02:45 AM PDT] We are investigating increased API error rates and
latencies in the US-EAST-1 Region.\n\n[03:16 AM PDT] Between 2:10 AM and 2:59 AM
```

```

PDT we experienced increased API error rates and latencies in the US-EAST-1 Region.
The issue has been resolved and the service is operating normally."
    }
  },
  "failedSet": []
}

```

Untuk informasi selengkapnya, lihat [panel Detail acara](#) di Panduan Pengguna AWS Kesehatan.

- Untuk detail API, lihat [DescribeEventDetails](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Contoh 1: Untuk membuat daftar acara AWS Kesehatan

`describe-events` Contoh berikut mencantumkan peristiwa AWS Kesehatan terbaru.

```

aws health describe-events \
  --region us-east-1

```

Output:

```

{
  "events": [
    {
      "arn": "arn:aws:health:us-west-1::event/ECS/AWS_ECS_OPERATIONAL_ISSUE/
AWS_ECS_OPERATIONAL_ISSUE_KWQPY_EXAMPLE111",
      "service": "ECS",
      "eventTypeCode": "AWS_ECS_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "us-west-1",
      "startTime": 1589077890.53,
      "endTime": 1589086345.597,
      "lastUpdatedTime": 1589086345.905,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    },
    {

```

```
    "arn": "arn:aws:health:global::event/BILLING/AWS_BILLING_NOTIFICATION/
AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-EXAMPLE1118b",
    "service": "BILLING",
    "eventTypeCode": "AWS_BILLING_NOTIFICATION",
    "eventTypeCategory": "accountNotification",
    "region": "global",
    "startTime": 1588356000.0,
    "lastUpdatedTime": 1588356524.358,
    "statusCode": "open",
    "eventScopeCode": "ACCOUNT_SPECIFIC"
  },
  {
    "arn": "arn:aws:health:us-west-2::event/
CLOUDFORMATION/AWS_CLOUDFORMATION_OPERATIONAL_ISSUE/
AWS_CLOUDFORMATION_OPERATIONAL_ISSUE_OHTWY_EXAMPLE111",
    "service": "CLOUDFORMATION",
    "eventTypeCode": "AWS_CLOUDFORMATION_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "us-west-2",
    "startTime": 1588279630.761,
    "endTime": 1588284650.0,
    "lastUpdatedTime": 1588284691.941,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:ap-northeast-1::event/LAMBDA/
AWS_LAMBDA_OPERATIONAL_ISSUE/AWS_LAMBDA_OPERATIONAL_ISSUE_JZDND_EXAMPLE111",
    "service": "LAMBDA",
    "eventTypeCode": "AWS_LAMBDA_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "ap-northeast-1",
    "startTime": 1587379534.08,
    "endTime": 1587391771.0,
    "lastUpdatedTime": 1587395689.316,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_COBXJ_EXAMPLE111",
    "service": "EC2",
    "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
```

```
    "region": "us-east-1",
    "startTime": 1586473044.284,
    "endTime": 1586479706.091,
    "lastUpdatedTime": 1586479706.153,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:global::event/SECURITY/AWS_SECURITY_NOTIFICATION/
AWS_SECURITY_NOTIFICATION_42007387-8129-42da-8c88-EXAMPLE11139",
    "service": "SECURITY",
    "eventTypeCode": "AWS_SECURITY_NOTIFICATION",
    "eventTypeCategory": "accountNotification",
    "region": "global",
    "startTime": 1585674000.0,
    "lastUpdatedTime": 1585674004.132,
    "statusCode": "open",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:global::event/CLOUDFRONT/
AWS_CLOUDFRONT_OPERATIONAL_ISSUE/AWS_CLOUDFRONT_OPERATIONAL_ISSUE_FRQXG_EXAMPLE111",
    "service": "CLOUDFRONT",
    "eventTypeCode": "AWS_CLOUDFRONT_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "global",
    "startTime": 1585610898.589,
    "endTime": 1585617671.0,
    "lastUpdatedTime": 1585620638.869,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:us-east-1::event/SES/AWS_SES_OPERATIONAL_ISSUE/
AWS_SES_OPERATIONAL_ISSUE_URNDF_EXAMPLE111",
    "service": "SES",
    "eventTypeCode": "AWS_SES_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "us-east-1",
    "startTime": 1585342008.46,
    "endTime": 1585344017.0,
    "lastUpdatedTime": 1585344355.989,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  }
}
```

```

    },
    {
      "arn": "arn:aws:health:global::event/IAM/
AWS_IAM_OPERATIONAL_NOTIFICATION/
AWS_IAM_OPERATIONAL_NOTIFICATION_b6771c34-6ecd-4aea-9d3e-EXAMPLE1117e",
      "service": "IAM",
      "eventTypeCode": "AWS_IAM_OPERATIONAL_NOTIFICATION",
      "eventTypeCategory": "accountNotification",
      "region": "global",
      "startTime": 1584978300.0,
      "lastUpdatedTime": 1584978553.572,
      "statusCode": "open",
      "eventScopeCode": "ACCOUNT_SPECIFIC"
    },
    {
      "arn": "arn:aws:health:ap-southeast-2::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_HNGHE_EXAMPLE1111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "ap-southeast-2",
      "startTime": 1583881487.483,
      "endTime": 1583885056.785,
      "lastUpdatedTime": 1583885057.052,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Memulai Dashboard AWS Personal AWS Health](#) di Panduan Pengguna Kesehatan.

Contoh 2: Untuk daftar acara AWS Kesehatan berdasarkan layanan dan kode status acara

`describe-events` Contoh berikut mencantumkan peristiwa AWS Kesehatan untuk Amazon Elastic Compute Cloud (Amazon EC2) di mana status acara ditutup.

```

aws health describe-events \
  --filter "services=EC2,eventStatusCodes=closed"

```

Output:



```
{
  "events": [
    {
      "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "us-east-1",
      "startTime": 1587462325.096,
      "endTime": 1587464204.774,
      "lastUpdatedTime": 1587464204.865,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    },
    {
      "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_COBXJ_EXAMPLE111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "us-east-1",
      "startTime": 1586473044.284,
      "endTime": 1586479706.091,
      "lastUpdatedTime": 1586479706.153,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    },
    {
      "arn": "arn:aws:health:ap-southeast-2::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_HNGHE_EXAMPLE111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "ap-southeast-2",
      "startTime": 1583881487.483,
      "endTime": 1583885056.785,
      "lastUpdatedTime": 1583885057.052,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Dashboard AWS Personal AWS Health](#) di Panduan Pengguna Kesehatan.

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## HealthImaging contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with HealthImaging.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **copy-image-set**

Contoh kode berikut menunjukkan cara menggunakan `copy-image-set`.

AWS CLI

Contoh 1: Untuk menyalin set gambar tanpa tujuan.

Contoh `copy-image-set` kode berikut membuat salinan duplikat dari gambar yang ditetapkan tanpa tujuan.

```
aws medical-imaging copy-image-set \  
  --datastore-id 12345678901234567890123456789012 \  
  --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \  
  --copy-image-set-information '{"sourceImageSet": {"latestVersionId": "1" } }'
```

**Output:**

```
{
  "destinationImageSetProperties": {
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "COPYING",
    "updatedAt": 1680042357.432,
    "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
    "imageSetState": "LOCKED",
    "createdAt": 1680042357.432
  },
  "sourceImageSetProperties": {
    "latestVersionId": "1",
    "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
    "updatedAt": 1680042357.432,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436
  },
  "datastoreId": "12345678901234567890123456789012"
}
```

Contoh 2: Untuk menyalin gambar yang ditetapkan dengan tujuan.

Contoh `copy-image-set` kode berikut membuat salinan duplikat dari gambar yang ditetapkan dengan tujuan.

```
aws medical-imaging copy-image-set \
  --datastore-id 12345678901234567890123456789012 \
  --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --copy-image-set-information '{"sourceImageSet": {"latestVersionId": "1" },
"destinationImageSet": { "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
"latestVersionId": "1"} }'
```

**Output:**

```
{
  "destinationImageSetProperties": {
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "COPYING",
    "updatedAt": 1680042505.135,
    "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
    "imageSetState": "LOCKED",
```

```
    "createdAt": 1680042357.432
  },
  "sourceImageSetProperties": {
    "latestVersionId": "1",
    "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
    "updatedAt": 1680042505.135,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436
  },
  "datastoreId": "12345678901234567890123456789012"
}
```

Untuk informasi selengkapnya, lihat [Menyalin set gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [CopyImageSet](#) di Referensi AWS CLI Perintah.

## create-datastore

Contoh kode berikut menunjukkan cara menggunakan `create-datastore`.

### AWS CLI

Untuk membuat penyimpanan data

Contoh `create-datastore` kode berikut membuat penyimpanan data dengan nama `my-datastore`.

```
aws medical-imaging create-datastore \
  --datastore-name "my-datastore"
```

Output:

```
{
  "datastoreId": "12345678901234567890123456789012",
  "datastoreStatus": "CREATING"
}
```

Untuk informasi selengkapnya, lihat [Membuat penyimpanan data](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [CreateDatastore](#) di Referensi AWS CLI Perintah.

## delete-datastore

Contoh kode berikut menunjukkan cara menggunakan `delete-datastore`.

### AWS CLI

Untuk menghapus penyimpanan data

Contoh `delete-datastore` kode berikut menghapus penyimpanan data.

```
aws medical-imaging delete-datastore \  
  --datastore-id "12345678901234567890123456789012"
```

Output:

```
{  
  "datastoreId": "12345678901234567890123456789012",  
  "datastoreStatus": "DELETING"  
}
```

Untuk informasi selengkapnya, lihat [Menghapus penyimpanan data](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [DeleteDatastore](#) di Referensi AWS CLI Perintah.

## delete-image-set

Contoh kode berikut menunjukkan cara menggunakan `delete-image-set`.

### AWS CLI

Untuk menghapus kumpulan gambar

Contoh `delete-image-set` kode berikut menghapus set gambar.

```
aws medical-imaging delete-image-set \  
  --datastore-id 12345678901234567890123456789012 \  
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e
```

Output:

```
{
```

```
"imageSetWorkflowStatus": "DELETING",
"imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
"imageSetState": "LOCKED",
"datastoreId": "12345678901234567890123456789012"
}
```

Untuk informasi selengkapnya, lihat [Menghapus set gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [DeleteImageSet](#) di Referensi AWS CLI Perintah.

## get-datastore

Contoh kode berikut menunjukkan cara menggunakan `get-datastore`.

### AWS CLI

Untuk mendapatkan properti penyimpanan data

Contoh `get-datastore` kode berikut mendapatkan properti penyimpanan data ini.

```
aws medical-imaging get-datastore \
  --datastore-id 12345678901234567890123456789012
```

Output:

```
{
  "datastoreProperties": {
    "datastoreId": "12345678901234567890123456789012",
    "datastoreName": "TestDatastore123",
    "datastoreStatus": "ACTIVE",
    "datastoreArn": "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012",
    "createdAt": "2022-11-15T23:33:09.643000+00:00",
    "updatedAt": "2022-11-15T23:33:09.643000+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan properti penyimpanan data](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [GetDatastore](#) di Referensi AWS CLI Perintah.

## get-dicom-import-job

Contoh kode berikut menunjukkan cara menggunakan `get-dicom-import-job`.

### AWS CLI

Untuk mendapatkan properti pekerjaan impor dicom

Contoh `get-dicom-import-job` kode berikut mendapatkan properti pekerjaan dicom import.

```
aws medical-imaging get-dicom-import-job \  
  --datastore-id "12345678901234567890123456789012" \  
  --job-id "09876543210987654321098765432109"
```

Output:

```
{  
  "jobProperties": {  
    "jobId": "09876543210987654321098765432109",  
    "jobName": "my-job",  
    "jobStatus": "COMPLETED",  
    "datastoreId": "12345678901234567890123456789012",  
    "dataAccessRoleArn": "arn:aws:iam::123456789012:role/  
ImportJobDataAccessRole",  
    "endedAt": "2022-08-12T11:29:42.285000+00:00",  
    "submittedAt": "2022-08-12T11:28:11.152000+00:00",  
    "inputS3Uri": "s3://medical-imaging-dicom-input/dicom_input/",  
    "outputS3Uri": "s3://medical-imaging-output/  
job_output/12345678901234567890123456789012-  
DicomImport-09876543210987654321098765432109/"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan properti pekerjaan impor](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [GetDicom ImportJob](#) di Referensi AWS CLI Perintah.

## get-image-frame

Contoh kode berikut menunjukkan cara menggunakan `get-image-frame`.

## AWS CLI

Untuk mendapatkan data piksel set gambar

Contoh `get-image-frame` kode berikut mendapat bingkai gambar.

```
aws medical-imaging get-image-frame \  
  --datastore-id "12345678901234567890123456789012" \  
  --image-set-id "98765412345612345678907890789012" \  
  --image-frame-information imageFrameId=3abf5d5d7ae72f80a0ec81b2c0de3ef4 \  
  imageframe.jpg
```

Catatan: Contoh kode ini tidak menyertakan output karena `GetImageFrame` tindakan mengembalikan aliran data piksel ke file `imageframe.jpg`. Untuk informasi tentang decoding dan melihat bingkai gambar, lihat [HTJ2K decoding library](#).

Untuk informasi selengkapnya, lihat [Mendapatkan data piksel yang disetel gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [GetImageFrame](#) di Referensi AWS CLI Perintah.

## `get-image-set-metadata`

Contoh kode berikut menunjukkan cara menggunakan `get-image-set-metadata`.

## AWS CLI

Contoh 1: Untuk mendapatkan metadata set gambar tanpa versi

Contoh `get-image-set-metadata` kode berikut mendapatkan metadata untuk kumpulan gambar tanpa menentukan versi.

Catatan: `outfile` adalah parameter yang diperlukan

```
aws medical-imaging get-image-set-metadata \  
  --datastore-id 12345678901234567890123456789012 \  
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \  
  studymetadata.json.gz
```

Metadata yang dikembalikan dikompresi dengan gzip dan disimpan dalam file `studymetadata.json.gz`. Untuk melihat isi objek JSON yang dikembalikan, Anda harus terlebih dahulu mendekompresnya.



**Output:**

```
{
  "contentType": "application/json",
  "contentEncoding": "gzip"
}
```

Contoh 2: Untuk mendapatkan metadata set gambar dengan versi

Contoh `get-image-set-metadata` kode berikut mendapatkan metadata untuk set gambar dengan versi tertentu.

Catatan: `outfile` adalah parameter yang diperlukan

```
aws medical-imaging get-image-set-metadata \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --version-id 1 \
  studymetadata.json.gz
```

Metadata yang dikembalikan dikompresi dengan gzip dan disimpan dalam file `studymetadata.json.gz`. Untuk melihat isi objek JSON yang dikembalikan, Anda harus terlebih dahulu mendekompresnya.

**Output:**

```
{
  "contentType": "application/json",
  "contentEncoding": "gzip"
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan metadata set gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [GetImageSetMetadata](#) di Referensi AWS CLI Perintah.

**get-image-set**

Contoh kode berikut menunjukkan cara menggunakan `get-image-set`.

## AWS CLI

Untuk mendapatkan properti set gambar

Contoh `get-image-set` kode berikut mendapatkan properti untuk set gambar.

```
aws medical-imaging get-image-set \  
  --datastore-id 12345678901234567890123456789012 \  
  --image-set-id 18f88ac7870584f58d56256646b4d92b \  
  --version-id 1
```

Output:

```
{  
  "versionId": "1",  
  "imageSetWorkflowStatus": "COPIED",  
  "updatedAt": 1680027253.471,  
  "imageSetId": "18f88ac7870584f58d56256646b4d92b",  
  "imageSetState": "ACTIVE",  
  "createdAt": 1679592510.753,  
  "datastoreId": "12345678901234567890123456789012"  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan properti set gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [GetImageSet](#) di Referensi AWS CLI Perintah.

## list-datastores

Contoh kode berikut menunjukkan cara menggunakan `list-datastores`.

### AWS CLI

Untuk daftar penyimpanan data

Contoh `list-datastores` kode berikut mencantumkan penyimpanan data yang tersedia.

```
aws medical-imaging list-datastores
```

Output:

```
{
  "datastoreSummaries": [
    {
      "datastoreId": "12345678901234567890123456789012",
      "datastoreName": "TestDatastore123",
      "datastoreStatus": "ACTIVE",
      "datastoreArn": "arn:aws:medical-imaging:us-east-1:123456789012:datastore/12345678901234567890123456789012",
      "createdAt": "2022-11-15T23:33:09.643000+00:00",
      "updatedAt": "2022-11-15T23:33:09.643000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyimpan penyimpanan data](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [ListDatastores](#) di Referensi AWS CLI Perintah.

## list-dicom-import-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-dicom-import-jobs`.

### AWS CLI

Untuk daftar pekerjaan dicom import

Contoh `list-dicom-import-jobs` kode berikut mencantumkan pekerjaan impor dicom.

```
aws medical-imaging list-dicom-import-jobs \
  --datastore-id "12345678901234567890123456789012"
```

Output:

```
{
  "jobSummaries": [
    {
      "jobId": "09876543210987654321098765432109",
      "jobName": "my-job",
      "jobStatus": "COMPLETED",
      "datastoreId": "12345678901234567890123456789012",
    }
  ]
}
```

```
        "dataAccessRoleArn": "arn:aws:iam::123456789012:role/
ImportJobDataAccessRole",
        "endedAt": "2022-08-12T11:21:56.504000+00:00",
        "submittedAt": "2022-08-12T11:20:21.734000+00:00"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Daftar lowongan impor](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [ListDicom ImportJobs](#) di Referensi AWS CLI Perintah.

## list-image-set-versions

Contoh kode berikut menunjukkan cara menggunakan `list-image-set-versions`.

### AWS CLI

Untuk daftar versi set gambar

Contoh `list-image-set-versions` kode berikut mencantumkan riwayat versi untuk kumpulan gambar.

```
aws medical-imaging list-image-set-versions \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e
```

Output:

```
{
  "imageSetPropertiesList": [
    {
      "ImageSetWorkflowStatus": "UPDATED",
      "versionId": "4",
      "updatedAt": 1680029436.304,
      "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
      "imageSetState": "ACTIVE",
      "createdAt": 1680027126.436
    },
    {
      "ImageSetWorkflowStatus": "UPDATED",
      "versionId": "3",
```

```

        "updatedAt": 1680029163.325,
        "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
        "imageSetState": "ACTIVE",
        "createdAt": 1680027126.436
    },
    {
        "ImageSetWorkflowStatus": "COPY_FAILED",
        "versionId": "2",
        "updatedAt": 1680027455.944,
        "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
        "imageSetState": "ACTIVE",
        "message": "INVALID_REQUEST: Series of SourceImageSet and
DestinationImageSet don't match.",
        "createdAt": 1680027126.436
    },
    {
        "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
        "imageSetState": "ACTIVE",
        "versionId": "1",
        "ImageSetWorkflowStatus": "COPIED",
        "createdAt": 1680027126.436
    }
]
}

```

Untuk informasi selengkapnya, lihat [Daftar versi kumpulan gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [ListImageSetVersions](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Contoh 1: Untuk daftar tag sumber daya untuk penyimpanan data

Contoh `list-tags-for-resource` kode berikut mencantumkan tag untuk penyimpanan data.

```

aws medical-imaging list-tags-for-resource \
  --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012"

```

**Output:**

```
{
  "tags":{
    "Deployment":"Development"
  }
}
```

Contoh 2: Untuk mencantumkan tag sumber daya untuk kumpulan gambar

Contoh `list-tags-for-resource` kode berikut mencantumkan tag untuk kumpulan gambar.

```
aws medical-imaging list-tags-for-resource \
  --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012/
imageset/18f88ac7870584f58d56256646b4d92b"
```

**Output:**

```
{
  "tags":{
    "Deployment":"Development"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS HealthImaging](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

**search-image-sets**

Contoh kode berikut menunjukkan cara menggunakan `search-image-sets`.

**AWS CLI**

Contoh 1: Untuk mencari set gambar dengan operator EQUAL

Contoh `search-image-sets` kode berikut menggunakan operator EQUAL untuk mencari set gambar berdasarkan nilai tertentu.

```
aws medical-imaging search-image-sets \  
  --datastore-id 12345678901234567890123456789012 \  
  --search-criteria file://search-criteria.json
```

Isi dari `search-criteria.json`

```
{  
  "filters": [{  
    "values": [{"DICOMPatientId" : "SUBJECT08701"}],  
    "operator": "EQUAL"  
  }]  
}
```

Output:

```
{  
  "imageSetsMetadataSummaries": [{  
    "imageSetId": "09876543210987654321098765432109",  
    "createdAt": "2022-12-06T21:40:59.429000+00:00",  
    "version": 1,  
    "DICOMTags": {  
      "DICOMStudyId": "2011201407",  
      "DICOMStudyDate": "19991122",  
      "DICOMPatientSex": "F",  
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",  
      "DICOMPatientBirthDate": "19201120",  
      "DICOMStudyDescription": "UNKNOWN",  
      "DICOMPatientId": "SUBJECT08701",  
      "DICOMPatientName": "Melissa844 Huel628",  
      "DICOMNumberOfStudyRelatedInstances": 1,  
      "DICOMStudyTime": "140728",  
      "DICOMNumberOfStudyRelatedSeries": 1  
    },  
    "updatedAt": "2022-12-06T21:40:59.429000+00:00"  
  }]  
}
```

Contoh 2: Untuk mencari set gambar dengan operator BETWEEN menggunakan DICOM StudyDate dan DICOM StudyTime

Contoh `search-image-sets` kode berikut mencari kumpulan gambar dengan Studi DICOM yang dihasilkan antara 1 Januari 1990 (12:00 AM) dan 1 Januari 2023 (12:00 AM).

Catatan: DICOM StudyTime adalah opsional. Jika tidak ada, 12:00 AM (awal hari) adalah nilai waktu untuk tanggal yang disediakan untuk penyaringan.

```
aws medical-imaging search-image-sets \  
  --datastore-id 12345678901234567890123456789012 \  
  --search-criteria file://search-criteria.json
```

Isi dari `search-criteria.json`

```
{  
  "filters": [{  
    "values": [{  
      "DICOMStudyDateAndTime": {  
        "DICOMStudyDate": "19900101",  
        "DICOMStudyTime": "000000"  
      }  
    },  
    {  
      "DICOMStudyDateAndTime": {  
        "DICOMStudyDate": "20230101",  
        "DICOMStudyTime": "000000"  
      }  
    }  
  ]},  
  "operator": "BETWEEN"  
}]  
}
```

Output:

```
{  
  "imageSetsMetadataSummaries": [{  
    "imageSetId": "09876543210987654321098765432109",  
    "createdAt": "2022-12-06T21:40:59.429000+00:00",  
    "version": 1,  
    "DICOMTags": {  
      "DICOMStudyId": "2011201407",  
      "DICOMStudyDate": "19991122",  
      "DICOMPatientSex": "F",  
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",  
      "DICOMPatientBirthDate": "19201120",  
      "DICOMStudyDescription": "UNKNOWN",  
      "DICOMPatientId": "SUBJECT08701",  
    }  
  }  
}
```



```

        "DICOMPatientName": "Melissa844 Huel628",
        "DICOMNumberOfStudyRelatedInstances": 1,
        "DICOMStudyTime": "140728",
        "DICOMNumberOfStudyRelatedSeries": 1
    },
    "updatedAt": "2022-12-06T21:40:59.429000+00:00"
  ]
}

```

Contoh 3: Untuk mencari set gambar dengan operator BETWEEN menggunakan createDat (studi waktu sebelumnya dipertahankan)

Contoh search-image-sets kode berikut mencari set gambar dengan Studi DICOM bertahan di HealthImaging antara rentang waktu di zona waktu UTC.

Catatan: Berikan CreateDat dalam format contoh ("1985-04-12T 23:20:50.52 Z").

```

aws medical-imaging search-image-sets \
  --datastore-id 12345678901234567890123456789012 \
  --search-criteria file://search-criteria.json

```

Isi dari search-criteria.json

```

{
  "filters": [{
    "values": [{
      "createdAt": "1985-04-12T23:20:50.52Z"
    },
    {
      "createdAt": "2022-04-12T23:20:50.52Z"
    }
  ]],
  "operator": "BETWEEN"
}

```

Output:

```

{
  "imageSetsMetadataSummaries": [{
    "imageSetId": "09876543210987654321098765432109",
    "createdAt": "2022-12-06T21:40:59.429000+00:00",

```

```

    "version": 1,
    "DICOMTags": {
      "DICOMStudyId": "2011201407",
      "DICOMStudyDate": "19991122",
      "DICOMPatientSex": "F",
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
      "DICOMPatientBirthDate": "19201120",
      "DICOMStudyDescription": "UNKNOWN",
      "DICOMPatientId": "SUBJECT08701",
      "DICOMPatientName": "Melissa844 Huel628",
      "DICOMNumberOfStudyRelatedInstances": 1,
      "DICOMStudyTime": "140728",
      "DICOMNumberOfStudyRelatedSeries": 1
    },
    "lastUpdatedAt": "2022-12-06T21:40:59.429000+00:00"
  ]
}

```

Contoh 4: Untuk mencari set gambar dengan operator EQUAL pada DICOM SeriesInstance UID dan ANTARA di UpdateDat dan mengurutkan respons dalam urutan ASC di bidang UpdateDAT

Contoh search-image-sets kode berikut mencari kumpulan gambar dengan operator EQUAL pada DICOM SeriesInstance UID dan BETWIDE pada UpdateDat dan mengurutkan respons dalam urutan ASC pada bidang UpdateDAT.

Catatan: Berikan UpdateDat dalam format contoh ("1985-04-12T 23:20:50.52 Z").

```

aws medical-imaging search-image-sets \
  --datastore-id 12345678901234567890123456789012 \
  --search-criteria file://search-criteria.json

```

Isi dari search-criteria.json

```

{
  "filters": [{
    "values": [{
      "updatedAt": "2024-03-11T15:00:05.074000-07:00"
    }, {
      "updatedAt": "2024-03-11T16:00:05.074000-07:00"
    }],
    "operator": "BETWEEN"
  }, {

```

```

    "values": [{
      "DICOMSeriesInstanceUID": "1.2.840.99999999.84710745.943275268089"
    }],
    "operator": "EQUAL"
  }],
  "sort": {
    "sortField": "updatedAt",
    "sortOrder": "ASC"
  }
}

```

### Output:

```

{
  "imageSetsMetadataSummaries": [{
    "imageSetId": "09876543210987654321098765432109",
    "createdAt": "2022-12-06T21:40:59.429000+00:00",
    "version": 1,
    "DICOMTags": {
      "DICOMStudyId": "2011201407",
      "DICOMStudyDate": "19991122",
      "DICOMPatientSex": "F",
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
      "DICOMPatientBirthDate": "19201120",
      "DICOMStudyDescription": "UNKNOWN",
      "DICOMPatientId": "SUBJECT08701",
      "DICOMPatientName": "Melissa844 Huel628",
      "DICOMNumberOfStudyRelatedInstances": 1,
      "DICOMStudyTime": "140728",
      "DICOMNumberOfStudyRelatedSeries": 1
    },
    "lastUpdatedAt": "2022-12-06T21:40:59.429000+00:00"
  }]
}

```

Untuk informasi selengkapnya, lihat [Mencari kumpulan gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [SearchImageSets](#) di Referensi AWS CLI Perintah.

## start-dicom-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-dicom-import-job`.

## AWS CLI

Untuk memulai pekerjaan impor dicom

Contoh `start-dicom-import-job` kode berikut memulai pekerjaan impor dicom.

```
aws medical-imaging start-dicom-import-job \  
  --job-name "my-job" \  
  --datastore-id "12345678901234567890123456789012" \  
  --input-s3-uri "s3://medical-imaging-dicom-input/dicom_input/" \  
  --output-s3-uri "s3://medical-imaging-output/job_output/" \  
  --data-access-role-arn "arn:aws:iam::123456789012:role/ImportJobDataAccessRole"
```

Output:

```
{  
  "datastoreId": "12345678901234567890123456789012",  
  "jobId": "09876543210987654321098765432109",  
  "jobStatus": "SUBMITTED",  
  "submittedAt": "2022-08-12T11:28:11.152000+00:00"  
}
```

Untuk informasi selengkapnya, lihat [Memulai pekerjaan impor](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [StartDicom ImportJob](#) di AWS CLI Referensi Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Contoh 1: Untuk menandai penyimpanan data

Contoh `tag-resource` kode berikut menandai penyimpanan data.

```
aws medical-imaging tag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012" \  
  --tags '{"Deployment":"Development}"'
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menandai set gambar

Contoh `tag-resource` kode berikut menandai set gambar.

```
aws medical-imaging tag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012/  
imageset/18f88ac7870584f58d56256646b4d92b" \  
  --tags '{"Deployment":"Development"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS HealthImaging](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Contoh 1: Untuk menghapus tag penyimpanan data

Contoh `untag-resource` kode berikut untags penyimpanan data.

```
aws medical-imaging untag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012" \  
  --tag-keys ["Deployment"]'
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menghapus tag set gambar

Contoh `untag-resource` kode berikut untag set gambar.

```
aws medical-imaging untag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012/  
imageset/18f88ac7870584f58d56256646b4d92b" \  
  --tag-keys ["Deployment"]'
```

```
--resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012/
imageset/18f88ac7870584f58d56256646b4d92b" \
--tag-keys '["Deployment"]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya AWS HealthImaging](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-image-set-metadata

Contoh kode berikut menunjukkan cara menggunakan `update-image-set-metadata`.

### AWS CLI

Untuk menyisipkan atau memperbarui atribut dalam metadata set gambar

Contoh `update-image-set-metadata` kode berikut menyisipkan atau memperbarui atribut dalam metadata set gambar.

```
aws medical-imaging update-image-set-metadata \
--datastore-id 12345678901234567890123456789012 \
--image-set-id ea92b0d8838c72a3f25d00d13616f87e \
--latest-version-id 1 \
--update-image-set-metadata-updates file://metadata-updates.json
```

Isi dari `metadata-updates.json`

```
{
  "DICOMUpdates": {
    "updateableAttributes":
      "eyJTY2h1bWFWZXJzaW9uIjoxLjEsIlBhdGllbnQiOnsiRElDT00iOnsiUGF0aWVudE5hbWUiOiJNWf5NWCJ9fX0="
  }
}
```

Catatan: `updateableAttributes` adalah string JSON yang dikodekan Base64. Berikut adalah string JSON yang tidak dikodekan.

```
{"SchemaVersion": "1.1", "Pasien": {"DICOM": {"PatientName": ".:MX^MX"}}
```

**Output:**

```
{
  "latestVersionId": "2",
  "imageSetWorkflowStatus": "UPDATING",
  "updatedAt": 1680042257.908,
  "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
  "imageSetState": "LOCKED",
  "createdAt": 1680027126.436,
  "datastoreId": "12345678901234567890123456789012"
}
```

Untuk menghapus atribut dari metadata set gambar

Contoh `update-image-set-metadata` kode berikut menghapus atribut dari metadata set gambar.

```
aws medical-imaging update-image-set-metadata \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --latest-version-id 1 \
  --update-image-set-metadata-updates file://metadata-updates.json
```

Isi dari `metadata-updates.json`

```
{
  "DICOMUpdates": {
    "removableAttributes":
    "e1NjaGVtYVZlcnNpb246MS4xLFN0dWR50ntESUNPTTp7U3R1ZH1EZSNjcm1wdG1vbjpdSEVTVH19fQo="
  }
}
```

Catatan: `removableAttributes` adalah string JSON yang dikodekan Base64. Berikut adalah string JSON yang tidak dikodekan. Kunci dan nilai harus sesuai dengan atribut yang akan dihapus.

```
{ "SchemaVersion": "1.1", "Belajar": { "DICOM": { "StudyDescription": "DADA" } } }
```

**Output:**

```
{
```

```

    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
  }

```

Untuk menghapus instance dari metadata set gambar

Contoh update-image-set-metadata kode berikut menghapus instance dari metadata set gambar.

```

aws medical-imaging update-image-set-metadata \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --latest-version-id 1 \
  --update-image-set-metadata-updates file://metadata-updates.json

```

Isi dari metadata-updates.json

```

{
  "DICOMUpdates": {
    "removableAttributes":
    "eezEuMS4xLjEuMS4xLjEyMzQ1LjEyMzQ1Njc4OTAxMi4xMjMuMTIzNDU2Nzg5MDEyMzQuMTp7SW5zdGFuY2Vz0nsxL
  }
}

```

Catatan: removableAttributes adalah string JSON yang dikodekan Base64. Berikut adalah string JSON yang tidak dikodekan.

```

{"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1": {"Contoh":
{"1.1.1.1.1.1.1.12345.123456789012.123.12345678901234.1": {}}}}

```

Output:

```

{
  "latestVersionId": "2",
  "imageSetWorkflowStatus": "UPDATING",
  "updatedAt": 1680042257.908,

```



```
"imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
"imageSetState": "LOCKED",
"createdAt": 1680027126.436,
"datastoreId": "12345678901234567890123456789012"
}
```

Untuk informasi selengkapnya, lihat [Memperbarui metadata set gambar](#) di Panduan AWS HealthImaging Pengembang.

- Untuk detail API, lihat [UpdateImageSetMetadada](#) di Referensi AWS CLI Perintah.

## HealthLake contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with HealthLake.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-fhir-datastore**

Contoh kode berikut menunjukkan cara menggunakan `create-fhir-datastore`.

AWS CLI

Untuk membuat FHIR Data Store.

`create-fhir-datastore` Contoh berikut menunjukkan cara membuat Data Store baru di Amazon HealthLake.

```
aws healthlake create-fhir-datastore \  
  --region us-east-1 \  
  --datastore-type-version R4 \  
  --datastore-type-version R4 \  
  --datastore-name "FhirTestDatastore"
```

Output:

```
{  
  "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/  
(Datastore ID)/r4/",  
  "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/  
(Datastore ID)",  
  "DatastoreStatus": "CREATING",  
  "DatastoreId": "(Datastore ID)"  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan memantau Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [CreateFhirDatastore](#) di Referensi AWS CLI Perintah.

## delete-fhir-datastore

Contoh kode berikut menunjukkan cara menggunakan `delete-fhir-datastore`.

### AWS CLI

Untuk menghapus FHIR Data Store

`delete-fhir-datastore` Contoh berikut menunjukkan cara menghapus Data Store dan semua isinya di Amazon HealthLake.

```
aws healthlake delete-fhir-datastore \  
  --datastore-id (Data Store ID) \  
  --region us-east-1
```

Output:

```
{
```

```

    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
(Datastore ID)/r4/",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/
(Datastore ID)",
    "DatastoreStatus": "DELETING",
    "DatastoreId": "(Datastore ID)"
}

```

Untuk informasi selengkapnya, lihat [Membuat dan memantau Penyimpanan Data FHIR](https://docs.aws.amazon.com/healthlake/latest/devguide/working-with-FHIR-healthlake.html) < <https://docs.aws.amazon.com/healthlake/latest/devguide/working-with-FHIR-healthlake.html> > di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [DeleteFhirDatastore](#) di Referensi AWS CLI Perintah.

## describe-fhir-datastore

Contoh kode berikut menunjukkan cara menggunakan `describe-fhir-datastore`.

### AWS CLI

Untuk menggambarkan Toko Data FHIR

`describe-fhir-datastore` Contoh berikut menunjukkan bagaimana menemukan properti dari Data Store di Amazon HealthLake.

```

aws healthlake describe-fhir-datastore \
  --datastore-id "1f2f459836ac6c513ce899f9e4f66a59" \
  --region us-east-1

```

Output:

```

{
  "DatastoreProperties": {
    "PreloadDataConfig": {
      "PreloadDataType": "SYNTHEA"
    },
    "DatastoreName": "FhirTestDatastore",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/
(Datastore ID)",
    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
(Datastore ID)/r4/",

```

```

    "DatastoreId": "(Datastore ID)"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat dan memantau Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [DescribeFhirDatastore](#) di Referensi AWS CLI Perintah.

## describe-fhir-export-job

Contoh kode berikut menunjukkan cara menggunakan `describe-fhir-export-job`.

### AWS CLI

Untuk menggambarkan pekerjaan ekspor FHIR

`describe-fhir-export-job` Contoh berikut menunjukkan cara menemukan properti pekerjaan ekspor FHIR di Amazon HealthLake.

```

aws healthlake describe-fhir-export-job \
  --datastore-id (Datastore ID) \
  --job-id 9b9a51943afaedd0a8c0c26c49135a31

```

Output:

```

{
  "ExportJobProperties": {
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "IN_PROGRESS",
    "JobId": "9009813e9d69ba7cf79bcb3468780f16",
    "SubmitTime": 1609175692.715,
    "OutputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix
Name)/59593b2d0367ce252b5e66bf5fd6b574-
FHIR_EXPORT-9009813e9d69ba7cf79bcb3468780f16/"
    },
    "DatastoreId": "(Datastore ID)"
  }
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Mengekspor file dari Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [DescribeFhirExportJob](#) di Referensi AWS CLI Perintah.

## describe-fhir-import-job

Contoh kode berikut menunjukkan cara menggunakan `describe-fhir-import-job`.

### AWS CLI

Untuk menggambarkan pekerjaan impor FHIR

`describe-fhir-import-job` Contoh berikut menunjukkan cara mempelajari properti pekerjaan impor FHIR menggunakan Amazon HealthLake.

```
aws healthlake describe-fhir-import-job \  
  --datastore-id (Datastore ID) \  
  --job-id c145fbb27b192af392f8ce6e7838e34f \  
  --region us-east-1
```

### Output:

```
{  
  "ImportJobProperties": {  
    "InputDataConfig": {  
      "S3Uri": "s3://(Bucket Name)/(Prefix Name)/"  
      { "arrayitem2": 2 }  
    },  
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",  
    "JobStatus": "COMPLETED",  
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",  
    "SubmitTime": 1606272542.161,  
    "EndTime": 1606272609.497,  
    "DatastoreId": "(Datastore ID)"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengimpor file ke Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [DescribeFhirImportJob](#) di Referensi AWS CLI Perintah.

## list-fhir-datastores

Contoh kode berikut menunjukkan cara menggunakan `list-fhir-datastores`.

### AWS CLI

Untuk daftar Toko Data FHIR

`list-fhir-datastores` Contoh berikut menunjukkan cara menggunakan perintah dan bagaimana pengguna dapat memfilter hasil berdasarkan status Penyimpanan Data di Amazon HealthLake.

```
aws healthlake list-fhir-datastores \
  --region us-east-1 \
  --filter DatastoreStatus=ACTIVE
```

Output:

```
{
  "DatastorePropertiesList": [
    {
      "PreloadDataConfig": {
        "PreloadDataType": "SYNTHEA"
      },
      "DatastoreName": "FhirTestDatastore",
      "DatastoreArn": "arn:aws:healthlake:us-east-1:<AWS Account ID>:datastore/
<Datastore ID>",
      "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
<Datastore ID>/r4/",
      "DatastoreStatus": "ACTIVE",
      "DatastoreTypeVersion": "R4",
      "CreatedAt": 1605574003.209,
      "DatastoreId": "<Datastore ID>"
    },
    {
      "DatastoreName": "Demo",
      "DatastoreArn": "arn:aws:healthlake:us-east-1:<AWS Account ID>:datastore/
<Datastore ID>",
      "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
<Datastore ID>/r4/",
```

```

    "DatastoreStatus": "ACTIVE",
    "DatastoreTypeVersion": "R4",
    "CreatedAt": 1603761064.881,
    "DatastoreId": "<Datastore ID>"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat dan memantau Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [ListFhirDatastores](#) di Referensi AWS CLI Perintah.

## list-fhir-export-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-fhir-export-jobs`.

### AWS CLI

Untuk mencantumkan semua pekerjaan ekspor FHIR

`list-fhir-export-jobs` Contoh berikut menunjukkan cara menggunakan perintah untuk melihat daftar pekerjaan ekspor yang terkait dengan akun.

```

aws healthlake list-fhir-export-jobs \
  --datastore-id (Datastore ID) \
  --submitted-before (DATE like 2024-10-13T19:00:00Z)\
  --submitted-after (DATE like 2020-10-13T19:00:00Z) \
  --job-name "FHIR-EXPORT" \
  --job-status SUBMITTED \
  --max-results (Integer between 1 and 500)

```

### Output:

```

{
  "ExportJobProperties": {
    "OutputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix Name)/"
      "S3Configuration": {
        "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
        "KmsKeyId" : "(KmsKey Id)"
      },

```

```

    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "JobName": "FHIR-EXPORT",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Datastore ID)"
  }
}
"NextToken": String

```

Untuk informasi selengkapnya, lihat [Mengekspor file dari Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [ListFhirExportJobs](#) di Referensi AWS CLI Perintah.

## list-fhir-import-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-fhir-import-jobs`.

### AWS CLI

Untuk mencantumkan semua pekerjaan impor FHIR

`list-fhir-import-jobs` Contoh berikut menunjukkan cara menggunakan perintah untuk melihat daftar semua pekerjaan impor yang terkait dengan akun.

```

aws healthlake list-fhir-import-jobs \
  --datastore-id (Datastore ID) \
  --submitted-before (DATE like 2024-10-13T19:00:00Z) \
  --submitted-after (DATE like 2020-10-13T19:00:00Z ) \
  --job-name "FHIR-IMPORT" \
  --job-status SUBMITTED \
  -max-results (Integer between 1 and 500)

```

### Output:

```

{
  "ImportJobProperties": {
    "OutputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",

```



```

        "S3Configuration": {
            "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
            "KmsKeyId" : "(KmsKey Id)"
        },
    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "JobName" "FHIR-IMPORT",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Datastore ID)"
}
}
"NextToken": String

```

Untuk informasi selengkapnya, lihat [Mengimpor file ke FHIR Data Store](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [ListFhirImportJobs](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag untuk Penyimpanan Data

`list-tags-for-resource` Contoh berikut mencantumkan tag yang terkait dengan Data Store yang ditentukan. :

```

aws healthlake list-tags-for-resource \
  --resource-arn "arn:aws:healthlake:us-east-1:674914422125:datastore/
fhir/0725c83f4307f263e16fd56b6d8ebdbe" \
  --region us-east-1

```

Output:

```

{
  "tags": {
    "key": "value",

```

```
    "key1": "value1"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya di Amazon HealthLake](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## start-fhir-export-job

Contoh kode berikut menunjukkan cara menggunakan `start-fhir-export-job`.

### AWS CLI

Untuk memulai pekerjaan ekspor FHIR

`start-fhir-export-job` Contoh berikut menunjukkan cara memulai pekerjaan ekspor FHIR menggunakan Amazon HealthLake.

```
aws healthlake start-fhir-export-job \
  --output-data-config S3Uri="s3://(Bucket Name)/(Prefix Name)/" \
  --datastore-id (Datastore ID) \
  --data-access-role-arn arn:aws:iam::(AWS Account ID):role/(Role Name)
```

Output:

```
{
  "DatastoreId": "(Datastore ID)",
  "JobStatus": "SUBMITTED",
  "JobId": "9b9a51943afaedd0a8c0c26c49135a31"
}
```

Untuk informasi selengkapnya, lihat [Mengekspor file dari Penyimpanan Data FHIR](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [StartFhirExportJob](#) di Referensi AWS CLI Perintah.

## start-fhir-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-fhir-import-job`.

## AWS CLI

Untuk memulai pekerjaan impor FHIR

`start-fhir-import-job` Contoh berikut menunjukkan cara memulai pekerjaan impor FHIR menggunakan Amazon HealthLake.

```
aws healthlake start-fhir-import-job \  
  --input-data-config S3Uri="s3://(Bucket Name)/(Prefix Name)/" \  
  --datastore-id (Datastore ID) \  
  --data-access-role-arn "arn:aws:iam::(AWS Account ID):role/(Role Name)" \  
  --region us-east-1
```

Output:

```
{  
  "DatastoreId": "(Datastore ID)",  
  "JobStatus": "SUBMITTED",  
  "JobId": "c145fbb27b192af392f8ce6e7838e34f"  
}
```

Untuk informasi selengkapnya, lihat Mengimpor file ke Toko Data FHIR 'https://docs.aws.amazon.com/healthlake/latest/devguide/import-datastore.html' di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [StartFhirImportJob](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menambahkan tag ke Data Store

`tag-resource` Contoh berikut menunjukkan cara menambahkan tag ke Data Store.

```
aws healthlake tag-resource \  
  --resource-arn "arn:aws:healthlake:us-east-1:691207106566:datastore/  
fhir/0725c83f4307f263e16fd56b6d8ebdbe" \  
  --tags '[{"Key": "key1", "Value": "value1"}]' \  

```

```
--region us-east-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat 'Menambahkan tag ke Penyimpanan Data < <https://docs.aws.amazon.com/healthlake/latest/devguide/add-a-tag.html>>'\_\_ di Panduan Pengembang Amazon. HealthLake .

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari Penyimpanan Data.

`untag-resource` Contoh berikut menunjukkan cara menghapus tag dari Data Store.

```
aws healthlake untag-resource \  
  --resource-arn "arn:aws:healthlake:us-east-1:674914422125:datastore/fhir/  
b91723d65c6fdeb1d26543a49d2ed1fa" \  
  --tag-keys '["key1"]' \  
  --region us-east-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus tag dari Penyimpanan Data](#) di Panduan HealthLake Pengembang Amazon.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## HealthOmics contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with HealthOmics.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **abort-multipart-read-set-upload**

Contoh kode berikut menunjukkan cara menggunakan `abort-multipart-read-set-upload`.

AWS CLI

Untuk menghentikan unggahan set baca multibagian

`abort-multipart-read-set-upload` Contoh berikut menghentikan unggahan set baca multibagian ke toko HealthOmics urutan Anda.

```
aws omics abort-multipart-read-set-upload \  
  --sequence-store-id 0123456789 \  
  --upload-id 1122334455
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Unggahan langsung ke penyimpanan urutan](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [AbortMultipartReadSetUpload](#) di Referensi AWS CLI Perintah.

### **accept-share**

Contoh kode berikut menunjukkan cara menggunakan `accept-share`.

AWS CLI

Untuk menerima bagian data penyimpanan analitik

`accept-share` Contoh berikut menerima bagian data penyimpanan HealthOmics analitik.

```
aws omics accept-share \  
  ----share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Output:

```
{  
  "status": "ACTIVATING"  
}
```

Untuk informasi selengkapnya, lihat [Berbagi lintas akun](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [AcceptShare](#) di Referensi AWS CLI Perintah.

## batch-delete-read-set

Contoh kode berikut menunjukkan cara menggunakan `batch-delete-read-set`.

### AWS CLI

Untuk menghapus beberapa set baca

`batch-delete-read-set` Contoh berikut menghapus dua set baca.

```
aws omics batch-delete-read-set \  
  --sequence-store-id 1234567890 \  
  --ids 1234567890 0123456789
```

Jika ada kesalahan menghapus salah satu set baca yang ditentukan, layanan mengembalikan daftar kesalahan.

```
{  
  "errors": [  
    {  
      "code": "",  
      "id": "0123456789",  
      "message": "The specified readset does not exist."  
    }  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [BatchDeleteReadSet](#) di Referensi AWS CLI Perintah.

## **cancel-annotation-import-job**

Contoh kode berikut menunjukkan cara menggunakan `cancel-annotation-import-job`.

### AWS CLI

Untuk membatalkan pekerjaan impor anotasi

`cancel-annotation-import-job` Contoh berikut membatalkan pekerjaan impor anotasi dengan ID. `04f57618-xmpl-4fd0-9349-e5a85aefb997`

```
aws omics cancel-annotation-import-job \  
  --job-id 04f57618-xmpl-4fd0-9349-e5a85aefb997
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CancelAnnotationImportJob](#) di Referensi AWS CLI Perintah.

## **cancel-run**

Contoh kode berikut menunjukkan cara menggunakan `cancel-run`.

### AWS CLI

Untuk membatalkan proses

`cancel-run` Contoh berikut membatalkan run dengan ID `1234567`.

```
aws omics cancel-run \  
  --id 1234567
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CancelRun](#) di Referensi AWS CLI Perintah.

## **cancel-variant-import-job**

Contoh kode berikut menunjukkan cara menggunakan `cancel-variant-import-job`.

### AWS CLI

Untuk membatalkan pekerjaan impor varian

`cancel-variant-import-job` Contoh berikut membatalkan pekerjaan impor varian dengan ID `69cb65d6-xmpl-4a4a-9025-4565794b684e`.

```
aws omics cancel-variant-import-job \  
  --job-id 69cb65d6-xmpl-4a4a-9025-4565794b684e
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CancelVariantImportJob](#) di Referensi AWS CLI Perintah.

## **complete-multipart-read-set-upload**

Contoh kode berikut menunjukkan cara menggunakan `complete-multipart-read-set-upload`.

### AWS CLI

Untuk mengakhiri unggahan multipart setelah Anda mengunggah semua komponen.

`complete-multipart-read-set-upload` Contoh berikut menyimpulkan upload multipart ke dalam toko urutan setelah semua komponen telah diunggah.

```
aws omics complete-multipart-read-set-upload \  
  --sequence-store-id 0123456789 \  
  --upload-id 1122334455 \  
  --parts '[{"checksum":"gaCBQMe+rpCFZxLpoP6gydBoXaKKDA/  
Vobh5zBD4W4=", "partNumber":1, "partSource":"SOURCE1"}]'
```

Output:

```
{
```



```
"readSetId": "0000000001"  
"readSetId": "0000000002"  
"readSetId": "0000000003"  
}
```

Untuk informasi selengkapnya, lihat [Unggahan langsung ke penyimpanan urutan](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [CompleteMultipartReadSetUpload](#) di Referensi AWS CLI Perintah.

## create-annotation-store-version

Contoh kode berikut menunjukkan cara menggunakan `create-annotation-store-version`.

### AWS CLI

Untuk membuat versi baru dari toko anotasi

`create-annotation-store-version` Contoh berikut membuat versi baru dari toko anotasi.

```
aws omics create-annotation-store-version \  
  --name my_annotation_store \  
  --version-name my_version
```

Output:

```
{  
  "creationTime": "2023-07-21T17:15:49.251040+00:00",  
  "id": "3b93cdef69d2",  
  "name": "my_annotation_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-  
west-2:555555555555:referenceStore/6505293348/reference/5987565360"  
  },  
  "status": "CREATING",  
  "versionName": "my_version"  
}
```

Untuk informasi selengkapnya, lihat [Membuat versi baru penyimpanan anotasi](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [CreateAnnotationStoreVersion](#) di Referensi AWS CLI Perintah.

## create-annotation-store

Contoh kode berikut menunjukkan cara menggunakan `create-annotation-store`.

### AWS CLI

Contoh 1: Untuk membuat toko anotasi VCF

`create-annotation-store` Contoh berikut membuat penyimpanan anotasi format VCF.

```
aws omics create-annotation-store \  
  --name my_ann_store \  
  --store-format VCF \  
  --reference referenceArn=arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

Output:

```
{  
  "creationTime": "2022-11-23T22:48:39.226492Z",  
  "id": "0a91xmplc71f",  
  "name": "my_ann_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
  },  
  "status": "CREATING",  
  "storeFormat": "VCF"  
}
```

Contoh 2: Untuk membuat toko anotasi TSV

`create-annotation-store` Contoh berikut membuat penyimpanan anotasi format TSV.

```
aws omics create-annotation-store \  
  --name tsv_ann_store \  
  --store-format TSV \  
  --reference referenceArn=arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890 \  
  --store-options file://tsv-store-options.json
```

`tsv-store-options.json` mengonfigurasi opsi format untuk anotasi.

```
{
  "tsvStoreOptions": {
    "annotationType": "CHR_START_END_ZERO_BASE",
    "formatToHeader": {
      "CHR": "chromosome",
      "START": "start",
      "END": "end"
    },
    "schema": [
      {
        "chromosome": "STRING"
      },
      {
        "start": "LONG"
      },
      {
        "end": "LONG"
      },
      {
        "name": "STRING"
      }
    ]
  }
}
```

### Output:

```
{
  "creationTime": "2022-11-30T01:28:08.525586Z",
  "id": "861cxmpl96b0",
  "name": "tsv_ann_store",
  "reference": {
    "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890"
  },
  "status": "CREATING",
  "storeFormat": "TSV",
  "storeOptions": {
    "tsvStoreOptions": {
      "annotationType": "CHR_START_END_ZERO_BASE",
      "formatToHeader": {
        "CHR": "chromosome",
        "END": "end",
```

```

        "START": "start"
    },
    "schema": [
        {
            "chromosome": "STRING"
        },
        {
            "start": "LONG"
        },
        {
            "end": "LONG"
        },
        {
            "name": "STRING"
        }
    ]
}
}
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CreateAnnotationStore](#) di Referensi AWS CLI Perintah.

## create-multipart-read-set-upload

Contoh kode berikut menunjukkan cara menggunakan `create-multipart-read-set-upload`.

### AWS CLI

Untuk memulai unggahan set baca multibagian.

`create-multipart-read-set-upload` Contoh berikut memulai upload set baca multipart.

```

aws omics create-multipart-read-set-upload \
  --sequence-store-id 0123456789 \
  --name HG00146 \
  --source-file-type FASTQ \
  --subject-id mySubject\
  --sample-id mySample\
  --description "FASTQ for HG00146"\
  --generated-from "1000 Genomes"

```

**Output:**

```
{
  "creationTime": "2022-07-13T23:25:20Z",
  "description": "FASTQ for HG00146",
  "generatedFrom": "1000 Genomes",
  "name": "HG00146",
  "sampleId": "mySample",
  "sequenceStoreId": "0123456789",
  "sourceFileType": "FASTQ",
  "subjectId": "mySubject",
  "uploadId": "1122334455"
}
```

Untuk informasi selengkapnya, lihat [Unggahan langsung ke penyimpanan urutan](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [CreateMultipartReadSetUpload](#) di Referensi AWS CLI Perintah.

**create-reference-store**

Contoh kode berikut menunjukkan cara menggunakan `create-reference-store`.

**AWS CLI**

Untuk membuat toko referensi

`create-reference-store` Contoh berikut membuat toko referensi `my-ref-store`.

```
aws omics create-reference-store \
  --name my-ref-store
```

**Output:**

```
{
  "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",
  "creationTime": "2022-11-22T22:13:25.947Z",
  "id": "1234567890",
  "name": "my-ref-store"
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CreateReferenceStore](#) di Referensi AWS CLI Perintah.

## **create-run-group**

Contoh kode berikut menunjukkan cara menggunakan `create-run-group`.

### AWS CLI

Untuk membuat grup run

`create-run-group` Contoh berikut membuat grup run bernama `cram-converter`.

```
aws omics create-run-group \  
  --name cram-converter \  
  --max-cpus 20 \  
  --max-duration 600
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",  
  "id": "1234567",  
  "tags": {}  
}
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CreateRunGroup](#) di Referensi AWS CLI Perintah.

## **create-sequence-store**

Contoh kode berikut menunjukkan cara menggunakan `create-sequence-store`.

### AWS CLI

Untuk membuat toko urutan

`create-sequence-store` Contoh berikut membuat toko urutan.

```
aws omics create-sequence-store \  
  --name my-seq-store
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890",  
  "creationTime": "2022-11-23T01:24:33.629Z",  
  "id": "1234567890",  
  "name": "my-seq-store"  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CreateSequenceStore](#) di Referensi AWS CLI Perintah.

## create-share

Contoh kode berikut menunjukkan cara menggunakan `create-share`.

### AWS CLI

Untuk membuat bagian dari toko HealthOmics analitik

`create-share` Contoh berikut menunjukkan cara membuat bagian dari toko HealthOmics analitik yang dapat diterima oleh pelanggan di luar akun.

```
aws omics create-share \  
  --resource-arn "arn:aws:omics:us-west-2:555555555555:variantStore/  
omics_dev_var_store" \  
  --principal-subscriber "123456789012" \  
  --name "my_Share-123"
```

Output:

```
{  
  "shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",  
  "name": "my_Share-123",  
  "status": "PENDING"  
}
```

Untuk informasi selengkapnya, lihat [Berbagi lintas akun di Panduan Pengguna.AWS HealthOmics](#)

- Untuk detail API, lihat [CreateShare](#) di Referensi AWS CLI Perintah.

## **create-variant-store**

Contoh kode berikut menunjukkan cara menggunakan `create-variant-store`.

### AWS CLI

Untuk membuat toko varian

`create-variant-store` Contoh berikut membuat toko varian bernama `my_var_store`.

```
aws omics create-variant-store \  
  --name my_var_store \  
  --reference referenceArn=arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

Output:

```
{  
  "creationTime": "2022-11-23T22:09:07.534499Z",  
  "id": "02dexmplcfdd",  
  "name": "my_var_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
  },  
  "status": "CREATING"  
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CreateVariantStore](#) di Referensi AWS CLI Perintah.

## **create-workflow**

Contoh kode berikut menunjukkan cara menggunakan `create-workflow`.

### AWS CLI

Untuk membuat alur kerja



`create-workflow` Contoh berikut menciptakan alur kerja WDL.

```
aws omics create-workflow \  
  --name cram-converter \  
  --engine WDL \  
  --definition-zip fileb://workflow-crambam.zip \  
  --parameter-template file://workflow-params.json
```

`workflow-crambam.zip` adalah arsip ZIP yang berisi definisi alur kerja. `workflow-params.json` mendefinisikan parameter runtime untuk alur kerja.

```
{  
  "ref_fasta" : {  
    "description": "Reference genome fasta file",  
    "optional": false  
  },  
  "ref_fasta_index" : {  
    "description": "Index of the reference genome fasta file",  
    "optional": false  
  },  
  "ref_dict" : {  
    "description": "dictionary file for 'ref_fasta'",  
    "optional": false  
  },  
  "input_cram" : {  
    "description": "The Cram file to convert to BAM",  
    "optional": false  
  },  
  "sample_name" : {  
    "description": "The name of the input sample, used to name the output BAM",  
    "optional": false  
  }  
}
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",  
  "id": "1234567",  
  "status": "CREATING",  
  "tags": {}  
}
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [CreateWorkflow](#) di Referensi AWS CLI Perintah.

## **delete-annotation-store-versions**

Contoh kode berikut menunjukkan cara menggunakan `delete-annotation-store-versions`.

### AWS CLI

Untuk menghapus versi penyimpanan anotasi

`delete-annotation-store-versions` Contoh berikut menghapus versi toko anotasi.

```
aws omics delete-annotation-store-versions \  
  --name my_annotation_store \  
  --versions my_version
```

Output:

```
{  
  "errors": []  
}
```

Untuk informasi selengkapnya, lihat [Membuat versi baru penyimpanan anotasi](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [DeleteAnnotationStoreVersions](#) di Referensi AWS CLI Perintah.

## **delete-annotation-store**

Contoh kode berikut menunjukkan cara menggunakan `delete-annotation-store`.

### AWS CLI

Untuk menghapus toko anotasi

`delete-annotation-store` Contoh berikut menghapus toko anotasi bernama `my_vcf_store`

```
aws omics delete-annotation-store \  
  --name my_vcf_store
```

**Output:**

```
{
  "status": "DELETING"
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteAnnotationStore](#) di Referensi AWS CLI Perintah.

**delete-reference-store**

Contoh kode berikut menunjukkan cara menggunakan `delete-reference-store`.

**AWS CLI**

Untuk menghapus toko referensi

`delete-reference-store` Contoh berikut menghapus toko referensi dengan ID1234567890.

```
aws omics delete-reference-store \
  --id 1234567890
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteReferenceStore](#) di Referensi AWS CLI Perintah.

**delete-reference**

Contoh kode berikut menunjukkan cara menggunakan `delete-reference`.

**AWS CLI**

Untuk menghapus referensi

`delete-reference` Contoh berikut menghapus referensi.

```
aws omics delete-reference \
  --reference-store-id 1234567890 \
  --id 1234567890
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteReference](#) di Referensi AWS CLI Perintah.

## **delete-run-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-run-group`.

### AWS CLI

Untuk menghapus grup run

`delete-run-group` Contoh berikut menghapus grup run dengan ID1234567.

```
aws omics delete-run-group \  
  --id 1234567
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteRunGroup](#) di Referensi AWS CLI Perintah.

## **delete-run**

Contoh kode berikut menunjukkan cara menggunakan `delete-run`.

### AWS CLI

Untuk menghapus alur kerja yang dijalankan

`delete-run` Contoh berikut menghapus run dengan ID1234567.

```
aws omics delete-run \  
  --id 1234567
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteRun](#) di Referensi AWS CLI Perintah.

## **delete-sequence-store**

Contoh kode berikut menunjukkan cara menggunakan `delete-sequence-store`.

## AWS CLI

Untuk menghapus toko urutan

`delete-sequence-store` Contoh berikut menghapus toko urutan dengan ID1234567890.

```
aws omics delete-sequence-store \  
  --id 1234567890
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteSequenceStore](#) di Referensi AWS CLI Perintah.

## `delete-share`

Contoh kode berikut menunjukkan cara menggunakan `delete-share`.

## AWS CLI

Untuk menghapus bagian data HealthOmics analitik

`delete-share` Contoh berikut menghapus pangsa data analitik lintas akun.

```
aws omics delete-share \  
  --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Output:

```
{  
  "status": "DELETING"  
}
```

Untuk informasi selengkapnya, lihat [Berbagi lintas akun](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [DeleteShare](#) di Referensi AWS CLI Perintah.

## `delete-variant-store`

Contoh kode berikut menunjukkan cara menggunakan `delete-variant-store`.

## AWS CLI

Untuk menghapus toko varian

`delete-variant-store` Contoh berikut menghapus toko varian bernama `my_var_store`.

```
aws omics delete-variant-store \  
  --name my_var_store
```

Output:

```
{  
  "status": "DELETING"  
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteVariantStore](#) di Referensi AWS CLI Perintah.

## `delete-workflow`

Contoh kode berikut menunjukkan cara menggunakan `delete-workflow`.

## AWS CLI

Untuk menghapus alur kerja

`delete-workflow` Contoh berikut menghapus alur kerja dengan ID. 1234567

```
aws omics delete-workflow \  
  --id 1234567
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [DeleteWorkflow](#) di Referensi AWS CLI Perintah.

## `get-annotation-import-job`

Contoh kode berikut menunjukkan cara menggunakan `get-annotation-import-job`.

## AWS CLI

Untuk melihat pekerjaan impor anotasi

`get-annotation-import-job` Contoh berikut mendapatkan detail tentang pekerjaan impor anotasi.

```
aws omics get-annotation-import-job \  
  --job-id 984162c7-xmpl-4d23-ab47-286f7950bfbf
```

Output:

```
{  
  "creationTime": "2022-11-30T01:40:11.017746Z",  
  "destinationName": "tsv_ann_store",  
  "id": "984162c7-xmpl-4d23-ab47-286f7950bfbf",  
  "items": [  
    {  
      "jobStatus": "COMPLETED",  
      "source": "s3://omics-artifacts-01d6xmpl4e72dd32/targetedregions.bed.gz"  
    }  
  ],  
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ",  
  "runLeftNormalization": false,  
  "status": "COMPLETED",  
  "updateTime": "2022-11-30T01:42:39.134009Z"  
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetAnnotationImportJob](#) di Referensi AWS CLI Perintah.

## `get-annotation-store-version`

Contoh kode berikut menunjukkan cara menggunakan `get-annotation-store-version`.

AWS CLI

Untuk mengambil metadata untuk versi penyimpanan anotasi

`get-annotation-store-version` Contoh berikut mengambil metadata untuk versi penyimpanan anotasi yang diminta.

```
aws omics get-annotation-store-version \  
  --name my_annotation_store \  
  --version 1
```

```
--version-name my_version
```

Output:

```
{
  "storeId": "4934045d1c6d",
  "id": "2a3f4a44aa7b",
  "status": "ACTIVE",
  "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version",
  "name": "my_annotation_store",
  "versionName": "my_version",
  "creationTime": "2023-07-21T17:15:49.251040+00:00",
  "updateTime": "2023-07-21T17:15:56.434223+00:00",
  "statusMessage": "",
  "versionSizeBytes": 0
}
```

Untuk informasi selengkapnya, lihat [Membuat versi baru penyimpanan anotasi](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [GetAnnotationStoreVersion](#) di Referensi AWS CLI Perintah.

## get-annotation-store

Contoh kode berikut menunjukkan cara menggunakan `get-annotation-store`.

### AWS CLI

Untuk melihat toko anotasi

`get-annotation-store` Contoh berikut mendapatkan rincian tentang toko anotasi bernama `my_ann_store`.

```
aws omics get-annotation-store \
  --name my_ann_store
```

Output:

```
{
  "creationTime": "2022-11-23T22:48:39.226492Z",
  "id": "0a91xmplc71f",
```



```

    "name": "my_ann_store",
    "reference": {
      "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING",
    "storeArn": "arn:aws:omics:us-west-2:123456789012:annotationStore/my_ann_store",
    "storeFormat": "VCF",
    "storeSizeBytes": 0,
    "tags": {}
  }
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetAnnotationStore](#) di Referensi AWS CLI Perintah.

## get-read-set-activation-job

Contoh kode berikut menunjukkan cara menggunakan `get-read-set-activation-job`.

### AWS CLI

Untuk melihat pekerjaan aktivasi set baca

`get-read-set-activation-job` Contoh berikut mendapatkan detail tentang pekerjaan aktivasi set baca.

```

aws omics get-read-set-activation-job \
  --sequence-store-id 1234567890 \
  --id 1234567890

```

Output:

```

{
  "completionTime": "2022-12-06T22:33:42.828Z",
  "creationTime": "2022-12-06T22:32:45.213Z",
  "id": "1234567890",
  "sequenceStoreId": "1234567890",
  "sources": [
    {
      "readSetId": "1234567890",
      "status": "FINISHED",

```

```
        "statusMessage": "No activation needed as read set is already in
ACTIVATING or ACTIVE state."
    }
  ],
  "status": "COMPLETED",
  "statusMessage": "The job completed successfully."
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReadSetActivationJob](#) di Referensi AWS CLI Perintah.

## get-read-set-export-job

Contoh kode berikut menunjukkan cara menggunakan `get-read-set-export-job`.

### AWS CLI

Untuk melihat pekerjaan ekspor set baca

`get-read-set-export-job` Contoh berikut mendapatkan detail tentang pekerjaan ekspor set baca.

```
aws omics get-read-set-export-job \
  --sequence-store-id 1234567890 \
  --id 1234567890
```

Output:

```
{
  "completionTime": "2022-12-06T22:39:14.491Z",
  "creationTime": "2022-12-06T22:37:18.612Z",
  "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
  "id": "1234567890",
  "sequenceStoreId": "1234567890",
  "status": "COMPLETED",
  "statusMessage": "The job is submitted and will start soon."
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReadSetExportJob](#) di Referensi AWS CLI Perintah.

## get-read-set-import-job

Contoh kode berikut menunjukkan cara menggunakan `get-read-set-import-job`.

### AWS CLI

Untuk melihat pekerjaan impor set baca

`get-read-set-import-job` Contoh berikut mendapatkan rincian tentang pekerjaan impor set baca.

```
aws omics get-read-set-import-job \  
  --sequence-store-id 1234567890 \  
  --id 1234567890
```

Output:

```
{  
  "creationTime": "2022-11-23T01:36:38.158Z",  
  "id": "1234567890",  
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ",  
  "sequenceStoreId": "1234567890",  
  "sources": [  
    {  
      "name": "HG00100",  
      "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890",  
      "sampleId": "bam-sample",  
      "sourceFileType": "BAM",  
      "sourceFiles": {  
        "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/  
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam",  
        "source2": ""  
      },  
      "status": "IN_PROGRESS",  
      "statusMessage": "The source job is currently in progress.",  
      "subjectId": "bam-subject",  
      "tags": {  
        "aws:omics:sampleId": "bam-sample",  
        "aws:omics:subjectId": "bam-subject"  
      }  
    }  
  ]  
}
```

```
    }
  },
  {
    "name": "HG00146",
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
    "sampleId": "fastq-sample",
    "sourceFileType": "FASTQ",
    "sourceFiles": {
      "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
SRR233106_1.filt.fastq.gz",
      "source2": "s3://omics-artifacts-01d6xmpl4e72dd32/
SRR233106_2.filt.fastq.gz"
    },
    "status": "IN_PROGRESS",
    "statusMessage": "The source job is currently in progress.",
    "subjectId": "fastq-subject",
    "tags": {
      "aws:omics:sampleId": "fastq-sample",
      "aws:omics:subjectId": "fastq-subject"
    }
  },
  {
    "name": "HG00096",
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
    "sampleId": "cram-sample",
    "sourceFileType": "CRAM",
    "sourceFiles": {
      "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram",
      "source2": ""
    },
    "status": "IN_PROGRESS",
    "statusMessage": "The source job is currently in progress.",
    "subjectId": "cram-subject",
    "tags": {
      "aws:omics:sampleId": "cram-sample",
      "aws:omics:subjectId": "cram-subject"
    }
  }
],
"status": "IN_PROGRESS",
"statusMessage": "The job is currently in progress."
```

```
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReadSetImportJob](#) di Referensi AWS CLI Perintah.

## get-read-set-metadata

Contoh kode berikut menunjukkan cara menggunakan `get-read-set-metadata`.

### AWS CLI

Untuk melihat set baca

`get-read-set-metadata` Contoh berikut mendapatkan rincian tentang file set baca.

```
aws omics get-read-set-metadata \  
  --sequence-store-id 1234567890 \  
  --id 1234567890
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/  
readSet/1234567890",  
  "creationTime": "2022-11-23T21:55:00.515Z",  
  "fileType": "FASTQ",  
  "files": {  
    "source1": {  
      "contentLength": 310054739,  
      "partSize": 104857600,  
      "totalParts": 3  
    },  
    "source2": {  
      "contentLength": 307846621,  
      "partSize": 104857600,  
      "totalParts": 3  
    }  
  },  
  "id": "1234567890",  
  "name": "HG00146",
```

```
"referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/
reference/1234567890",
"sampleId": "fastq-sample",
"sequenceInformation": {
  "alignment": "UNALIGNED",
  "totalBaseCount": 677717384,
  "totalReadCount": 8917334
},
"sequenceStoreId": "1234567890",
"status": "ACTIVE",
"subjectId": "fastq-subject"
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReadSetMetadati](#) Referensi AWS CLI Perintah.

## get-read-set

Contoh kode berikut menunjukkan cara menggunakan `get-read-set`.

### AWS CLI

Untuk mengunduh set baca

`get-read-set` Contoh berikut mengunduh bagian 3 dari set baca sebagai `1234567890.3.bam`.

```
aws omics get-read-set \
  --sequence-store-id 1234567890 \
  --id 1234567890 \
  --part-number 3 1234567890.3.bam
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReadSet](#) di Referensi AWS CLI Perintah.

## get-reference-import-job

Contoh kode berikut menunjukkan cara menggunakan `get-reference-import-job`.

## AWS CLI

Untuk melihat pekerjaan impor referensi

`get-reference-import-job` Contoh contoh berikut mendapatkan rincian tentang pekerjaan impor referensi.

```
aws omics get-reference-import-job \  
  --reference-store-id 1234567890 \  
  --id 1234567890
```

Output:

```
{  
  "creationTime": "2022-11-22T22:25:41.124Z",  
  "id": "1234567890",  
  "referenceStoreId": "1234567890",  
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ",  
  "sources": [  
    {  
      "name": "assembly-38",  
      "sourceFile": "s3://omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.fasta",  
      "status": "IN_PROGRESS",  
      "statusMessage": "The source job is currently in progress."  
    }  
  ],  
  "status": "IN_PROGRESS",  
  "statusMessage": "The job is currently in progress."  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReferenceImportJob](#) di Referensi AWS CLI Perintah.

### **get-reference-metadata**

Contoh kode berikut menunjukkan cara menggunakan `get-reference-metadata`.

## AWS CLI

Untuk melihat referensi

`get-reference-metadata` Contoh berikut mendapatkan rincian tentang referensi.

```
aws omics get-reference-metadata \  
  --reference-store-id 1234567890 \  
  --id 1234567890
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/  
reference/1234567890",  
  "creationTime": "2022-11-22T22:27:09.033Z",  
  "files": {  
    "index": {  
      "contentLength": 160928,  
      "partSize": 104857600,  
      "totalParts": 1  
    },  
    "source": {  
      "contentLength": 3249912778,  
      "partSize": 104857600,  
      "totalParts": 31  
    }  
  },  
  "id": "1234567890",  
  "md5": "7ff134953dcca8c8997453bbb80b6b5e",  
  "name": "assembly-38",  
  "referenceStoreId": "1234567890",  
  "status": "ACTIVE",  
  "updateTime": "2022-11-22T22:27:09.033Z"  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReferenceMetadata](#) di Referensi AWS CLI Perintah.



## get-reference-store

Contoh kode berikut menunjukkan cara menggunakan `get-reference-store`.

### AWS CLI

Untuk melihat toko referensi

`get-reference-store` Contoh berikut mendapatkan rincian tentang toko referensi.

```
aws omics get-reference-store \  
  --id 1234567890
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",  
  "creationTime": "2022-09-23T23:27:20.364Z",  
  "id": "1234567890",  
  "name": "my-rstore-0"  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReferenceStore](#) di Referensi AWS CLI Perintah.

## get-reference

Contoh kode berikut menunjukkan cara menggunakan `get-reference`.

### AWS CLI

Untuk mengunduh referensi genom

`get-reference` Contoh berikut mengunduh bagian 1 dari genom sebagai `hg38.1.fa`.

```
aws omics get-reference \  
  --reference-store-id 1234567890 \  
  --id 1234567890 \  
  --part-number 1 hg38.1.fa
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetReferenced](#) di Referensi AWS CLI Perintah.

## get-run-group

Contoh kode berikut menunjukkan cara menggunakan `get-run-group`.

### AWS CLI

Untuk melihat grup run

`get-run-group` Contoh berikut mendapatkan rincian tentang grup run.

```
aws omics get-run-group \
  --id 1234567
```

Output:

```
{
  "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
  "creationTime": "2022-12-01T00:58:42.915219Z",
  "id": "1234567",
  "maxCpus": 20,
  "maxDuration": 600,
  "name": "cram-convert",
  "tags": {}
}
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetRunGroup](#) di Referensi AWS CLI Perintah.

## get-run-task

Contoh kode berikut menunjukkan cara menggunakan `get-run-task`.

### AWS CLI

Untuk melihat tugas

`get-run-task` Contoh berikut mendapatkan rincian tentang tugas alur kerja.

```
aws omics get-run-task \  
  --id 1234567 \  
  --task-id 1234567
```

Output:

```
{  
  "cpus": 1,  
  "creationTime": "2022-11-30T23:13:00.718651Z",  
  "logStream": "arn:aws:logs:us-west-2:123456789012:log-group:/aws/omics/  
WorkflowLog:log-stream:run/1234567/task/1234567",  
  "memory": 15,  
  "name": "CramToBamTask",  
  "startTime": "2022-11-30T23:17:47.016Z",  
  "status": "COMPLETED",  
  "stopTime": "2022-11-30T23:18:21.503Z",  
  "taskId": "1234567"  
}
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetRunTask](#) di Referensi AWS CLI Perintah.

## get-run

Contoh kode berikut menunjukkan cara menggunakan `get-run`.

### AWS CLI

Untuk melihat alur kerja yang dijalankan

`get-run` Contoh berikut mendapatkan rincian tentang alur kerja yang dijalankan.

```
aws omics get-run \  
  --id 1234567
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",  
  "creationTime": "2022-11-30T22:58:22.615865Z",
```

```

    "digest":
      "sha256:c54bxmpl1742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
      "id": "1234567",
      "name": "cram-to-bam",
      "outputUri": "s3://omics-artifacts-01d6xmpl14e72dd32/workflow-output/",
      "parameters": {
        "ref_dict": "s3://omics-artifacts-01d6xmpl14e72dd32/
Homo_sapiens_assembly38.dict",
        "ref_fasta_index": "s3://omics-artifacts-01d6xmpl14e72dd32/
Homo_sapiens_assembly38.fasta.fai",
        "ref_fasta": "s3://omics-artifacts-01d6xmpl14e72dd32/
Homo_sapiens_assembly38.fasta",
        "sample_name": "NA12878",
        "input_cram": "s3://omics-artifacts-01d6xmpl14e72dd32/NA12878.cram"
      },
      "resourceDigests": {
        "s3://omics-artifacts-01d6xmpl14e72dd32/Homo_sapiens_assembly38.fasta.fai":
"etag:f76371b113734a56cde236bc0372de0a",
        "s3://omics-artifacts-01d6xmpl14e72dd32/Homo_sapiens_assembly38.dict":
"etag:3884c62eb0e53fa92459ed9bfff133ae6",
        "s3://omics-artifacts-01d6xmpl14e72dd32/Homo_sapiens_assembly38.fasta":
"etag:e307d81c605fb91b7720a08f00276842-388",
        "s3://omics-artifacts-01d6xmpl14e72dd32/NA12878.cram":
"etag:a9f52976381286c6143b5cc681671ec6"
      },
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
      "startedBy": "arn:aws:iam::123456789012:user/laptop-2020",
      "status": "STARTING",
      "tags": {},
      "workflowId": "1234567",
      "workflowType": "PRIVATE"
  }
}

```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetRun](#) di Referensi AWS CLI Perintah.

## get-sequence-store

Contoh kode berikut menunjukkan cara menggunakan `get-sequence-store`.

## AWS CLI

Untuk melihat toko urutan

`get-sequence-store` Contoh berikut mendapatkan rincian tentang toko urutan dengan ID1234567890.

```
aws omics get-sequence-store \  
  --id 1234567890
```

Output:

```
{  
  "arn": "arn:aws:omics:us-east-1:123456789012:sequenceStore/1234567890",  
  "creationTime": "2022-11-23T19:55:48.376Z",  
  "id": "1234567890",  
  "name": "my-seq-store"  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetSequenceStore](#) di Referensi AWS CLI Perintah.

## `get-share`

Contoh kode berikut menunjukkan cara menggunakan `get-share`.

## AWS CLI

Untuk mengambil metadata tentang bagian data analitik HealthOmics

`get-share` Contoh berikut mengambil metadata untuk berbagi data analitik lintas akun.

```
aws omics get-share \  
  --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Output:

```
{  
  "share": {
```

```

    "shareId": "495c21bedc889d07d0ab69d710a6841e-
dd75ab7a1a9c384fa848b5bd8e5a7e0a",
    "name": "my_Share-123",
    "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store",
    "principalSubscriber": "123456789012",
    "ownerId": "555555555555",
    "status": "PENDING"
  }
}

```

Untuk informasi selengkapnya, lihat [Berbagi lintas akun](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [GetShared](#) di Referensi AWS CLI Perintah.

## get-variant-import-job

Contoh kode berikut menunjukkan cara menggunakan `get-variant-import-job`.

### AWS CLI

Untuk melihat pekerjaan impor varian

`get-variant-import-job` Contoh berikut mendapatkan detail tentang pekerjaan impor varian.

```

aws omics get-variant-import-job \
  --job-id edd7b8ce-xmpl-47e2-bc99-258cac95a508

```

Output:

```

{
  "creationTime": "2022-11-23T22:42:50.037812Z",
  "destinationName": "my_var_store",
  "id": "edd7b8ce-xmpl-47e2-bc99-258cac95a508",
  "items": [
    {
      "jobStatus": "IN_PROGRESS",
      "source": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.known_indels.vcf.gz"
    }
  ],
}

```

```
"roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "runLeftNormalization": false,
  "status": "IN_PROGRESS",
  "updateTime": "2022-11-23T22:43:05.898309Z"
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetVariantImportJob](#) di Referensi AWS CLI Perintah.

## get-variant-store

Contoh kode berikut menunjukkan cara menggunakan `get-variant-store`.

### AWS CLI

Untuk melihat toko varian

`get-variant-store` Contoh berikut mendapatkan detail tentang toko varian.

```
aws omics get-variant-store \
  --name my_var_store
```

Output:

```
{
  "creationTime": "2022-11-23T22:09:07.534499Z",
  "id": "02dexmplcfdd",
  "name": "my_var_store",
  "reference": {
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
  },
  "status": "CREATING",
  "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/my_var_store",
  "storeSizeBytes": 0,
  "tags": {},
  "updateTime": "2022-11-23T22:09:24.931711Z"
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetVariantStore](#) di Referensi AWS CLI Perintah.

## get-workflow

Contoh kode berikut menunjukkan cara menggunakan `get-workflow`.

### AWS CLI

Untuk melihat alur kerja

`get-workflow` Contoh berikut mendapatkan rincian tentang alur kerja dengan ID1234567.

```
aws omics get-workflow \  
  --id 1234567
```

Output:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",  
  "creationTime": "2022-11-30T22:33:16.225368Z",  
  "digest":  
    "sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",  
  "engine": "WDL",  
  "id": "1234567",  
  "main": "workflow-crambam.wdl",  
  "name": "cram-converter",  
  "parameterTemplate": {  
    "ref_dict": {  
      "description": "dictionary file for 'ref_fasta'"  
    },  
    "ref_fasta_index": {  
      "description": "Index of the reference genome fasta file"  
    },  
    "ref_fasta": {  
      "description": "Reference genome fasta file"  
    },  
    "input_cram": {  
      "description": "The Cram file to convert to BAM"  
    },  
    "sample_name": {  
      "description": "The name of the input sample, used to name the output  
BAM"    }  
  }  
}
```



```

    }
  },
  "status": "ACTIVE",
  "statusMessage": "workflow-crambam.wdl\n      workflow CramToBamFlow\n
call CramToBamTask\n      call ValidateSamFile\n      task CramToBamTask\n      task
ValidateSamFile\n",
  "tags": {},
  "type": "PRIVATE"
}

```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [GetWorkflow](#) di Referensi AWS CLI Perintah.

## list-annotation-import-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-annotation-import-jobs`.

### AWS CLI

Untuk mendapatkan daftar pekerjaan impor anotasi

Berikut ini `list-annotation-import-jobs` mendapatkan daftar pekerjaan impor anotasi.

```
aws omics list-annotation-import-jobs
```

Output:

```

{
  "annotationImportJobs": [
    {
      "creationTime": "2022-11-30T01:39:41.478294Z",
      "destinationName": "gff_ann_store",
      "id": "18a9e792-xmpl-4869-a105-e5b602900444",
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
      "runLeftNormalization": false,
      "status": "COMPLETED",
      "updateTime": "2022-11-30T01:47:09.145178Z"
    },
    {
      "creationTime": "2022-11-30T00:45:58.007838Z",
      "destinationName": "my_ann_store",

```

```

        "id": "4e9eafc8-xmpl-431e-a0b2-3bda27cb600a",
        "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
        "runLeftNormalization": false,
        "status": "FAILED",
        "updateTime": "2022-11-30T00:47:01.706325Z"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListAnnotationImportJobs](#) di Referensi AWS CLI Perintah.

## list-annotation-store-versions

Contoh kode berikut menunjukkan cara menggunakan `list-annotation-store-versions`.

### AWS CLI

Untuk membuat daftar semua versi toko anotasi.

`list-annotation-store-versions` Contoh berikut mencantumkan semua versi yang ada dari toko anotasi.

```

aws omics list-annotation-store-versions \
  --name my_annotation_store

```

### Output:

```

{
  "annotationStoreVersions": [
    {
      "storeId": "4934045d1c6d",
      "id": "2a3f4a44aa7b",
      "status": "CREATING",
      "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_2",
      "name": "my_annotation_store",
      "versionName": "my_version_2",
      "creationTime": "2023-07-21T17:20:59.380043+00:00",
      "versionSizeBytes": 0
    }
  ]
}

```

```
},
{
  "storeId": "4934045d1c6d",
  "id": "4934045d1c6d",
  "status": "ACTIVE",
  "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_1",
  "name": "my_annotation_store",
  "versionName": "my_version_1",
  "creationTime": "2023-07-21T17:15:49.251040+00:00",
  "updateTime": "2023-07-21T17:15:56.434223+00:00",
  "statusMessage": "",
  "versionSizeBytes": 0
}
}
```

Untuk informasi selengkapnya, lihat [Membuat versi baru penyimpanan anotasi](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [ListAnnotationStoreVersions](#) di Referensi AWS CLI Perintah.

## list-annotation-stores

Contoh kode berikut menunjukkan cara menggunakan `list-annotation-stores`.

### AWS CLI

Untuk mendapatkan daftar toko anotasi

`list-annotation-stores` Contoh berikut mendapat daftar toko anotasi.

```
aws omics list-annotation-stores
```

Output:

```
{
  "annotationStores": [
    {
      "creationTime": "2022-11-23T22:48:39.226492Z",
      "id": "0a91xmplc71f",
      "name": "my_ann_store",
      "reference": {
```

```

        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "ACTIVE",
    "statusMessage": "",
    "storeArn": "arn:aws:omics:us-west-2:123456789012:annotationStore/
my_ann_store",
    "storeFormat": "VCF",
    "storeSizeBytes": 0,
    "updateTime": "2022-11-23T22:53:27.372840Z"
}
]
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListAnnotationStores](#) di Referensi AWS CLI Perintah.

## list-multipart-read-set-uploads

Contoh kode berikut menunjukkan cara menggunakan `list-multipart-read-set-uploads`.

### AWS CLI

Untuk mencantumkan semua unggahan set baca multibagian dan statusnya.

`list-multipart-read-set-uploads` Contoh berikut mencantumkan semua unggahan set baca multibagian dan statusnya.

```

aws omics list-multipart-read-set-uploads \
  --sequence-store-id 0123456789

```

### Output:

```

{
  "uploads":
    [
      {
        "sequenceStoreId": "0123456789",
        "uploadId": "8749584421",
        "sourceFileType": "FASTQ",
        "subjectId": "mySubject",

```

```

    "sampleId": "mySample",
    "generatedFrom": "1000 Genomes",
    "name": "HG00146",
    "description": "FASTQ for HG00146",
    "creationTime": "2023-11-29T19:22:51.349298+00:00"
  },
  {
    "sequenceStoreId": "0123456789",
    "uploadId": "5290538638",
    "sourceFileType": "BAM",
    "subjectId": "mySubject",
    "sampleId": "mySample",
    "generatedFrom": "1000 Genomes",
    "referenceArn": "arn:aws:omics:us-
west-2:845448930428:referenceStore/8168613728/reference/2190697383",
    "name": "HG00146",
    "description": "BAM for HG00146",
    "creationTime": "2023-11-29T19:23:33.116516+00:00"
  },
  {
    "sequenceStoreId": "0123456789",
    "uploadId": "4174220862",
    "sourceFileType": "BAM",
    "subjectId": "mySubject",
    "sampleId": "mySample",
    "generatedFrom": "1000 Genomes",
    "referenceArn": "arn:aws:omics:us-
west-2:845448930428:referenceStore/8168613728/reference/2190697383",
    "name": "HG00147",
    "description": "BAM for HG00147",
    "creationTime": "2023-11-29T19:23:47.007866+00:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Unggahan langsung ke penyimpanan urutan](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [ListMultipartReadSetUploads](#) di Referensi AWS CLI Perintah.

## list-read-set-activation-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-read-set-activation-jobs`.

## AWS CLI

Untuk mendapatkan daftar pekerjaan aktivasi set baca

`list-read-set-activation-jobs` Contoh berikut mendapatkan daftar pekerjaan aktivasi untuk toko urutan dengan id `1234567890`.

```
aws omics list-read-set-activation-jobs \  
--sequence-store-id 1234567890
```

Output:

```
{  
  "activationJobs": [  
    {  
      "completionTime": "2022-12-06T22:33:42.828Z",  
      "creationTime": "2022-12-06T22:32:45.213Z",  
      "id": "1234567890",  
      "sequenceStoreId": "1234567890",  
      "status": "COMPLETED"  
    },  
    {  
      "creationTime": "2022-12-06T22:35:10.100Z",  
      "id": "1234567890",  
      "sequenceStoreId": "1234567890",  
      "status": "IN_PROGRESS"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReadSetActivationJobs](#) di Referensi AWS CLI Perintah.

## **list-read-set-export-jobs**

Contoh kode berikut menunjukkan cara menggunakan `list-read-set-export-jobs`.

## AWS CLI

Untuk mendapatkan daftar pekerjaan ekspor set baca

`list-read-set-export-jobs` Contoh berikut mendapatkan daftar pekerjaan ekspor untuk toko urutan dengan id1234567890.

```
aws omics list-read-set-export-jobs \
  --sequence-store-id 1234567890
```

Output:

```
{
  "exportJobs": [
    {
      "completionTime": "2022-12-06T22:39:14.491Z",
      "creationTime": "2022-12-06T22:37:18.612Z",
      "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
      "id": "1234567890",
      "sequenceStoreId": "1234567890",
      "status": "COMPLETED"
    },
    {
      "creationTime": "2022-12-06T22:38:04.871Z",
      "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
      "id": "1234567890",
      "sequenceStoreId": "1234567890",
      "status": "IN_PROGRESS"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReadSetExportJobs](#) di Referensi AWS CLI Perintah.

## **list-read-set-import-jobs**

Contoh kode berikut menunjukkan cara menggunakan `list-read-set-import-jobs`.

AWS CLI

Untuk mendapatkan daftar pekerjaan impor set baca

`list-read-set-import-jobs` Contoh berikut mendapatkan daftar pekerjaan impor untuk toko urutan dengan id1234567890.

```
aws omics list-read-set-import-jobs \  
--sequence-store-id 1234567890
```

Output:

```
{  
  "importJobs": [  
    {  
      "completionTime": "2022-11-29T18:17:49.244Z",  
      "creationTime": "2022-11-29T17:32:47.700Z",  
      "id": "1234567890",  
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
      "sequenceStoreId": "1234567890",  
      "status": "COMPLETED"  
    },  
    {  
      "completionTime": "2022-11-23T22:01:34.090Z",  
      "creationTime": "2022-11-23T21:52:43.289Z",  
      "id": "1234567890",  
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
      "sequenceStoreId": "1234567890",  
      "status": "COMPLETED_WITH_FAILURES"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReadSetImportJobs](#) di Referensi AWS CLI Perintah.

## **list-read-set-upload-parts**

Contoh kode berikut menunjukkan cara menggunakan `list-read-set-upload-parts`.



## AWS CLI

Untuk mencantumkan semua bagian dalam unggahan multibagian yang diminta untuk penyimpanan urutan.

`list-read-set-upload-parts` Contoh berikut mencantumkan semua bagian dalam unggahan multipart yang diminta untuk toko urutan.

```
aws omics list-read-set-upload-parts \
  --sequence-store-id 0123456789 \
  --upload-id 1122334455 \
  --part-source SOURCE1
```

Output:

```
{
  "parts": [
    {
      "partNumber": 1,
      "partSize": 94371840,
      "file": "SOURCE1",
      "checksum":
"984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635",
      "lastUpdatedTime": "2023-02-02T20:14:47.533000+00:00"
    }
    {
      "partNumber": 2,
      "partSize": 10471840,
      "file": "SOURCE1",
      "checksum":
"984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635",
      "lastUpdatedTime": "2023-02-02T20:14:47.533000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Unggahan langsung ke penyimpanan urutan](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [ListReadSetUploadParts](#) di Referensi AWS CLI Perintah.

## list-read-sets

Contoh kode berikut menunjukkan cara menggunakan `list-read-sets`.

### AWS CLI

Untuk mendapatkan daftar set baca

`list-read-sets` Contoh berikut mendapatkan daftar set baca untuk toko urutan dengan id `1234567890`.

```
aws omics list-read-sets \  
  --sequence-store-id 1234567890
```

Output:

```
{  
  "readSets": [  
    {  
      "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/  
readSet/1234567890",  
      "creationTime": "2022-11-23T21:55:00.515Z",  
      "fileType": "FASTQ",  
      "id": "1234567890",  
      "name": "HG00146",  
      "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890",  
      "sampleId": "fastq-sample",  
      "sequenceStoreId": "1234567890",  
      "status": "ACTIVE",  
      "subjectId": "fastq-subject"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReadSets](#) di Referensi AWS CLI Perintah.

## list-reference-import-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-reference-import-jobs`.

## AWS CLI

Untuk mendapatkan daftar lowongan kerja impor referensi

`list-reference-import-jobs` Contoh berikut mendapatkan daftar pekerjaan impor referensi untuk toko referensi dengan id `1234567890`.

```
aws omics list-reference-import-jobs \  
--reference-store-id 1234567890
```

Output:

```
{  
  "importJobs": [  
    {  
      "completionTime": "2022-11-23T19:54:58.204Z",  
      "creationTime": "2022-11-23T19:53:20.729Z",  
      "id": "1234567890",  
      "referenceStoreId": "1234567890",  
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
      "status": "COMPLETED"  
    },  
    {  
      "creationTime": "2022-11-23T20:34:03.250Z",  
      "id": "1234567890",  
      "referenceStoreId": "1234567890",  
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
      "status": "IN_PROGRESS"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReferenceImportJobs](#) di Referensi AWS CLI Perintah.

## **list-reference-stores**

Contoh kode berikut menunjukkan cara menggunakan `list-reference-stores`.

## AWS CLI

Untuk mendapatkan daftar toko referensi

`list-reference-stores` Contoh berikut mendapat daftar toko referensi.

```
aws omics list-reference-stores
```

Output:

```
{
  "referenceStores": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",
      "creationTime": "2022-11-22T22:13:25.947Z",
      "id": "1234567890",
      "name": "my-ref-store"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReferenceStores](#) di Referensi AWS CLI Perintah.

## `list-references`

Contoh kode berikut menunjukkan cara menggunakan `list-references`.

## AWS CLI

Untuk mendapatkan daftar referensi

`list-references` Contoh berikut mendapatkan daftar referensi genom untuk toko referensi dengan id `1234567890`.

```
aws omics list-references \
  --reference-store-id 1234567890
```

Output:

```
{
  "references": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890",
      "creationTime": "2022-11-22T22:27:09.033Z",
      "id": "1234567890",
      "md5": "7ff134953dcca8c8997453bbb80b6b5e",
      "name": "assembly-38",
      "referenceStoreId": "1234567890",
      "status": "ACTIVE",
      "updateTime": "2022-11-22T22:27:09.033Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListReferences](#) di Referensi AWS CLI Perintah.

## list-run-groups

Contoh kode berikut menunjukkan cara menggunakan `list-run-groups`.

### AWS CLI

Untuk mendapatkan daftar grup lari

`list-run-groups` Contoh berikut mendapat daftar grup run.

```
aws omics list-run-groups
```

Output:

```
{
  "items": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
      "creationTime": "2022-12-01T00:58:42.915219Z",
      "id": "1234567",

```

```
        "maxCpus": 20,  
        "maxDuration": 600,  
        "name": "cram-convert"  
    }  
]  
}
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListRunGroups](#) di Referensi AWS CLI Perintah.

## list-run-tasks

Contoh kode berikut menunjukkan cara menggunakan `list-run-tasks`.

### AWS CLI

Untuk mendapatkan daftar tugas

`list-run-tasks` Contoh berikut mendapatkan daftar tugas untuk menjalankan alur kerja.

```
aws omics list-run-tasks \  
  --id 1234567
```

Output:

```
{  
  "items": [  
    {  
      "cpus": 1,  
      "creationTime": "2022-11-30T23:13:00.718651Z",  
      "memory": 15,  
      "name": "CramToBamTask",  
      "startTime": "2022-11-30T23:17:47.016Z",  
      "status": "COMPLETED",  
      "stopTime": "2022-11-30T23:18:21.503Z",  
      "taskId": "1234567"  
    },  
    {  
      "cpus": 1,  
      "creationTime": "2022-11-30T23:18:32.315606Z",  
      "memory": 4,  
      "name": "CramToBamTask",  
      "startTime": "2022-11-30T23:18:32.315606Z",  
      "status": "PENDING",  
      "stopTime": "2022-11-30T23:18:32.315606Z",  
      "taskId": "1234567"  
    }  
  ]  
}
```

```
        "name": "ValidateSamFile",
        "startTime": "2022-11-30T23:23:40.165Z",
        "status": "COMPLETED",
        "stopTime": "2022-11-30T23:24:14.766Z",
        "taskId": "1234567"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListRunTasks](#) di Referensi AWS CLI Perintah.

## list-runs

Contoh kode berikut menunjukkan cara menggunakan `list-runs`.

### AWS CLI

Untuk mendapatkan daftar alur kerja berjalan

`list-runs` Contoh berikut mendapatkan daftar alur kerja berjalan.

```
aws omics list-runs
```

Output:

```
{
  "items": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
      "creationTime": "2022-12-02T23:20:01.202074Z",
      "id": "1234567",
      "name": "cram-to-bam",
      "priority": 1,
      "startTime": "2022-12-02T23:29:18.115Z",
      "status": "COMPLETED",
      "stopTime": "2022-12-02T23:57:54.428812Z",
      "storageCapacity": 10,
      "workflowId": "1234567"
    },
    {
```

```
    "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
    "creationTime": "2022-12-03T00:16:57.180066Z",
    "id": "1234567",
    "name": "cram-to-bam",
    "priority": 1,
    "startTime": "2022-12-03T00:26:50.233Z",
    "status": "FAILED",
    "stopTime": "2022-12-03T00:37:21.451340Z",
    "storageCapacity": 10,
    "workflowId": "1234567"
  },
  {
    "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
    "creationTime": "2022-12-05T17:57:08.444817Z",
    "id": "1234567",
    "name": "cram-to-bam",
    "status": "STARTING",
    "workflowId": "1234567"
  }
]
```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListRuns](#) di Referensi AWS CLI Perintah.

## list-sequence-stores

Contoh kode berikut menunjukkan cara menggunakan `list-sequence-stores`.

AWS CLI

Untuk mendapatkan daftar toko urutan

`list-sequence-stores` Contoh berikut mendapat daftar toko urutan.

```
aws omics list-sequence-stores
```

Output:

```
{
  "sequenceStores": [
```



```
{
  "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890",
  "creationTime": "2022-11-23T01:24:33.629Z",
  "id": "1234567890",
  "name": "my-seq-store"
}
]
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListSequenceStores](#) di Referensi AWS CLI Perintah.

## list-shares

Contoh kode berikut menunjukkan cara menggunakan `list-shares`.

### AWS CLI

Untuk mencantumkan saham yang tersedia dari data HealthOmics analitik

`list-shares` Contoh berikut mencantumkan semua saham yang telah dibuat untuk pemilik sumber daya.

```
aws omics list-shares \
  --resource-owner SELF
```

### Output:

```
{
  "shares": [
    {
      "shareId": "595c1cbd-a008-4eca-a887-954d30c91c6e",
      "name": "myShare",
      "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_1",
      "principalSubscriber": "123456789012",
      "ownerId": "555555555555",
      "status": "PENDING"
    }
  ]
}
```

```

    "shareId": "39b65d0d-4368-4a19-9814-b0e31d73c10a",
    "name": "myShare3456",
    "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_2",
    "principalSubscriber": "123456789012",
    "ownerId": "555555555555",
    "status": "ACTIVE"
  },
  {
    "shareId": "203152f5-eef9-459d-a4e0-a691668d44ef",
    "name": "myShare4",
    "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_3",
    "principalSubscriber": "123456789012",
    "ownerId": "555555555555",
    "status": "ACTIVE"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Berbagi lintas akun](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [ListShares](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mendapatkan daftar tag

`list-tags-for-resource` Contoh berikut mendapat daftar tag untuk alur kerja dengan id1234567.

```
aws omics list-tags-for-resource \
  --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567
```

Output:

```
{
```

```
"tags": {
  "department": "analytics"
}
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya di Amazon Omics di Panduan Pengembang Amazon Omics](#).

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## list-variant-import-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-variant-import-jobs`.

### AWS CLI

Untuk mendapatkan daftar lowongan pekerjaan impor varian

`list-variant-import-jobs` Contoh berikut mendapatkan daftar pekerjaan impor varian.

```
aws omics list-variant-import-jobs
```

### Output:

```
{
  "variantImportJobs": [
    {
      "creationTime": "2022-11-23T22:47:02.514002Z",
      "destinationName": "my_var_store",
      "id": "69cb65d6-xmpl-4a4a-9025-4565794b684e",
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
      "runLeftNormalization": false,
      "status": "COMPLETED",
      "updateTime": "2022-11-23T22:49:17.976597Z"
    },
    {
      "creationTime": "2022-11-23T22:42:50.037812Z",
      "destinationName": "my_var_store",
      "id": "edd7b8ce-xmpl-47e2-bc99-258cac95a508",
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",

```

```

        "runLeftNormalization": false,
        "status": "COMPLETED",
        "updateTime": "2022-11-23T22:45:26.009880Z"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListVariantImportJobs](#) di Referensi AWS CLI Perintah.

## list-variant-stores

Contoh kode berikut menunjukkan cara menggunakan `list-variant-stores`.

### AWS CLI

Untuk mendapatkan daftar toko varian

`list-variant-stores` Contoh berikut mendapat daftar toko varian.

```
aws omics list-variant-stores
```

Output:

```

{
  "variantStores": [
    {
      "creationTime": "2022-11-23T22:09:07.534499Z",
      "id": "02dexmplcfdd",
      "name": "my_var_store",
      "reference": {
        "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890"
      },
      "status": "CREATING",
      "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/my_var_store",
      "storeSizeBytes": 0,
      "updateTime": "2022-11-23T22:09:24.931711Z"
    },
    {

```

```

        "creationTime": "2022-09-23T23:00:09.140265Z",
        "id": "8777xmpl1a24",
        "name": "myvstore0",
        "status": "ACTIVE",
        "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/
myvstore0",
        "storeSizeBytes": 0,
        "updateTime": "2022-09-23T23:03:26.013220Z"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListVariantStores](#) di Referensi AWS CLI Perintah.

## list-workflows

Contoh kode berikut menunjukkan cara menggunakan `list-workflows`.

### AWS CLI

Untuk mendapatkan daftar alur kerja

`list-workflows` Contoh berikut mendapat daftar alur kerja.

```
aws omics list-workflows
```

Output:

```

{
  "items": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
      "creationTime": "2022-09-23T23:08:22.041227Z",
      "digest": "nSCNo/qMWFxmplXpUdokXJnwgne0axyyc2Y0xVxrJTE=",
      "id": "1234567",
      "name": "my-wkflow-0",
      "status": "ACTIVE",
      "type": "PRIVATE"
    },
    {

```

```

        "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
        "creationTime": "2022-11-30T22:33:16.225368Z",
        "digest":
    "sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
        "id": "1234567",
        "name": "cram-converter",
        "status": "ACTIVE",
        "type": "PRIVATE"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [ListWorkflows](#) di Referensi AWS CLI Perintah.

### start-annotation-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-annotation-import-job`.

#### AWS CLI

Untuk mengimpor anotasi

`start-annotation-import-job` Contoh berikut mengimpor anotasi dari Amazon S3.

```

aws omics start-annotation-import-job \
  --destination-name tsv_ann_store \
  --no-run-left-normalization \
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
  --items source=s3://omics-artifacts-01d6xmpl4e72dd32/targetedregions.bed.gz

```

Output:

```

{
  "jobId": "984162c7-xmpl-4d23-ab47-286f7950bfbf"
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartAnnotationImportJob](#) di Referensi AWS CLI Perintah.

## start-read-set-activation-job

Contoh kode berikut menunjukkan cara menggunakan `start-read-set-activation-job`.

### AWS CLI

Untuk mengaktifkan set baca yang diarsipkan

`start-read-set-activation-job` Contoh berikut mengaktifkan dua set baca.

```
aws omics start-read-set-activation-job \  
  --sequence-store-id 1234567890 \  
  --sources readSetId=1234567890 readSetId=1234567890
```

Output:

```
{  
  "creationTime": "2022-12-06T22:35:10.100Z",  
  "id": "1234567890",  
  "sequenceStoreId": "1234567890",  
  "status": "SUBMITTED"  
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartReadSetActivationJob](#) di Referensi AWS CLI Perintah.

## start-read-set-export-job

Contoh kode berikut menunjukkan cara menggunakan `start-read-set-export-job`.

### AWS CLI

Untuk mengekspor set baca

`start-read-set-export-job` Contoh berikut mengekspor dua set baca ke Amazon S3.

```
aws omics start-read-set-export-job \  
  --sequence-store-id 1234567890 \  
  --sources readSetId=1234567890 readSetId=1234567890 \  
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ
```

```
\
--destination s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/
```

### Output:

```
{
  "creationTime": "2022-12-06T22:37:18.612Z",
  "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
  "id": "1234567890",
  "sequenceStoreId": "1234567890",
  "status": "SUBMITTED"
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartReadSetExportJob](#) di Referensi AWS CLI Perintah.

## start-read-set-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-read-set-import-job`.

### AWS CLI

Untuk mengimpor set baca

`start-read-set-import-job` Contoh berikut mengimpor set baca.

```
aws omics start-read-set-import-job \
  --sequence-store-id 1234567890 \
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
  --sources file://readset-sources.json
```

`readset-sources.json` adalah dokumen JSON dengan konten berikut.

```
[
  {
    "sourceFiles":
    {
      "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam"
```



```

    },
    "sourceFileType": "BAM",
    "subjectId": "bam-subject",
    "sampleId": "bam-sample",
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
    "name": "HG00100"
  }
]

```

### Output:

```

{
  "creationTime": "2022-11-23T01:36:38.158Z",
  "id": "1234567890",
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "sequenceStoreId": "1234567890",
  "status": "SUBMITTED"
}

```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartReadSetImportJob](#) di Referensi AWS CLI Perintah.

## start-reference-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-reference-import-job`.

### AWS CLI

Untuk mengimpor genom referensi

`start-reference-import-job` Contoh berikut mengimpor genom referensi dari Amazon S3.

```

aws omics start-reference-import-job \
  --reference-store-id 1234567890 \
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
  --sources sourceFile=s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta,name=assembly-38

```

**Output:**

```
{
  "creationTime": "2022-11-22T22:25:41.124Z",
  "id": "1234567890",
  "referenceStoreId": "1234567890",
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "status": "SUBMITTED"
}
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartReferenceImportJob](#) di Referensi AWS CLI Perintah.

**start-run**

Contoh kode berikut menunjukkan cara menggunakan `start-run`.

## AWS CLI

Untuk menjalankan alur kerja

`start-run` Contoh berikut menjalankan alur kerja dengan ID1234567.

```
aws omics start-run \
  --workflow-id 1234567 \
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
  --name 'cram-to-bam' \
  --output-uri s3://omics-artifacts-01d6xmpl4e72dd32/workflow-output/ \
  --run-group-id 1234567 \
  --priority 1 \
  --storage-capacity 10 \
  --log-level ALL \
  --parameters file://workflow-inputs.json
```

`workflow-inputs.json` adalah dokumen JSON dengan konten berikut.

```
{
```

```

    "sample_name": "NA12878",
    "input_cram": "s3://omics-artifacts-01d6xmpl4e72dd32/NA12878.cram",
    "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.dict",
    "ref_fasta": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta",
    "ref_fasta_index": "omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta.fai"
}

```

### Output:

```

{
  "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
  "id": "1234567",
  "status": "PENDING",
  "tags": {}
}

```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

Untuk memuat file sumber dari Amazon Omics

Anda juga dapat memuat file sumber dari penyimpanan Amazon Omics, dengan menggunakan URI khusus layanan. Contoh file workflow-inputs.json berikut menggunakan URI Amazon Omics untuk kumpulan baca dan sumber genom referensi.

```

{
  "sample_name": "NA12878",
  "input_cram": "omics://123456789012.storage.us-west-2.amazonaws.com/1234567890/
readSet/1234567890/source1",
  "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.dict",
  "ref_fasta": "omics://123456789012.storage.us-west-2.amazonaws.com/1234567890/
reference/1234567890",
  "ref_fasta_index": "omics://123456789012.storage.us-
west-2.amazonaws.com/1234567890/reference/1234567890/index"
}

```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartRun](#) di Referensi AWS CLI Perintah.

## start-variant-import-job

Contoh kode berikut menunjukkan cara menggunakan `start-variant-import-job`.

### AWS CLI

Untuk mengimpor file varian

`start-variant-import-job` Contoh berikut mengimpor file varian format VCF.

```
aws omics start-variant-import-job \  
  --destination-name my_var_store \  
  --no-run-left-normalization \  
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ \  
  --items source=s3://omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.known_indels.vcf.gz
```

Output:

```
{  
  "jobId": "edd7b8ce-xmpl-47e2-bc99-258cac95a508"  
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [StartVariantImportJob](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan department tag ke alur kerja dengan id1234567.

```
aws omics tag-resource \  
  --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567 \  
  --tags department=analytics
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya di Amazon Omics di Panduan Pengembang Amazon Omics](#).

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus `department` tag dari alur kerja.

```
aws omics untag-resource \
  --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567 \
  --tag-keys department
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-annotation-store

Contoh kode berikut menunjukkan cara menggunakan `update-annotation-store`.

### AWS CLI

Untuk memperbarui toko anotasi

`update-annotation-store` Contoh berikut memperbarui deskripsi toko anotasi bernama `my_vcf_store`.

```
aws omics update-annotation-store \
  --name my_vcf_store \
  --description "VCF annotation store"
```

Output:

```
{
```

```

    "creationTime": "2022-12-05T18:00:56.101860Z",
    "description": "VCF annotation store",
    "id": "bd6axmpl2444",
    "name": "my_vcf_store",
    "reference": {
      "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "ACTIVE",
    "storeFormat": "VCF",
    "updateTime": "2022-12-05T18:13:16.100051Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [UpdateAnnotationStore](#) di Referensi AWS CLI Perintah.

## update-run-group

Contoh kode berikut menunjukkan cara menggunakan `update-run-group`.

### AWS CLI

Untuk memperbarui grup run

`update-run-group` Contoh berikut memperbarui pengaturan grup run dengan id1234567.

```

aws omics update-run-group \
  --id 1234567 \
  --max-cpus 10

```

Output:

```

{
  "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
  "creationTime": "2022-12-01T00:58:42.915219Z",
  "id": "1234567",
  "maxCpus": 10,
  "maxDuration": 600,
  "name": "cram-convert",
  "tags": {}
}

```

Untuk informasi selengkapnya, lihat [Alur Kerja Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [UpdateRunGroup](#) di Referensi AWS CLI Perintah.

## update-variant-store

Contoh kode berikut menunjukkan cara menggunakan `update-variant-store`.

### AWS CLI

Untuk memperbarui toko varian

Contoh berikut memperbarui deskripsi toko varian bernama `my_var_store`.

```
aws omics update-variant-store \  
  --name my_var_store \  
  --description "variant store"
```

Output:

```
{  
  "creationTime": "2022-11-23T22:09:07.534499Z",  
  "description": "variant store",  
  "id": "02dexmplcfdd",  
  "name": "my_var_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
  },  
  "status": "ACTIVE",  
  "updateTime": "2022-12-05T18:23:37.686402Z"  
}
```

Untuk informasi selengkapnya, lihat [Omics Analytics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [UpdateVariantStore](#) di Referensi AWS CLI Perintah.

## update-workflow

Contoh kode berikut menunjukkan cara menggunakan `update-workflow`.

## AWS CLI

Untuk memperbarui alur kerja

`update-workflow` Contoh berikut memperbarui deskripsi alur kerja dengan ID1234567.

```
aws omics update-workflow \  
  --id 1234567 \  
  --description "copy workflow"
```

Untuk informasi selengkapnya, lihat [Penyimpanan Omics](#) di Panduan Pengembang Amazon Omics.

- Untuk detail API, lihat [UpdateWorkflow](#) di Referensi AWS CLI Perintah.

## `upload-read-set-part`

Contoh kode berikut menunjukkan cara menggunakan `upload-read-set-part`.

## AWS CLI

Untuk mengunggah bagian set baca.

`upload-read-set-part` Contoh berikut mengunggah bagian tertentu dari set baca.

```
aws omics upload-read-set-part \  
  --sequence-store-id 0123456789 \  
  --upload-id 1122334455 \  
  --part-source SOURCE1 \  
  --part-number 1 \  
  --payload /path/to/file/read_1_part_1.fastq.gz
```

Output:

```
{  
  "checksum": "984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635"  
}
```

Untuk informasi selengkapnya, lihat [Unggahan langsung ke penyimpanan urutan](#) di Panduan AWS HealthOmics Pengguna.

- Untuk detail API, lihat [UploadReadSetPart](#) di Referensi AWS CLI Perintah.



## Contoh IAM menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with IAM.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-client-id-to-open-id-connect-provider**

Contoh kode berikut menunjukkan cara menggunakan `add-client-id-to-open-id-connect-provider`.

AWS CLI

Untuk menambahkan ID klien (audiens) ke penyedia Open-ID Connect (OIDC)

`add-client-id-to-open-id-connect-provider` Perintah berikut menambahkan ID klien `my-application-ID` ke penyedia OIDC bernama `server.example.com`

```
aws iam add-client-id-to-open-id-connect-provider \
  --client-id my-application-ID \
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com
```

Perintah ini tidak menghasilkan output.

Untuk membuat penyedia OIDC, gunakan perintah `create-open-id-connect-provider`

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [AddClientIdToOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## add-role-to-instance-profile

Contoh kode berikut menunjukkan cara menggunakan `add-role-to-instance-profile`.

### AWS CLI

Untuk menambahkan peran ke profil instance

`add-role-to-instance-profile` Perintah berikut menambahkan peran bernama `S3Access` ke profil instance bernama `Webserver`.

```
aws iam add-role-to-instance-profile \  
  --role-name S3Access \  
  --instance-profile-name Webserver
```

Perintah ini tidak menghasilkan output.

Untuk membuat profil instance, gunakan `create-instance-profile` perintah.

Untuk informasi selengkapnya, lihat [Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan dalam instans Amazon EC2](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [AddRoleToInstanceProfile](#) di Referensi AWS CLI Perintah.

## add-user-to-group

Contoh kode berikut menunjukkan cara menggunakan `add-user-to-group`.

### AWS CLI

Untuk menambahkan pengguna ke grup IAM

`add-user-to-group` Perintah berikut menambahkan pengguna IAM bernama Bob ke grup IAM bernama Admins

```
aws iam add-user-to-group \  
  --user-name Bob \  
  --group-name Admins
```

```
--group-name Admins
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus pengguna dalam grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [AddUserToGroup](#) di Referensi AWS CLI Perintah.

## **attach-group-policy**

Contoh kode berikut menunjukkan cara menggunakan `attach-group-policy`.

### AWS CLI

Untuk melampirkan kebijakan terkelola ke grup IAM

`attach-group-policy` Perintah berikut melampirkan kebijakan AWS terkelola bernama `ReadOnlyAccess` ke grup IAM bernama `Finance`

```
aws iam attach-group-policy \  
  --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \  
  --group-name Finance
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan terkelola dan kebijakan sebaris](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [AttachGroupPolicy](#) di Referensi AWS CLI Perintah.

## **attach-role-policy**

Contoh kode berikut menunjukkan cara menggunakan `attach-role-policy`.

### AWS CLI

Untuk melampirkan kebijakan terkelola ke peran IAM

`attach-role-policy` Perintah berikut melampirkan kebijakan AWS terkelola bernama `ReadOnlyAccess` ke peran IAM bernama `ReadOnlyRole`

```
aws iam attach-role-policy \  
  --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \  
  --role-name ReadOnlyRole
```

```
--policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \  
--role-name ReadOnlyRole
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan terkelola dan kebijakan sebaris](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [AttachRolePolicy](#) di Referensi AWS CLI Perintah.

## **attach-user-policy**

Contoh kode berikut menunjukkan cara menggunakan `attach-user-policy`.

### AWS CLI

Untuk melampirkan kebijakan terkelola ke pengguna IAM

`attach-user-policy` Perintah berikut melampirkan kebijakan AWS terkelola bernama `AdministratorAccess` ke pengguna IAM bernama `Alice`

```
aws iam attach-user-policy \  
  --policy-arn arn:aws:iam::aws:policy/AdministratorAccess \  
  --user-name Alice
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan terkelola dan kebijakan sebaris](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [AttachUserPolicy](#) di Referensi AWS CLI Perintah.

## **change-password**

Contoh kode berikut menunjukkan cara menggunakan `change-password`.

### AWS CLI

Untuk mengubah kata sandi untuk pengguna IAM Anda

Untuk mengubah kata sandi untuk pengguna IAM Anda, sebaiknya gunakan `--cli-input-json` parameter untuk meneruskan file JSON yang berisi kata sandi lama dan baru Anda.

Dengan menggunakan metode ini, Anda dapat menggunakan kata sandi yang kuat dengan karakter non-alfanumerik. Mungkin sulit untuk menggunakan kata sandi dengan karakter non-alfanumerik ketika Anda meneruskannya sebagai parameter baris perintah. Untuk menggunakan `--cli-input-json` parameter, mulailah dengan menggunakan `change-password` perintah dengan `--generate-cli-skeleton` parameter, seperti pada contoh berikut.

```
aws iam change-password \  
  --generate-cli-skeleton > change-password.json
```

Perintah sebelumnya membuat file JSON bernama `change-password.json` yang dapat Anda gunakan untuk mengisi kata sandi lama dan baru Anda. Misalnya, file mungkin terlihat seperti berikut ini.

```
{  
  "OldPassword": "3s0K_;xh4~8XXI",  
  "NewPassword": "]35d/{pB9Fo9wJ"  
}
```

Selanjutnya, untuk mengubah kata sandi Anda, gunakan `change-password` perintah lagi, kali ini melewati `--cli-input-json` parameter untuk menentukan file JSON Anda. `change-password` Perintah berikut menggunakan `--cli-input-json` parameter dengan file JSON yang disebut `change-password.json`.

```
aws iam change-password \  
  --cli-input-json file://change-password.json
```

Perintah ini tidak menghasilkan output.

Perintah ini hanya dapat dipanggil oleh pengguna IAM. Jika perintah ini disebut menggunakan kredensi AWS akun (root), perintah mengembalikan kesalahan `InvalidUserType`.

Untuk informasi selengkapnya, lihat [Cara pengguna IAM mengubah kata sandi mereka sendiri](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ChangePassword](#) di Referensi AWS CLI Perintah.

## create-access-key

Contoh kode berikut menunjukkan cara menggunakan `create-access-key`.

## AWS CLI

Untuk membuat kunci akses untuk pengguna IAM

`create-access-key`Perintah berikut membuat kunci akses (ID kunci akses dan kunci akses rahasia) untuk pengguna IAM bernama Bob.

```
aws iam create-access-key \  
  --user-name Bob
```

Output:

```
{  
  "AccessKey": {  
    "UserName": "Bob",  
    "Status": "Active",  
    "CreateDate": "2015-03-09T18:39:23.411Z",  
    "SecretAccessKey": "wJa1rXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY",  
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE"  
  }  
}
```

Simpan kunci akses rahasia di lokasi yang aman. Jika hilang, itu tidak dapat dipulihkan, dan Anda harus membuat kunci akses baru.

Untuk informasi lebih lanjut, lihat [Mengelola access key untuk pengguna IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [CreateAccessKey](#) di Referensi AWS CLI Perintah.

## **create-account-alias**

Contoh kode berikut menunjukkan cara menggunakan `create-account-alias`.

## AWS CLI

Untuk membuat alias akun

`create-account-alias`Perintah berikut membuat alias `examplecorp` untuk AWS akun Anda.

```
aws iam create-account-alias \  
  --account-alias-name examplecorp
```

```
--account-alias examplecorp
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ID AWS akun Anda dan aliasnya](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateAccountAlias](#) di Referensi AWS CLI Perintah.

## create-group

Contoh kode berikut menunjukkan cara menggunakan `create-group`.

### AWS CLI

Untuk membuat grup IAM

`create-group` Perintah berikut membuat grup IAM bernama `Admins`.

```
aws iam create-group \  
  --group-name Admins
```

Output:

```
{  
  "Group": {  
    "Path": "/",  
    "CreateDate": "2015-03-09T20:30:24.940Z",  
    "GroupId": "AIDGPMS9R04H3FEXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:group/Admins",  
    "GroupName": "Admins"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateGroup](#) di Referensi AWS CLI Perintah.

## create-instance-profile

Contoh kode berikut menunjukkan cara menggunakan `create-instance-profile`.

## AWS CLI

Untuk membuat profil instance

`create-instance-profile` Perintah berikut menciptakan sebuah contoh profil bernama `Webserver`.

```
aws iam create-instance-profile \  
  --instance-profile-name Webserver
```

Output:

```
{  
  "InstanceProfile": {  
    "InstanceId": "AIPAJMBC7DLSPEXAMPLE",  
    "Roles": [],  
    "CreateDate": "2015-03-09T20:33:19.626Z",  
    "InstanceProfileName": "Webserver",  
    "Path": "/",  
    "Arn": "arn:aws:iam::123456789012:instance-profile/Webserver"  
  }  
}
```

Untuk menambahkan peran ke profil instance, gunakan `add-role-to-instance-profile` perintah.

Untuk informasi selengkapnya, lihat [Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan dalam instans Amazon EC2](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [CreateInstanceProfile](#) di Referensi AWS CLI Perintah.

## **create-login-profile**

Contoh kode berikut menunjukkan cara menggunakan `create-login-profile`.

## AWS CLI

Untuk membuat kata sandi untuk pengguna IAM

Untuk membuat kata sandi untuk pengguna IAM, sebaiknya gunakan `--cli-input-json` parameter untuk meneruskan file JSON yang berisi kata sandi. Dengan menggunakan metode ini, Anda dapat membuat kata sandi yang kuat dengan karakter non-alfanumerik. Mungkin sulit



untuk membuat kata sandi dengan karakter non-alfanumerik ketika Anda meneruskannya sebagai parameter baris perintah.

Untuk menggunakan `--cli-input-json` parameter, mulailah dengan menggunakan `create-login-profile` perintah dengan `--generate-cli-skeleton` parameter, seperti pada contoh berikut.

```
aws iam create-login-profile \  
  --generate-cli-skeleton > create-login-profile.json
```

Perintah sebelumnya membuat file JSON yang disebut `create-login-profile.json` yang dapat Anda gunakan untuk mengisi informasi untuk perintah berikutnya. `create-login-profile` Sebagai contoh:

```
{  
  "UserName": "Bob",  
  "Password": "&1-3a6u:RA0djs",  
  "PasswordResetRequired": true  
}
```

Selanjutnya, untuk membuat kata sandi untuk pengguna IAM, gunakan `create-login-profile` perintah lagi, kali ini melewati `--cli-input-json` parameter untuk menentukan file JSON Anda. `create-login-profile` Perintah berikut menggunakan `--cli-input-json` parameter dengan file JSON yang disebut `create-login-profile.json`.

```
aws iam create-login-profile \  
  --cli-input-json file://create-login-profile.json
```

Output:

```
{  
  "LoginProfile": {  
    "UserName": "Bob",  
    "CreateDate": "2015-03-10T20:55:40.274Z",  
    "PasswordResetRequired": true  
  }  
}
```

Jika kata sandi baru melanggar kebijakan kata sandi akun, perintah mengembalikan `PasswordPolicyViolation` kesalahan.

Untuk mengubah kata sandi untuk pengguna yang sudah memilikinya, gunakan `update-login-profile`. Untuk menetapkan kebijakan kata sandi untuk akun, gunakan `update-account-password-policy` perintah.

Jika kebijakan kata sandi akun memungkinkan mereka, pengguna IAM dapat mengubah kata sandi mereka sendiri menggunakan `change-password` perintah.

Untuk informasi selengkapnya, lihat [Mengelola kata sandi untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateLoginProfile](#) di Referensi AWS CLI Perintah.

## **create-open-id-connect-provider**

Contoh kode berikut menunjukkan cara menggunakan `create-open-id-connect-provider`.

### AWS CLI

Untuk membuat penyedia OpenID Connect (OIDC)

Untuk membuat penyedia OpenID Connect (OIDC), sebaiknya gunakan `--cli-input-json` parameter untuk meneruskan file JSON yang berisi parameter yang diperlukan. Saat Anda membuat penyedia OIDC, Anda harus meneruskan URL penyedia, dan URL harus dimulai. `https://` Mungkin sulit untuk meneruskan URL sebagai parameter baris perintah, karena karakter titik dua (`:`) dan garis miring maju (`/`) memiliki arti khusus di beberapa lingkungan baris perintah. Menggunakan `--cli-input-json` parameter mengatasi batasan ini.

Untuk menggunakan `--cli-input-json` parameter, mulailah dengan menggunakan `create-open-id-connect-provider` perintah dengan `--generate-cli-skeleton` parameter, seperti pada contoh berikut.

```
aws iam create-open-id-connect-provider \
  --generate-cli-skeleton > create-open-id-connect-provider.json
```

Perintah sebelumnya membuat file JSON bernama `create-open-id-connect-provider.json` yang dapat Anda gunakan untuk mengisi informasi untuk perintah berikutnya. `create-open-id-connect-provider` Sebagai contoh:

```
{
  "Url": "https://server.example.com",
```

```
"ClientIDList": [  
  "example-application-ID"  
],  
"ThumbprintList": [  
  "c3768084dfb3d2b68b7897bf5f565da8eEXAMPLE"  
]  
}
```

Selanjutnya, untuk membuat penyedia OpenID Connect (OIDC), gunakan `create-open-id-connect-provider` perintah lagi, kali ini melewati `--cli-input-json` parameter untuk menentukan file JSON Anda. `create-open-id-connect-provider` Perintah berikut menggunakan `--cli-input-json` parameter dengan file JSON bernama `create-open-id-connect-provider.json`.

```
aws iam create-open-id-connect-provider \  
  --cli-input-json file://create-open-id-connect-provider.json
```

Output:

```
{  
  "OpenIDConnectProviderArn": "arn:aws:iam::123456789012:oidc-provider/  
server.example.com"  
}
```

Untuk informasi selengkapnya tentang penyedia OIDC, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\) di Panduan Pengguna IAM.AWS](#)

Untuk informasi selengkapnya tentang mendapatkan cap jempol untuk penyedia OIDC, lihat [Memperoleh cap jempol untuk Penyedia Identitas OpenID Connect](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [CreateOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## **create-policy-version**

Contoh kode berikut menunjukkan cara menggunakan `create-policy-version`.

### AWS CLI

Untuk membuat versi baru dari kebijakan terkelola

Contoh ini membuat v2 versi baru dari kebijakan IAM yang `arn:aws:iam::123456789012:policy/MyPolicy` ARN-nya dan menjadikannya versi default.

```
aws iam create-policy-version \  
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \  
  --policy-document file://NewPolicyVersion.json \  
  --set-as-default
```

Output:

```
{  
  "PolicyVersion": {  
    "CreateDate": "2015-06-16T18:56:03.721Z",  
    "VersionId": "v2",  
    "IsDefaultVersion": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Kebijakan IAM Versi di Panduan](#) Pengguna AWS IAM.

- Untuk detail API, lihat [CreatePolicyVersion](#) di Referensi AWS CLI Perintah.

## create-policy

Contoh kode berikut menunjukkan cara menggunakan `create-policy`.

### AWS CLI

Contoh 1: Untuk membuat kebijakan terkelola pelanggan

Perintah berikut membuat kebijakan terkelola pelanggan bernama `my-policy`.

```
aws iam create-policy \  
  --policy-name my-policy \  
  --policy-document file://policy
```

File tersebut `policy` adalah dokumen JSON di folder saat ini yang memberikan akses baca saja ke `shared` folder dalam bucket Amazon S3 bernama `my-bucket`

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "s3:Get*",
      "s3:List*"
    ],
    "Resource": [
      "arn:aws:s3:::my-bucket/shared/*"
    ]
  }
]
```

### Output:

```
{
  "Policy": {
    "PolicyName": "my-policy",
    "CreateDate": "2015-06-01T19:31:18.620Z",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
    "DefaultVersionId": "v1",
    "Path": "/",
    "Arn": "arn:aws:iam::0123456789012:policy/my-policy",
    "UpdateDate": "2015-06-01T19:31:18.620Z"
  }
}
```

Untuk informasi lebih lanjut tentang menggunakan file sebagai input untuk parameter string, lihat [Menentukan nilai parameter untuk AWS CLI di Panduan Pengguna AWS CLI](#).

Contoh 2: Untuk membuat kebijakan yang dikelola pelanggan dengan deskripsi

Perintah berikut membuat kebijakan terkelola pelanggan bernama `my-policy` dengan deskripsi yang tidak dapat diubah:

```
aws iam create-policy \
  --policy-name my-policy \
  --policy-document file://policy.json \
```

```
--description "This policy grants access to all Put, Get, and List actions for my-bucket"
```

File tersebut `policy.json` adalah dokumen JSON di folder saat ini yang memberikan akses ke semua tindakan Put, List, dan Get untuk bucket Amazon S3 bernama `my-bucket`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket*",
        "s3:PutBucket*",
        "s3:GetBucket*"
      ],
      "Resource": [
        "arn:aws:s3:::my-bucket"
      ]
    }
  ]
}
```

Output:

```
{
  "Policy": {
    "PolicyName": "my-policy",
    "PolicyId": "ANPAWGSUGIDPEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:policy/my-policy",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "PermissionsBoundaryUsageCount": 0,
    "IsAttachable": true,
    "CreateDate": "2023-05-24T22:38:47+00:00",
    "UpdateDate": "2023-05-24T22:38:47+00:00"
  }
}
```

Untuk informasi selengkapnya tentang Kebijakan Berbasis Identifikasi, lihat Kebijakan berbasis identitas dan kebijakan berbasis sumber daya di [Panduan Pengguna IAM.AWS](#)

### Contoh 3: Membuat kebijakan terkelola pelanggan dengan tag

Perintah berikut membuat kebijakan terkelola pelanggan bernama `my-policy` dengan tag.

Contoh ini menggunakan bendera `--tags` parameter dengan tag berformat JSON berikut:

```
'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",  
"Value": "Seattle"}' Atau, --tags bendera dapat digunakan dengan tag dalam format  
singkatan: 'Key=Department,Value=Accounting Key=Location,Value=Seattle'
```

```
aws iam create-policy \  
  --policy-name my-policy \  
  --policy-document file://policy.json \  
  --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",  
"Value": "Seattle"}'
```

File tersebut `policy.json` adalah dokumen JSON di folder saat ini yang memberikan akses ke semua tindakan Put, List, dan Get untuk bucket Amazon S3 bernama `my-bucket`

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:ListBucket*",  
        "s3:PutBucket*",  
        "s3:GetBucket*"  
      ],  
      "Resource": [  
        "arn:aws:s3:::my-bucket"  
      ]  
    }  
  ]  
}
```

### Output:

```
{  
  "Policy": {  
    "PolicyName": "my-policy",  
    "PolicyId": "ANPAWGSUGIDPEXAMPLE",  
    "Arn": "arn:aws:iam::12345678012:policy/my-policy",  
    "Path": "/",
```

```
"DefaultVersionId": "v1",
"AttachmentCount": 0,
"PermissionsBoundaryUsageCount": 0,
"IsAttachable": true,
"CreateDate": "2023-05-24T23:16:39+00:00",
"UpdateDate": "2023-05-24T23:16:39+00:00",
"Tags": [
  {
    "Key": "Department",
    "Value": "Accounting"
  },
  {
    "Key": "Location",
    "Value": "Seattle"
  }
]
```

Untuk informasi selengkapnya tentang kebijakan Penandaan, lihat [Menandai kebijakan yang dikelola pelanggan](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreatePolicy](#) di Referensi AWS CLI Perintah.

## create-role

Contoh kode berikut menunjukkan cara menggunakan `create-role`.

### AWS CLI

Contoh 1: Untuk membuat peran IAM

`create-role` Perintah berikut membuat peran bernama `Test-Role` dan melampirkan kebijakan kepercayaan padanya.

```
aws iam create-role \
  --role-name Test-Role \
  --assume-role-policy-document file://Test-Role-Trust-Policy.json
```

Output:

```
{
```



```
"Role": {
  "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
  "RoleId": "AKIAIOSFODNN7EXAMPLE",
  "CreateDate": "2013-06-07T20:43:32.821Z",
  "RoleName": "Test-Role",
  "Path": "/",
  "Arn": "arn:aws:iam::123456789012:role/Test-Role"
}
```

Kebijakan kepercayaan didefinisikan sebagai dokumen JSON dalam file `test-role-trust-policy.json`. (Nama file dan ekstensi tidak memiliki signifikansi.) Kebijakan kepercayaan harus menentukan kepala sekolah.

Untuk melampirkan kebijakan izin ke peran, gunakan `put-role-policy` perintah.

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna AWS IAM.

Contoh 2: Untuk membuat peran IAM dengan durasi sesi maksimum yang ditentukan

`create-role`Perintah berikut membuat peran bernama `Test-Role` dan menetapkan durasi sesi maksimum 7200 detik (2 jam).

```
aws iam create-role \
  --role-name Test-Role \
  --assume-role-policy-document file://Test-Role-Trust-Policy.json \
  --max-session-duration 7200
```

Output:

```
{
  "Role": {
    "Path": "/",
    "RoleName": "Test-Role",
    "RoleId": "AKIAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:role/Test-Role",
    "CreateDate": "2023-05-24T23:50:25+00:00",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Sid": "Statement1",
          "Effect": "Allow",
```

```

        "Principal": {
            "AWS": "arn:aws:iam::12345678012:root"
        },
        "Action": "sts:AssumeRole"
    }
]
}
}
}
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi durasi sesi maksimum peran \(AWS API\)](#) di Panduan Pengguna AWS IAM.

Contoh 3: Untuk membuat Peran IAM dengan tag

Perintah berikut membuat Peran IAM Test-Role dengan tag. Contoh ini menggunakan bendera `--tags` parameter dengan tag berformat JSON berikut: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'` Atau, `--tags` bendera dapat digunakan dengan tag dalam format singkatan: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`

```

aws iam create-role \
  --role-name Test-Role \
  --assume-role-policy-document file://Test-Role-Trust-Policy.json \
  --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
"Value": "Seattle"}'

```

Output:

```

{
  "Role": {
    "Path": "/",
    "RoleName": "Test-Role",
    "RoleId": "AKIAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:role/Test-Role",
    "CreateDate": "2023-05-25T23:29:41+00:00",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Sid": "Statement1",
          "Effect": "Allow",

```

```

        "Principal": {
            "AWS": "arn:aws:iam::123456789012:root"
        },
        "Action": "sts:AssumeRole"
    }
]
},
"Tags": [
    {
        "Key": "Department",
        "Value": "Accounting"
    },
    {
        "Key": "Location",
        "Value": "Seattle"
    }
]
}
}

```

Untuk informasi selengkapnya, lihat [Menandai peran IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateRole](#) di Referensi AWS CLI Perintah.

## create-saml-provider

Contoh kode berikut menunjukkan cara menggunakan `create-saml-provider`.

### AWS CLI

Untuk membuat penyedia SAFL

Contoh ini menciptakan penyedia SAFL baru di IAM bernama `MySAMLProvider` Ini dijelaskan oleh dokumen metadata SALL yang ditemukan dalam file `SAMLMetaData.xml`

```

aws iam create-saml-provider \
  --saml-metadata-document file://SAMLMetaData.xml \
  --name MySAMLProvider

```

Output:

```

{
  "SAMLProviderArn": "arn:aws:iam::123456789012:saml-provider/MySAMLProvider"
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas IAM SALL](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateSamlProvider](#) di [Referensi Perintah](#).AWS CLI

## create-service-linked-role

Contoh kode berikut menunjukkan cara menggunakan `create-service-linked-role`.

### AWS CLI

Untuk membuat peran terkait layanan

`create-service-linked-role` Contoh berikut membuat peran terkait layanan untuk AWS layanan tertentu dan melampirkan deskripsi yang ditentukan.

```
aws iam create-service-linked-role \  
  --aws-service-name lex.amazonaws.com \  
  --description "My service-linked role to support Lex"
```

### Output:

```
{  
  "Role": {  
    "Path": "/aws-service-role/lex.amazonaws.com/",  
    "RoleName": "AWSServiceRoleForLexBots",  
    "RoleId": "ARO1234567890EXAMPLE",  
    "Arn": "arn:aws:iam::1234567890:role/aws-service-role/lex.amazonaws.com/  
AWSServiceRoleForLexBots",  
    "CreateDate": "2019-04-17T20:34:14+00:00",  
    "AssumeRolePolicyDocument": {  
      "Version": "2012-10-17",  
      "Statement": [  
        {  
          "Action": [  
            "sts:AssumeRole"  
          ],  
          "Effect": "Allow",  
          "Principal": {  
            "Service": [  

```



Untuk informasi selengkapnya, lihat [Membuat kredensi Git untuk koneksi HTTPS CodeCommit](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [CreateServiceSpecificCredential](#) di Referensi AWS CLI Perintah.

## create-user

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

### AWS CLI

Contoh 1: Untuk membuat pengguna IAM

`create-user` Perintah berikut membuat pengguna IAM bernama Bob di akun saat ini.

```
aws iam create-user \  
  --user-name Bob
```

Output:

```
{  
  "User": {  
    "UserName": "Bob",  
    "Path": "/",  
    "CreateDate": "2023-06-08T03:20:41.270Z",  
    "UserId": "AIDAIOSFODNN7EXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:user/Bob"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di AWS akun Anda](#) di Panduan Pengguna AWS IAM.

Contoh 2: Untuk membuat pengguna IAM di jalur tertentu

`create-user` Perintah berikut membuat pengguna IAM bernama Bob di jalur yang ditentukan.

```
aws iam create-user \  
  --user-name Bob \  
  --path /division_abc/subdivision_xyz/
```

Output:

```
{
  "User": {
    "Path": "/division_abc/subdivision_xyz/",
    "UserName": "Bob",
    "UserId": "AIDAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:user/division_abc/subdivision_xyz/Bob",
    "CreateDate": "2023-05-24T18:20:17+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [pengenal IAM](#) di Panduan Pengguna AWS IAM.

Contoh 3: Untuk Membuat Pengguna IAM dengan tag

`create-user`Perintah berikut membuat pengguna IAM bernama Bob dengan tag. Contoh ini menggunakan bendera `--tags` parameter dengan tag berformat JSON berikut: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'` Atau, `--tags` bendera dapat digunakan dengan tag dalam format singkatan: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`

```
aws iam create-user \
  --user-name Bob \
  --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
  "Value": "Seattle"}'
```

Output:

```
{
  "User": {
    "Path": "/",
    "UserName": "Bob",
    "UserId": "AIDAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:user/Bob",
    "CreateDate": "2023-05-25T17:14:21+00:00",
    "Tags": [
      {
        "Key": "Department",
        "Value": "Accounting"
      },
      {
        "Key": "Location",
        "Value": "Seattle"
      }
    ]
  }
}
```

```
    }
  ]
}
}
```

Untuk informasi selengkapnya, lihat [Menandai pengguna IAM di Panduan Pengguna AWS IAM](#).

Contoh 3: Untuk membuat pengguna IAM dengan batas izin yang ditetapkan

`create-user`Perintah berikut membuat pengguna IAM bernama Bob dengan batas izin AmazonS3. FullAccess

```
aws iam create-user \
  --user-name Bob \
  --permissions-boundary arn:aws:iam::aws:policy/AmazonS3FullAccess
```

Output:

```
{
  "User": {
    "Path": "/",
    "UserName": "Bob",
    "UserId": "AIDAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:user/Bob",
    "CreateDate": "2023-05-24T17:50:53+00:00",
    "PermissionsBoundary": {
      "PermissionsBoundaryType": "Policy",
      "PermissionsBoundaryArn": "arn:aws:iam::aws:policy/AmazonS3FullAccess"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Batas izin untuk entitas IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## **create-virtual-mfa-device**

Contoh kode berikut menunjukkan cara menggunakan `create-virtual-mfa-device`.

### AWS CLI

Untuk membuat perangkat MFA virtual



Contoh ini menciptakan perangkat MFA virtual baru yang disebut BobsMFADevice. Ini membuat file yang berisi informasi bootstrap yang disebut QRCode.png dan menemukannya di C:/ direktori. Metode bootstrap yang digunakan dalam contoh ini adalah QRCodePNG.

```
aws iam create-virtual-mfa-device \  
  --virtual-mfa-device-name BobsMFADevice \  
  --outfile C:/QRCode.png \  
  --bootstrap-method QRCodePNG
```

Output:

```
{  
  "VirtualMFADevice": {  
    "SerialNumber": "arn:aws:iam::210987654321:mfa/BobsMFADevice"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan autentikasi multifaktor \(MFA\) dalam AWS](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [CreateVirtualMfaDevice](#) di Referensi AWS CLI Perintah.

## **deactivate-mfa-device**

Contoh kode berikut menunjukkan cara menggunakan `deactivate-mfa-device`.

### AWS CLI

Untuk menonaktifkan perangkat MFA

Perintah ini menonaktifkan perangkat MFA virtual dengan ARN

`arn:aws:iam::210987654321:mfa/BobsMFADevice` yang terkait dengan pengguna Bob

```
aws iam deactivate-mfa-device \  
  --user-name Bob \  
  --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan autentikasi multifaktor \(MFA\) dalam AWS](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeactivateMfaDevice](#) di Referensi AWS CLI Perintah.

## decode-authorization-message

Contoh kode berikut menunjukkan cara menggunakan `decode-authorization-message`.

### AWS CLI

Untuk memecahkan kode pesan kegagalan otorisasi

`decode-authorization-message` Contoh berikut menerjemahkan pesan yang dikembalikan oleh konsol EC2 saat mencoba meluncurkan instance tanpa izin yang diperlukan.

```
aws sts decode-authorization-message \
  --encoded-message lxzA8VEjEvu-s0TTt3PgYCXik9Yak0qsrFJGRZR98xNcyWAxwRq14xIvd-
  npzbgTevuufCTbjeBAaDARg9cbTK1rJbg3awM33o-Vy3ebPErE2-
  mWR9hVYdvX-0zKgV0WF9pWjZaJSMqxB-aLXo-I_8TTvBq88x8IFPbMArNdpu0IjxDjzf22PF3S0E3XvIQ-
  _PE00aUqHCCcsSrFtvxm6yQD1nbm6VTIVrfa0Bzy8lsoMo7SjIaJ2r5vph6SY5vCCwg6o2JKe3hIHTa8zRrDbZSFMkcX
  Xx9AYAAIr6bhcis7C__bZh4d1AAWooHFGKgfoJcWGWgdzgbu9hWyVvKTpeot5hsb8qANYjJRCPTKpi6PZfdijIkwb6g
```

Outputnya diformat sebagai string baris tunggal teks JSON yang dapat Anda uraikan dengan prosesor teks JSON apa pun.

```
{
  "DecodedMessage": "{\"allowed\":false,\"explicitDeny\":false,\"matchedStatements\
  \":{\\"items\":[],\"failures\":{\\"items\":[],\"context\":{\\"principal\
  \":{\\"id\":"AIDAV3ZUEFP6J7GY706L0\",\\"name\":"chain-user\",\\"arn\
  \":\"arn:aws:iam:403299380220:user/chain-user\",\\"action\":"ec2:RunInstances\",
  \\"resource\":"arn:aws:ec2:us-east-2:403299380220:instance/*\",\\"conditions\
  \":{\\"items\":[{\\"key\":"ec2:InstanceMarketType\",\\"values\":{\\"items\
  \":[{\\"value\":"on-demand\"}]}]},{\\"key\":"aws:Resource\",\\"values\":{\
  \\"items\":[{\\"value\":"instance/*\"}]}]},{\\"key\":"aws:Account\",\\"values\
  \":{\\"items\":[{\\"value\":"403299380220\"}]}]},{\\"key\":"ec2:AvailabilityZone\",
  \\"values\":{\\"items\":[{\\"value\":"us-east-2b\"}]}]},{\\"key\":"ec2:eb
  sOptimized\",\\"values\":{\\"items\":[{\\"value\":"false\"}]}]},{\\"key\
  \":\"ec2:IsLaunchTemplateResource\",\\"values\":{\\"items\":[{\\"value\
  \":\"false\"}]}]},{\\"key\":"ec2:InstanceType\",\\"values\":{\\"items\
  \":[{\\"value\":"t2.micro\"}]}]},{\\"key\":"ec2:RootDeviceType\",
  \\"values\":{\\"items\":[{\\"value\":"ebs\"}]}]},{\\"key\":"aws:Region\",
  \\"values\":{\\"items\":[{\\"value\":"us-east-2\"}]}]},{\\"key\":"aws:Service\",
  \\"values\":{\\"items\":[{\\"value\":"ec2\"}]}]},{\\"key\":"ec2:InstanceID\",
  \\"values\":{\\"items\":[{\\"value\":"*\"}]}]},{\\"key\":"aws:Type\",
  \\"values\":{\\"items\":[{\\"value\":"instance\"}]}]},{\\"key\":"ec2:Tenancy\",
  \\"values\":{\\"items\":[{\\"value\":"default\"}]}]},{\\"key\":"ec2:Region\",
  \\"values\":{\\"items\":[{\\"value\":"us-east-2\"}]}]},{\\"key\":"aws:ARN\",
  \\"values\":{\\"items\":[{\\"value\":"arn:aws:ec2:us-east-2:403299380220:instance/*\"}]}]}]}}"
```

```
}
```

Untuk informasi lebih lanjut, lihat [Bagaimana saya bisa memecahkan kode pesan kegagalan otorisasi setelah menerima kesalahan "UnauthorizedOperation" selama peluncuran instans EC2?](#) di AWS re:post.

- Untuk detail API, lihat [DecodeAuthorizationMessage](#) di Referensi AWS CLI Perintah.

## **delete-access-key**

Contoh kode berikut menunjukkan cara menggunakan `delete-access-key`.

### AWS CLI

Untuk menghapus kunci akses untuk pengguna IAM

`delete-access-key` Perintah berikut menghapus kunci akses yang ditentukan (ID kunci akses dan kunci akses rahasia) untuk pengguna IAM bernama Bob

```
aws iam delete-access-key \  
  --access-key-id AKIDPMS9R04H3FEXAMPLE \  
  --user-name Bob
```

Perintah ini tidak menghasilkan output.

Untuk membuat daftar kunci akses yang ditentukan untuk pengguna IAM, gunakan `list-access-keys` perintah.

Untuk informasi lebih lanjut, lihat [Mengelola access key untuk pengguna IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeleteAccessKey](#) di Referensi AWS CLI Perintah.

## **delete-account-alias**

Contoh kode berikut menunjukkan cara menggunakan `delete-account-alias`.

### AWS CLI

Untuk menghapus alias akun

`delete-account-alias` Perintah berikut menghapus alias mycompany untuk akun saat ini.

```
aws iam delete-account-alias \  
  --account-alias mycompany
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ID AWS akun Anda dan aliasnya](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteAccountAlias](#) di Referensi AWS CLI Perintah.

## **delete-account-password-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-account-password-policy`.

### AWS CLI

Untuk menghapus kebijakan kata sandi akun saat ini

`delete-account-password-policy` Perintah berikut menghapus kebijakan kata sandi untuk akun saat ini.

```
aws iam delete-account-password-policy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyetel kebijakan kata sandi akun untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteAccountPasswordPolicy](#) di Referensi AWS CLI Perintah.

## **delete-group-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-group-policy`.

### AWS CLI

Untuk menghapus kebijakan dari grup IAM

`delete-group-policy` Perintah berikut menghapus kebijakan yang dinamai `ExamplePolicy` dari grup bernama `Admins`.

```
aws iam delete-group-policy \  
  --group-name Admins
```

```
--group-name Admins \  
--policy-name ExamplePolicy
```

Perintah ini tidak menghasilkan output.

Untuk melihat kebijakan yang dilampirkan ke grup, gunakan `list-group-policies` perintah.

Untuk informasi selengkapnya, lihat [Mengelola kebijakan IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteGroupPolicy](#) di Referensi AWS CLI Perintah.

## **delete-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-group`.

### AWS CLI

Untuk menghapus grup IAM

`delete-group` Perintah berikut menghapus grup IAM bernama `MyTestGroup`

```
aws iam delete-group \  
--group-name MyTestGroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteGroup](#) di Referensi AWS CLI Perintah.

## **delete-instance-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-instance-profile`.

### AWS CLI

Untuk menghapus profil instance

`delete-instance-profile` Perintah berikut menghapus profil instance bernama `ExampleInstanceProfile`.

```
aws iam delete-instance-profile \  
--instance-profile-name ExampleInstanceProfile
```

```
--instance-profile-name ExampleInstanceProfile
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menggunakan profil instance](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeleteInstanceProfile](#) di Referensi AWS CLI Perintah.

## **delete-login-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-login-profile`.

### AWS CLI

Untuk menghapus kata sandi untuk pengguna IAM

`delete-login-profile` Perintah berikut menghapus kata sandi untuk pengguna IAM bernama Bob

```
aws iam delete-login-profile \  
  --user-name Bob
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola kata sandi untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteLoginProfile](#) di Referensi AWS CLI Perintah.

## **delete-open-id-connect-provider**

Contoh kode berikut menunjukkan cara menggunakan `delete-open-id-connect-provider`.

### AWS CLI

Untuk menghapus penyedia identitas OpenID Connect IAM

Contoh ini menghapus penyedia IAM OIDC yang terhubung ke penyedia `example.oidcprovider.com`

```
aws iam delete-open-id-connect-provider \  
  --open-id-connect-provider-urn   
  --open-id-connect-provider-name   
  --open-id-connect-provider-url   
  --open-id-connect-provider-client-id   
  --open-id-connect-provider-client-secret   
  --open-id-connect-provider-logout-url   
  --open-id-connect-provider-logout-client-id   
  --open-id-connect-provider-logout-client-secret   
  --open-id-connect-provider-logout-url   
  --open-id-connect-provider-logout-client-id   
  --open-id-connect-provider-logout-client-secret
```

```
--open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
example.oidcprovider.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [DeleteOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## **delete-policy-version**

Contoh kode berikut menunjukkan cara menggunakan `delete-policy-version`.

### AWS CLI

Untuk menghapus versi kebijakan terkelola

Contoh ini menghapus versi yang diidentifikasi sebagai v2 dari kebijakan yang `arn:aws:iam::123456789012:policy/MySamplePolicy` ARN-nya.

```
aws iam delete-policy-version \  
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \  
  --version-id v2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeletePolicyVersion](#) di Referensi AWS CLI Perintah.

## **delete-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-policy`.

### AWS CLI

Untuk menghapus kebijakan IAM

Contoh ini menghapus kebijakan `arn:aws:iam::123456789012:policy/MySamplePolicy` ARN-nya.

```
aws iam delete-policy \  
  --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## **delete-role-permissions-boundary**

Contoh kode berikut menunjukkan cara menggunakan `delete-role-permissions-boundary`.

### AWS CLI

Untuk menghapus batas izin dari peran IAM

`delete-role-permissions-boundary` Contoh berikut menghapus batas izin untuk peran IAM yang ditentukan. Untuk menerapkan batas izin ke peran, gunakan perintah. `put-role-permissions-boundary`

```
aws iam delete-role-permissions-boundary \  
  --role-name lambda-application-role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeleteRolePermissionsBoundary](#) di Referensi AWS CLI Perintah.

## **delete-role-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-role-policy`.

### AWS CLI

Untuk menghapus kebijakan dari peran IAM

`delete-role-policy` Perintah berikut menghapus kebijakan yang dinamai `ExamplePolicy` dari peran yang dinamai `Test-Role`.



```
aws iam delete-role-policy \  
  --role-name Test-Role \  
  --policy-name ExamplePolicy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memodifikasi peran](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteRolePolicy](#) di Referensi AWS CLI Perintah.

## **delete-role**

Contoh kode berikut menunjukkan cara menggunakan `delete-role`.

### AWS CLI

Untuk menghapus peran IAM

`delete-role` Perintah berikut menghapus peran bernama `Test-Role`.

```
aws iam delete-role \  
  --role-name Test-Role
```

Perintah ini tidak menghasilkan output.

Sebelum dapat menghapus peran, Anda harus menghapus peran tersebut dari profil instance (`remove-role-from-instance-profile`) apa pun, melepaskan kebijakan terkelola (`detach-role-policy`), dan menghapus kebijakan sebaris apa pun yang dilampirkan ke role (`delete-role-policy`).

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) dan [Menggunakan profil instans](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteRole](#) di Referensi AWS CLI Perintah.

## **delete-saml-provider**

Contoh kode berikut menunjukkan cara menggunakan `delete-saml-provider`.

### AWS CLI

Untuk menghapus penyedia SAFL

Contoh ini menghapus penyedia IAM SALL 2.0 yang ARN-nya.

```
arn:aws:iam::123456789012:saml-provider/SAMLADFSProvider
```

```
aws iam delete-saml-provider \  
--saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFSProvider
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas IAM SALL](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteSamlProvider di Referensi Perintah.AWS CLI](#)

## **delete-server-certificate**

Contoh kode berikut menunjukkan cara menggunakan `delete-server-certificate`.

### AWS CLI

Untuk menghapus sertifikat server dari AWS akun Anda

`delete-server-certificate` Perintah berikut menghapus sertifikat server yang ditentukan dari AWS akun Anda.

```
aws iam delete-server-certificate \  
--server-certificate-name myUpdatedServerCertificate
```

Perintah ini tidak menghasilkan output.

Untuk membuat daftar sertifikat server yang tersedia di AWS akun Anda, gunakan `list-server-certificates` perintah.

Untuk informasi selengkapnya, lihat [Mengelola sertifikat server di IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteServerCertificate](#) di Referensi AWS CLI Perintah.

## **delete-service-linked-role**

Contoh kode berikut menunjukkan cara menggunakan `delete-service-linked-role`.

## AWS CLI

Untuk menghapus peran terkait layanan

`delete-service-linked-role` Contoh berikut menghapus peran terkait layanan tertentu yang tidak lagi Anda perlukan. Penghapusan terjadi secara asinkron. Anda dapat memeriksa status penghapusan dan mengonfirmasi kapan dilakukan dengan menggunakan perintah. `get-service-linked-role-deletion-status`

```
aws iam delete-service-linked-role \  
  --role-name AWSServiceRoleForLexBots
```

Output:

```
{  
  "DeletionTaskId": "task/aws-service-role/lex.amazonaws.com/  
AWSServiceRoleForLexBots/1a2b3c4d-1234-abcd-7890-abcdeEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan peran terkait layanan](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeleteServiceLinkedRole](#) di Referensi AWS CLI Perintah.

## **delete-service-specific-credential**

Contoh kode berikut menunjukkan cara menggunakan `delete-service-specific-credential`.

### AWS CLI

Contoh 1: Menghapus kredensi khusus layanan untuk pengguna yang meminta

`delete-service-specific-credential` Contoh berikut menghapus kredensi khusus layanan yang ditentukan untuk pengguna yang membuat permintaan. `service-specific-credential-id` ini disediakan saat Anda membuat kredensi dan Anda dapat mengambilnya dengan menggunakan perintah. `list-service-specific-credentials`

```
aws iam delete-service-specific-credential \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Perintah ini tidak menghasilkan output.

## Contoh 2: Menghapus kredensi khusus layanan untuk pengguna tertentu

`delete-service-specific-credential` Contoh berikut menghapus kredensi khusus layanan yang ditentukan untuk pengguna yang ditentukan. `service-specific-credential-id` ini disediakan saat Anda membuat kredensi dan Anda dapat mengambilnya dengan menggunakan perintah `list-service-specific-credentials`

```
aws iam delete-service-specific-credential \  
  --user-name sofia \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat kredensi Git untuk koneksi HTTPS CodeCommit](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [DeleteServiceSpecificCredential](#) di Referensi AWS CLI Perintah.

## `delete-signing-certificate`

Contoh kode berikut menunjukkan cara menggunakan `delete-signing-certificate`.

### AWS CLI

Untuk menghapus sertifikat penandatanganan untuk pengguna IAM

`delete-signing-certificate` Perintah berikut menghapus sertifikat penandatanganan yang ditentukan untuk pengguna IAM bernama Bob

```
aws iam delete-signing-certificate \  
  --user-name Bob \  
  --certificate-id TA7SMP42TDN5Z260BPJE7EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk mendapatkan ID untuk sertifikat penandatanganan, gunakan `list-signing-certificates` perintah.

Untuk informasi selengkapnya, lihat [Mengelola sertifikat penandatanganan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [DeleteSigningCertificate](#) di Referensi AWS CLI Perintah.

## delete-ssh-public-key

Contoh kode berikut menunjukkan cara menggunakan `delete-ssh-public-key`.

### AWS CLI

Untuk menghapus kunci publik SSH yang dilampirkan ke pengguna IAM

`delete-ssh-public-key` Perintah berikut menghapus kunci publik SSH tertentu yang dilampirkan ke pengguna IAM. `sofia`

```
aws iam delete-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA123456789EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan kunci SSH dan SSH dengan CodeCommit](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteSshPublicKey](#) di Referensi AWS CLI Perintah.

## delete-user-permissions-boundary

Contoh kode berikut menunjukkan cara menggunakan `delete-user-permissions-boundary`.

### AWS CLI

Untuk menghapus batas izin dari pengguna IAM

`delete-user-permissions-boundary` Contoh berikut menghapus batas izin yang dilampirkan ke pengguna IAM bernama `intern` Untuk menerapkan batas izin ke pengguna, gunakan perintah `put-user-permissions-boundary`

```
aws iam delete-user-permissions-boundary \  
  --user-name intern
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [DeleteUserPermissionsBoundary](#) di Referensi AWS CLI Perintah.

## delete-user-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-user-policy`.

### AWS CLI

Untuk menghapus kebijakan dari pengguna IAM

`delete-user-policy` Perintah berikut menghapus kebijakan yang ditentukan dari pengguna IAM bernama Bob.

```
aws iam delete-user-policy \  
  --user-name Bob \  
  --policy-name ExamplePolicy
```

Perintah ini tidak menghasilkan output.

Untuk mendapatkan daftar kebijakan untuk pengguna IAM, gunakan `list-user-policies` perintah.

Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di AWS akun Anda](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteUserPolicy](#) di Referensi AWS CLI Perintah.

## delete-user

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

### AWS CLI

Untuk menghapus pengguna IAM

`delete-user` Perintah berikut menghapus nama pengguna IAM Bob dari akun saat ini.

```
aws iam delete-user \  
  --user-name Bob
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus pengguna IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## delete-virtual-mfa-device

Contoh kode berikut menunjukkan cara menggunakan `delete-virtual-mfa-device`.

### AWS CLI

Untuk menghapus perangkat MFA virtual

`delete-virtual-mfa-device` Perintah berikut menghapus perangkat MFA yang ditentukan dari akun saat ini.

```
aws iam delete-virtual-mfa-device \  
  --serial-number arn:aws:iam::123456789012:mfa/MFATest
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan perangkat MFA](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [DeleteVirtualMfaDevice](#) di Referensi AWS CLI Perintah.

## detach-group-policy

Contoh kode berikut menunjukkan cara menggunakan `detach-group-policy`.

### AWS CLI

Untuk melepaskan kebijakan dari grup

Contoh ini menghapus kebijakan terkelola dengan ARN `arn:aws:iam::123456789012:policy/TesterAccessPolicy` dari grup yang dipanggil `Testers`

```
aws iam detach-group-policy \  
  --group-name Testers \  
  --policy-arn arn:aws:iam::123456789012:policy/TesterAccessPolicy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DetachGroupPolicy](#) di Referensi AWS CLI Perintah.

## detach-role-policy

Contoh kode berikut menunjukkan cara menggunakan `detach-role-policy`.

### AWS CLI

Untuk melepaskan kebijakan dari peran

Contoh ini menghapus kebijakan terkelola dengan ARN

`arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy` dari peran yang dipanggil. `FedTesterRole`

```
aws iam detach-role-policy \  
  --role-name FedTesterRole \  
  --policy-arn arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memodifikasi peran](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DetachRolePolicy](#) di Referensi AWS CLI Perintah.

## detach-user-policy

Contoh kode berikut menunjukkan cara menggunakan `detach-user-policy`.

### AWS CLI

Untuk melepaskan kebijakan dari pengguna

Contoh ini menghapus kebijakan terkelola dengan ARN

`arn:aws:iam::123456789012:policy/TesterPolicy` dari pengguna. `Bob`

```
aws iam detach-user-policy \  
  --user-name Bob \  
  --policy-arn arn:aws:iam::123456789012:policy/TesterPolicy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengubah izin untuk pengguna IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [DetachUserPolicy](#) di Referensi AWS CLI Perintah.



## enable-mfa-device

Contoh kode berikut menunjukkan cara menggunakan `enable-mfa-device`.

### AWS CLI

Untuk mengaktifkan perangkat MFA

Setelah Anda menggunakan `create-virtual-mfa-device` perintah untuk membuat perangkat MFA virtual baru, Anda dapat menetapkan perangkat MFA ke pengguna. `enable-mfa-device` Contoh berikut menetapkan perangkat MFA dengan `arn:aws:iam::210987654321:mfa/BobsMFADevice` nomor seri kepada pengguna. Bob Perintah ini juga menyinkronkan perangkat AWS dengan memasukkan dua kode pertama secara berurutan dari perangkat MFA virtual.

```
aws iam enable-mfa-device \  
  --user-name Bob \  
  --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice \  
  --authentication-code1 123456 \  
  --authentication-code2 789012
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan perangkat autentikasi multi-faktor virtual \(MFA\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [EnableMfaDevice](#) di Referensi AWS CLI Perintah.

## generate-credential-report

Contoh kode berikut menunjukkan cara menggunakan `generate-credential-report`.

### AWS CLI

Untuk menghasilkan laporan kredensi

Contoh berikut mencoba untuk menghasilkan laporan kredensi untuk AWS akun.

```
aws iam generate-credential-report
```

Output:

```
{
  "State": "STARTED",
  "Description": "No report exists. Starting a new report generation task"
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan laporan kredensi untuk AWS akun Anda](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GenerateCredentialReport](#) di Referensi AWS CLI Perintah.

## **generate-organizations-access-report**

Contoh kode berikut menunjukkan cara menggunakan `generate-organizations-access-report`.

### AWS CLI

Contoh 1: Untuk menghasilkan laporan akses untuk root dalam organisasi

`generate-organizations-access-report` Contoh berikut memulai pekerjaan latar belakang untuk membuat laporan akses untuk root yang ditentukan dalam organisasi. Anda dapat menampilkan laporan setelah dibuat dengan menjalankan `get-organizations-access-report` perintah.

```
aws iam generate-organizations-access-report \
  --entity-path o-4fxmplt198/r-c3xb
```

Output:

```
{
  "JobId": "a8b6c06f-aaa4-8xmp-28bc-81da71836359"
}
```

Contoh 2: Untuk menghasilkan laporan akses untuk akun di organisasi

`generate-organizations-access-report` Contoh berikut memulai pekerjaan latar belakang untuk membuat laporan akses untuk ID akun 123456789012 di organisasi `o-4fxmplt198`. Anda dapat menampilkan laporan setelah dibuat dengan menjalankan `get-organizations-access-report` perintah.

```
aws iam generate-organizations-access-report \  
  --entity-path o-4fxmpl198/r-c3xb/123456789012
```

Output:

```
{  
  "JobId": "14b6c071-75f6-2xmp-fb77-faf6fb4201d2"  
}
```

Contoh 3: Untuk menghasilkan laporan akses untuk akun di unit organisasi dalam suatu organisasi

`generate-organizations-access-report` Contoh berikut memulai pekerjaan latar belakang untuk membuat laporan akses untuk ID akun 234567890123 di unit organisasi `ou-c3xb-lmu7j2yg` dalam organisasi `o-4fxmpl198`. Anda dapat menampilkan laporan setelah dibuat dengan menjalankan `get-organizations-access-report` perintah.

```
aws iam generate-organizations-access-report \  
  --entity-path o-4fxmpl198/r-c3xb/ou-c3xb-lmu7j2yg/234567890123
```

Output:

```
{  
  "JobId": "2eb6c2e6-0xmp-ec04-1425-c937916a64af"  
}
```

Untuk mendapatkan detail tentang akar dan unit organisasi di organisasi Anda, gunakan perintah `organizations list-roots` dan `organizations list-organizational-units-for-parent` perintah.

Untuk informasi selengkapnya, lihat [Menyempurnakan izin dalam AWS menggunakan informasi yang terakhir diakses](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GenerateOrganizationsAccessReport](#) di Referensi AWS CLI Perintah.

## **generate-service-last-accessed-details**

Contoh kode berikut menunjukkan cara menggunakan `generate-service-last-accessed-details`.

## AWS CLI

Contoh 1: Untuk membuat laporan akses layanan untuk kebijakan kustom

`generate-service-last-accessed-details` Contoh berikut memulai pekerjaan latar belakang untuk menghasilkan laporan yang mencantumkan layanan yang diakses oleh pengguna IAM dan entitas lain dengan kebijakan kustom bernama `intern-boundary`. Anda dapat menampilkan laporan setelah dibuat dengan menjalankan `get-service-last-accessed-details` perintah.

```
aws iam generate-service-last-accessed-details \  
  --arn arn:aws:iam::123456789012:policy/intern-boundary
```

Output:

```
{  
  "JobId": "2eb6c2b8-7b4c-3xmp-3c13-03b72c8cdfdc"  
}
```

Contoh 2: Untuk membuat laporan akses layanan untuk `AdministratorAccess` kebijakan AWS terkelola

`generate-service-last-accessed-details` Contoh berikut memulai pekerjaan latar belakang untuk menghasilkan laporan yang mencantumkan layanan yang diakses oleh pengguna IAM dan entitas lain dengan `AdministratorAccess` kebijakan AWS terkelola. Anda dapat menampilkan laporan setelah dibuat dengan menjalankan `get-service-last-accessed-details` perintah.

```
aws iam generate-service-last-accessed-details \  
  --arn arn:aws:iam::aws:policy/AdministratorAccess
```

Output:

```
{  
  "JobId": "78b6c2ba-d09e-6xmp-7039-ecde30b26916"  
}
```

Untuk informasi selengkapnya, lihat [Menyempurnakan izin dalam AWS menggunakan informasi yang terakhir diakses](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GenerateServiceLastAccessedDetails](#) di Referensi AWS CLI Perintah.

## get-access-key-last-used

Contoh kode berikut menunjukkan cara menggunakan `get-access-key-last-used`.

### AWS CLI

Untuk mengambil informasi tentang kapan kunci akses yang ditentukan terakhir digunakan

Contoh berikut mengambil informasi tentang kapan kunci akses ABCDEXAMPLE terakhir digunakan.

```
aws iam get-access-key-last-used \
  --access-key-id ABCDEXAMPLE
```

Output:

```
{
  "UserName": "Bob",
  "AccessKeyLastUsed": {
    "Region": "us-east-1",
    "ServiceName": "iam",
    "LastUsedDate": "2015-06-16T22:45:00Z"
  }
}
```

Untuk informasi lebih lanjut, lihat [Mengelola access key untuk pengguna IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetAccessKeyLastUsed](#) di Referensi AWS CLI Perintah.

## get-account-authorization-details

Contoh kode berikut menunjukkan cara menggunakan `get-account-authorization-details`.

### AWS CLI

Untuk mencantumkan AWS akun, pengguna, grup, peran, dan kebijakan IAM

`get-account-authorization-details` Perintah berikut menampilkan informasi tentang semua pengguna, grup, peran, dan kebijakan IAM di AWS akun.

```
aws iam get-account-authorization-details
```

## Output:

```
{
  "RoleDetailList": [
    {
      "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
              "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
          }
        ]
      },
      "RoleId": "AROA1234567890EXAMPLE",
      "CreateDate": "2014-07-30T17:09:20Z",
      "InstanceProfileList": [
        {
          "InstanceProfileId": "AIPA1234567890EXAMPLE",
          "Roles": [
            {
              "AssumeRolePolicyDocument": {
                "Version": "2012-10-17",
                "Statement": [
                  {
                    "Sid": "",
                    "Effect": "Allow",
                    "Principal": {
                      "Service": "ec2.amazonaws.com"
                    },
                    "Action": "sts:AssumeRole"
                  }
                ]
              },
              "RoleId": "AROA1234567890EXAMPLE",
              "CreateDate": "2014-07-30T17:09:20Z",
              "RoleName": "EC2role",
              "Path": "/",
              "Arn": "arn:aws:iam::123456789012:role/EC2role"
            }
          ]
        }
      ]
    }
  ]
}
```

```

        ],
        "CreateDate": "2014-07-30T17:09:20Z",
        "InstanceProfileName": "EC2role",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:instance-profile/EC2role"
    }
],
"RoleName": "EC2role",
"Path": "/",
"AttachedManagedPolicies": [
    {
        "PolicyName": "AmazonS3FullAccess",
        "PolicyArn": "arn:aws:iam::aws:policy/AmazonS3FullAccess"
    },
    {
        "PolicyName": "AmazonDynamoDBFullAccess",
        "PolicyArn": "arn:aws:iam::aws:policy/AmazonDynamoDBFullAccess"
    }
],
"RoleLastUsed": {
    "Region": "us-west-2",
    "LastUsedDate": "2019-11-13T17:30:00Z"
},
"RolePolicyList": [],
"Arn": "arn:aws:iam::123456789012:role/EC2role"
}
],
"GroupDetailList": [
    {
        "GroupId": "AIDA1234567890EXAMPLE",
        "AttachedManagedPolicies": {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        },
        "GroupName": "Admins",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:group/Admins",
        "CreateDate": "2013-10-14T18:32:24Z",
        "GroupPolicyList": []
    },
    {
        "GroupId": "AIDA1234567890EXAMPLE",
        "AttachedManagedPolicies": {
            "PolicyName": "PowerUserAccess",

```

```

        "PolicyArn": "arn:aws:iam::aws:policy/PowerUserAccess"
    },
    "GroupName": "Dev",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:group/Dev",
    "CreateDate": "2013-10-14T18:33:55Z",
    "GroupPolicyList": []
},
{
    "GroupId": "AIDA1234567890EXAMPLE",
    "AttachedManagedPolicies": [],
    "GroupName": "Finance",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:group/Finance",
    "CreateDate": "2013-10-14T18:57:48Z",
    "GroupPolicyList": [
        {
            "PolicyName": "policygen-201310141157",
            "PolicyDocument": {
                "Version": "2012-10-17",
                "Statement": [
                    {
                        "Action": "aws-portal:*",
                        "Sid": "Stmt1381777017000",
                        "Resource": "*",
                        "Effect": "Allow"
                    }
                ]
            }
        }
    ]
}
],
"UserDetailList": [
    {
        "UserName": "Alice",
        "GroupList": [
            "Admins"
        ],
        "CreateDate": "2013-10-14T18:32:24Z",
        "UserId": "AIDA1234567890EXAMPLE",
        "UserPolicyList": [],
        "Path": "/",
        "AttachedManagedPolicies": [],

```



```
    "Arn": "arn:aws:iam::123456789012:user/Alice"
  },
  {
    "UserName": "Bob",
    "GroupList": [
      "Admins"
    ],
    "CreateDate": "2013-10-14T18:32:25Z",
    "UserId": "AIDA1234567890EXAMPLE",
    "UserPolicyList": [
      {
        "PolicyName": "DenyBillingAndIAMPolicy",
        "PolicyDocument": {
          "Version": "2012-10-17",
          "Statement": {
            "Effect": "Deny",
            "Action": [
              "aws-portal:*",
              "iam:*"
            ],
            "Resource": "*"
          }
        }
      }
    ],
    "Path": "/",
    "AttachedManagedPolicies": [],
    "Arn": "arn:aws:iam::123456789012:user/Bob"
  },
  {
    "UserName": "Charlie",
    "GroupList": [
      "Dev"
    ],
    "CreateDate": "2013-10-14T18:33:56Z",
    "UserId": "AIDA1234567890EXAMPLE",
    "UserPolicyList": [],
    "Path": "/",
    "AttachedManagedPolicies": [],
    "Arn": "arn:aws:iam::123456789012:user/Charlie"
  }
],
"Policies": [
  {
```

```
"PolicyName": "create-update-delete-set-managed-policies",
"CreateDate": "2015-02-06T19:58:34Z",
"AttachmentCount": 1,
"IsAttachable": true,
"PolicyId": "ANPA1234567890EXAMPLE",
"DefaultVersionId": "v1",
"PolicyVersionList": [
  {
    "CreateDate": "2015-02-06T19:58:34Z",
    "VersionId": "v1",
    "Document": {
      "Version": "2012-10-17",
      "Statement": {
        "Effect": "Allow",
        "Action": [
          "iam:CreatePolicy",
          "iam:CreatePolicyVersion",
          "iam>DeletePolicy",
          "iam>DeletePolicyVersion",
          "iam:GetPolicy",
          "iam:GetPolicyVersion",
          "iam>ListPolicies",
          "iam>ListPolicyVersions",
          "iam:SetDefaultPolicyVersion"
        ],
        "Resource": "*"
      }
    },
    "IsDefaultVersion": true
  }
],
"Path": "/",
"Arn": "arn:aws:iam::123456789012:policy/create-update-delete-set-
managed-policies",
"UpdateDate": "2015-02-06T19:58:34Z"
},
{
  "PolicyName": "S3-read-only-specific-bucket",
  "CreateDate": "2015-01-21T21:39:41Z",
  "AttachmentCount": 1,
  "IsAttachable": true,
  "PolicyId": "ANPA1234567890EXAMPLE",
  "DefaultVersionId": "v1",
  "PolicyVersionList": [
```

```

    {
      "CreateDate": "2015-01-21T21:39:41Z",
      "VersionId": "v1",
      "Document": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Effect": "Allow",
            "Action": [
              "s3:Get*",
              "s3:List*"
            ],
            "Resource": [
              "arn:aws:s3:::example-bucket",
              "arn:aws:s3:::example-bucket/*"
            ]
          }
        ]
      },
      "IsDefaultVersion": true
    }
  ],
  "Path": "/",
  "Arn": "arn:aws:iam::123456789012:policy/S3-read-only-specific-bucket",
  "UpdateDate": "2015-01-21T23:39:41Z"
},
{
  "PolicyName": "AmazonEC2FullAccess",
  "CreateDate": "2015-02-06T18:40:15Z",
  "AttachmentCount": 1,
  "IsAttachable": true,
  "PolicyId": "ANPA1234567890EXAMPLE",
  "DefaultVersionId": "v1",
  "PolicyVersionList": [
    {
      "CreateDate": "2014-10-30T20:59:46Z",
      "VersionId": "v1",
      "Document": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Action": "ec2:*",
            "Effect": "Allow",
            "Resource": "*"
          }
        ]
      }
    }
  ]
}

```

```

        },
        {
            "Effect": "Allow",
            "Action": "elasticloadbalancing:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "cloudwatch:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "autoscaling:*",
            "Resource": "*"
        }
    ]
},
    "IsDefaultVersion": true
}
],
    "Path": "/",
    "Arn": "arn:aws:iam::aws:policy/AmazonEC2FullAccess",
    "UpdateDate": "2015-02-06T18:40:15Z"
}
],
    "Marker": "EXAMPLEkakov9BCuUNFDtxWSyetzYwEx2ADc8dnzfvERF5S6YMvXKx41t6gCl/
    eeaCX3Jo94/bKqezEAg8TEVS99EKFLxm3jtbpl25FDWEXAMPLE",
    "IsTruncated": true
}

```

Untuk informasi selengkapnya, lihat [pedoman audit AWS keamanan](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetAccountAuthorizationDetails](#) di Referensi AWS CLI Perintah.

## get-account-password-policy

Contoh kode berikut menunjukkan cara menggunakan `get-account-password-policy`.

### AWS CLI

Untuk melihat kebijakan kata sandi akun saat ini

`get-account-password-policy` Perintah berikut menampilkan detail tentang kebijakan kata sandi untuk akun saat ini.

```
aws iam get-account-password-policy
```

Output:

```
{
  "PasswordPolicy": {
    "AllowUsersToChangePassword": false,
    "RequireLowercaseCharacters": false,
    "RequireUppercaseCharacters": false,
    "MinimumPasswordLength": 8,
    "RequireNumbers": true,
    "RequireSymbols": true
  }
}
```

Jika tidak ada kebijakan kata sandi yang ditentukan untuk akun, perintah mengembalikan `NoSuchEntity` kesalahan.

Untuk informasi selengkapnya, lihat [Menyetel kebijakan kata sandi akun untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetAccountPasswordPolicy](#) di Referensi AWS CLI Perintah.

## **get-account-summary**

Contoh kode berikut menunjukkan cara menggunakan `get-account-summary`.

AWS CLI

Untuk mendapatkan informasi tentang penggunaan entitas IAM dan kuota IAM di akun saat ini

`get-account-summary` Perintah berikut mengembalikan informasi tentang penggunaan entitas IAM saat ini dan kuota entitas IAM saat ini di akun.

```
aws iam get-account-summary
```

Output:

```
{
```

```
"SummaryMap": {
  "UsersQuota": 5000,
  "GroupsQuota": 100,
  "InstanceProfiles": 6,
  "SigningCertificatesPerUserQuota": 2,
  "AccountAccessKeysPresent": 0,
  "RolesQuota": 250,
  "RolePolicySizeQuota": 10240,
  "AccountSigningCertificatesPresent": 0,
  "Users": 27,
  "ServerCertificatesQuota": 20,
  "ServerCertificates": 0,
  "AssumeRolePolicySizeQuota": 2048,
  "Groups": 7,
  "MFADevicesInUse": 1,
  "Roles": 3,
  "AccountMFAEnabled": 1,
  "MFADevices": 3,
  "GroupsPerUserQuota": 10,
  "GroupPolicySizeQuota": 5120,
  "InstanceProfilesQuota": 100,
  "AccessKeysPerUserQuota": 2,
  "Providers": 0,
  "UserPolicySizeQuota": 2048
}
```

Untuk informasi selengkapnya tentang batasan entitas, lihat [kuota IAM dan AWS STS](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetAccountSummary](#) di Referensi AWS CLI Perintah.

## get-context-keys-for-custom-policy

Contoh kode berikut menunjukkan cara menggunakan `get-context-keys-for-custom-policy`.

### AWS CLI

Contoh 1: Untuk mencantumkan kunci konteks yang direferensikan oleh satu atau beberapa kebijakan JSON kustom yang disediakan sebagai parameter pada baris perintah

`get-context-keys-for-custom-policy` Perintah berikut mem-parsing setiap kebijakan yang disediakan dan mencantumkan kunci konteks yang digunakan oleh kebijakan tersebut.

Gunakan perintah ini untuk mengidentifikasi nilai kunci konteks mana yang harus Anda berikan agar berhasil menggunakan perintah simulator kebijakan `simulate-custom-policy` dan `simulate-custom-policy`. Anda juga dapat mengambil daftar kunci konteks yang digunakan oleh semua kebijakan yang terkait oleh pengguna IAM atau peran dengan menggunakan perintah `get-context-keys-for-custom-policy`. Nilai parameter yang dimulai dengan `file://` menginstruksikan perintah untuk membaca file dan menggunakan konten sebagai nilai untuk parameter, bukan nama file itu sendiri.

```
aws iam get-context-keys-for-custom-policy \
  --policy-input-list '{"Version":"2012-10-17","Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"arn:aws:dynamodb:us-
west-2:123456789012:table/${aws:username}","Condition":{"DateGreaterThan":
{"aws:CurrentTime":"2015-08-16T12:00:00Z"}}}]'
```

Output:

```
{
  "ContextKeyNames": [
    "aws:username",
    "aws:CurrentTime"
  ]
}
```

Contoh 2: Untuk mencantumkan kunci konteks yang direferensikan oleh satu atau beberapa kebijakan JSON kustom yang disediakan sebagai input file

`get-context-keys-for-custom-policy` Perintah berikut ini sama dengan contoh sebelumnya, kecuali bahwa kebijakan disediakan dalam file, bukan sebagai parameter. Karena perintah mengharapkan daftar string JSON, dan bukan daftar struktur JSON, file harus terstruktur sebagai berikut, meskipun Anda dapat menciutkannya menjadi satu.

```
[
  "Policy1",
  "Policy2"
]
```

Jadi misalnya, file yang berisi kebijakan dari contoh sebelumnya harus terlihat seperti berikut. Anda harus menghindari setiap kutipan ganda yang disematkan di dalam string kebijakan dengan mendahuluinya dengan garis miring terbalik”.

```
[ {"Statement": {"Effect": "Allow", "Action": "dynamodb:*", "Resource": "arn:aws:dynamodb:us-west-2:128716708097:table/${aws:username}", "Condition": {"DateGreaterThan": {"aws:CurrentTime": "2015-08-16T12:00:00Z"}}}} ]
```

File ini kemudian dapat dikirimkan ke perintah berikut.

```
aws iam get-context-keys-for-custom-policy \
  --policy-input-list file://policyfile.json
```

Output:

```
{
  "ContextKeyNames": [
    "aws:username",
    "aws:CurrentTime"
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan IAM Policy Simulator \(AWS CLI AWS dan API\)](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetContextKeysForCustomPolicy](#) di Referensi AWS CLI Perintah.

## get-context-keys-for-principal-policy

Contoh kode berikut menunjukkan cara menggunakan `get-context-keys-for-principal-policy`.

### AWS CLI

Untuk membuat daftar kunci konteks yang direferensikan oleh semua kebijakan yang terkait dengan prinsipal IAM

`get-context-keys-for-principal-policy` Perintah berikut mengambil semua kebijakan yang dilampirkan ke pengguna `saarvi` dan grup mana pun yang menjadi anggotanya. Kemudian mem-parsing masing-masing dan mencantumkan kunci konteks yang digunakan oleh kebijakan tersebut. Gunakan perintah ini untuk mengidentifikasi nilai kunci konteks mana yang harus Anda berikan agar berhasil menggunakan `simulate-principal-policy` perintah `simulate-custom-policy` dan. Anda juga dapat mengambil daftar kunci konteks yang digunakan oleh



kebijakan JSON arbitrer dengan menggunakan perintah. `get-context-keys-for-custom-policy`

```
aws iam get-context-keys-for-principal-policy \  
  --policy-source-arn arn:aws:iam::123456789012:user/saanvi
```

Output:

```
{  
  "ContextKeyNames": [  
    "aws:username",  
    "aws:CurrentTime"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan IAM Policy Simulator \(AWS CLI AWS dan API\)](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetContextKeysForPrincipalPolicy](#) di Referensi AWS CLI Perintah.

## **get-credential-report**

Contoh kode berikut menunjukkan cara menggunakan `get-credential-report`.

AWS CLI

Untuk mendapatkan laporan kredensi

Contoh ini membuka laporan yang dikembalikan dan mengeluarkannya ke pipeline sebagai array baris teks.

```
aws iam get-credential-report
```

Output:

```
{  
  "GeneratedTime": "2015-06-17T19:11:50Z",  
  "ReportFormat": "text/csv"  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan laporan kredensi untuk AWS akun Anda](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetCredentialReport](#) di Referensi AWS CLI Perintah.

## get-group-policy

Contoh kode berikut menunjukkan cara menggunakan `get-group-policy`.

### AWS CLI

Untuk mendapatkan informasi tentang kebijakan yang dilampirkan ke grup IAM

`get-group-policy` Perintah berikut mendapatkan informasi tentang kebijakan tertentu yang dilampirkan ke grup bernama `Test-Group`.

```
aws iam get-group-policy \
  --group-name Test-Group \
  --policy-name S3-ReadOnly-Policy
```

Output:

```
{
  "GroupName": "Test-Group",
  "PolicyDocument": {
    "Statement": [
      {
        "Action": [
          "s3:Get*",
          "s3:List*"
        ],
        "Resource": "*",
        "Effect": "Allow"
      }
    ]
  },
  "PolicyName": "S3-ReadOnly-Policy"
}
```

Untuk informasi selengkapnya, lihat [Mengelola kebijakan IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetGroupPolicy](#) di Referensi AWS CLI Perintah.

## get-group

Contoh kode berikut menunjukkan cara menggunakan `get-group`.

### AWS CLI

Untuk mendapatkan grup IAM

Contoh ini mengembalikan rincian tentang grup Admins IAM.

```
aws iam get-group \
  --group-name Admins
```

Output:

```
{
  "Group": {
    "Path": "/",
    "CreateDate": "2015-06-16T19:41:48Z",
    "GroupId": "AIDGPMS9R04H3FEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/Admins",
    "GroupName": "Admins"
  },
  "Users": []
}
```

Untuk informasi selengkapnya, lihat [Identitas IAM \(pengguna, grup pengguna, dan peran\)](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetGroup](#) di Referensi AWS CLI Perintah.

## get-instance-profile

Contoh kode berikut menunjukkan cara menggunakan `get-instance-profile`.

### AWS CLI

Untuk mendapatkan informasi tentang profil instance

`get-instance-profile` Perintah berikut mendapatkan informasi tentang profil instance bernama `ExampleInstanceProfile`.

```
aws iam get-instance-profile \
```

```
--instance-profile-name ExampleInstanceProfile
```

Output:

```
{
  "InstanceProfile": {
    "InstanceProfileId": "AID2MAB8DPLSRHEXAMPLE",
    "Roles": [
      {
        "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
        "RoleId": "AIDGPM59R04H3FEXAMPLE",
        "CreateDate": "2013-01-09T06:33:26Z",
        "RoleName": "Test-Role",
        "Path": "/",
        "Arn": "arn:aws:iam::336924118301:role/Test-Role"
      }
    ],
    "CreateDate": "2013-06-12T23:52:02Z",
    "InstanceProfileName": "ExampleInstanceProfile",
    "Path": "/",
    "Arn": "arn:aws:iam::336924118301:instance-profile/ExampleInstanceProfile"
  }
}
```

Untuk informasi lebih lanjut, lihat [Menggunakan profil instance](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetInstanceProfile](#) di Referensi AWS CLI Perintah.

## get-login-profile

Contoh kode berikut menunjukkan cara menggunakan `get-login-profile`.

### AWS CLI

Untuk mendapatkan informasi kata sandi untuk pengguna IAM

`get-login-profile` Perintah berikut mendapatkan informasi tentang password untuk pengguna IAM bernama Bob.

```
aws iam get-login-profile \
  --user-name Bob
```

**Output:**

```
{
  "LoginProfile": {
    "UserName": "Bob",
    "CreateDate": "2012-09-21T23:03:39Z"
  }
}
```

`get-login-profile` Perintah ini dapat digunakan untuk memverifikasi bahwa pengguna IAM memiliki kata sandi. Perintah mengembalikan `NoSuchEntity` kesalahan jika tidak ada kata sandi yang ditentukan untuk pengguna.

Anda tidak dapat melihat kata sandi menggunakan perintah ini. Jika kata sandi hilang, Anda dapat mengatur ulang kata sandi (`update-login-profile`) untuk pengguna. Atau, Anda dapat menghapus profil login (`delete-login-profile`) untuk pengguna dan kemudian membuat yang baru (`create-login-profile`).

Untuk informasi selengkapnya, lihat [Mengelola kata sandi untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetLoginProfile](#) di Referensi AWS CLI Perintah.

**get-mfa-device**

Contoh kode berikut menunjukkan cara menggunakan `get-mfa-device`.

**AWS CLI**

Untuk mengambil informasi tentang kunci keamanan FIDO

Contoh `get-mfa-device` perintah berikut mengambil informasi tentang kunci keamanan FIDO yang ditentukan.

```
aws iam get-mfa-device \
  --serial-number arn:aws:iam::123456789012:u2f/user/alice/fidokeyname-
  EXAMPLEBN5FHTECLFG7EXAMPLE
```

**Output:**

```
{
```

```

    "UserName": "alice",
    "SerialNumber": "arn:aws:iam::123456789012:u2f/user/alice/fidokeyname-
EXAMPLEBN5FHTECLFG7EXAMPLE",
    "EnableDate": "2023-09-19T01:49:18+00:00",
    "Certifications": {
        "FIDO": "L1"
    }
}

```

Untuk informasi selengkapnya, lihat [Menggunakan autentikasi multifaktor \(MFA\) dalam AWS](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetMfaDevice](#) di Referensi AWS CLI Perintah.

## get-open-id-connect-provider

Contoh kode berikut menunjukkan cara menggunakan `get-open-id-connect-provider`.

### AWS CLI

Untuk mengembalikan informasi tentang penyedia OpenID Connect yang ditentukan

Contoh ini mengembalikan rincian tentang penyedia OpenID Connect yang ARN adalah `arn:aws:iam::123456789012:oidc-provider/server.example.com`

```

aws iam get-open-id-connect-provider \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com

```

### Output:

```

{
  "Url": "server.example.com"
  "CreateDate": "2015-06-16T19:41:48Z",
  "ThumbprintList": [
    "12345678901234567890lmnopqrst987example"
  ],
  "ClientIDList": [
    "example-application-ID"
  ]
}

```

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [GetOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## get-organizations-access-report

Contoh kode berikut menunjukkan cara menggunakan `get-organizations-access-report`.

### AWS CLI

Untuk mengambil laporan akses

`get-organizations-access-report` Contoh berikut menampilkan laporan akses yang dibuat sebelumnya untuk entitas AWS Organizations. Untuk menghasilkan laporan, gunakan `generate-organizations-access-report` perintah.

```
aws iam get-organizations-access-report \
  --job-id a8b6c06f-aaa4-8xmp-28bc-81da71836359
```

### Output:

```
{
  "JobStatus": "COMPLETED",
  "JobCreationDate": "2019-09-30T06:53:36.187Z",
  "JobCompletionDate": "2019-09-30T06:53:37.547Z",
  "NumberOfServicesAccessible": 188,
  "NumberOfServicesNotAccessed": 171,
  "AccessDetails": [
    {
      "ServiceName": "Alexa for Business",
      "ServiceNamespace": "a4b",
      "TotalAuthenticatedEntities": 0
    },
    ...
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyempurnakan izin dalam AWS menggunakan informasi yang terakhir diakses](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetOrganizationsAccessReport](#) di Referensi AWS CLI Perintah.

## get-policy-version

Contoh kode berikut menunjukkan cara menggunakan `get-policy-version`.

### AWS CLI

Untuk mengambil informasi tentang versi tertentu dari kebijakan terkelola yang ditentukan

Contoh ini mengembalikan dokumen kebijakan untuk versi v2 dari kebijakan yang ARN adalah `arn:aws:iam::123456789012:policy/MyManagedPolicy`

```
aws iam get-policy-version \
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \
  --version-id v2
```

Output:

```
{
  "PolicyVersion": {
    "Document": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Action": "iam:*",
          "Resource": "*"
        }
      ]
    },
    "VersionId": "v2",
    "IsDefaultVersion": true,
    "CreateDate": "2023-04-11T00:22:54+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetPolicyVersion](#) di Referensi AWS CLI Perintah.

## get-policy

Contoh kode berikut menunjukkan cara menggunakan `get-policy`.



## AWS CLI

Untuk mengambil informasi tentang kebijakan terkelola yang ditentukan

Contoh ini mengembalikan rincian tentang kebijakan terkelola yang ARN.

`arn:aws:iam::123456789012:policy/MySamplePolicy`

```
aws iam get-policy \  
  --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Output:

```
{  
  "Policy": {  
    "PolicyName": "MySamplePolicy",  
    "CreateDate": "2015-06-17T19:23:32Z",  
    "AttachmentCount": 0,  
    "IsAttachable": true,  
    "PolicyId": "Z27SI6FQMG2EXAMPLE1",  
    "DefaultVersionId": "v1",  
    "Path": "/",  
    "Arn": "arn:aws:iam::123456789012:policy/MySamplePolicy",  
    "UpdateDate": "2015-06-17T19:23:32Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetPolicy](#) di Referensi AWS CLI Perintah.

## get-role-policy

Contoh kode berikut menunjukkan cara menggunakan `get-role-policy`.

## AWS CLI

Untuk mendapatkan informasi tentang kebijakan yang dilampirkan pada peran IAM

`get-role-policy` Perintah berikut mendapatkan informasi tentang kebijakan tertentu yang dilampirkan pada peran bernama `Test-Role`.

```
aws iam get-role-policy \  
  --role-name Test-Role \  
  --policy-name ExamplePolicy
```

### Output:

```
{  
  "RoleName": "Test-Role",  
  "PolicyDocument": {  
    "Statement": [  
      {  
        "Action": [  
          "s3:ListBucket",  
          "s3:Put*",  
          "s3:Get*",  
          "s3:*MultipartUpload*"  
        ],  
        "Resource": "*",  
        "Effect": "Allow",  
        "Sid": "1"  
      }  
    ]  
  }  
  "PolicyName": "ExamplePolicy"  
}
```

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetRolePolicy](#) di Referensi AWS CLI Perintah.

## get-role

Contoh kode berikut menunjukkan cara menggunakan `get-role`.

### AWS CLI

Untuk mendapatkan informasi tentang peran IAM

`get-role` Perintah berikut mendapatkan informasi tentang peran bernama `Test-Role`.

```
aws iam get-role \  
  --role-name Test-Role
```

**Output:**

```
{
  "Role": {
    "Description": "Test Role",
    "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
    "MaxSessionDuration": 3600,
    "RoleId": "AROA1234567890EXAMPLE",
    "CreateDate": "2019-11-13T16:45:56Z",
    "RoleName": "Test-Role",
    "Path": "/",
    "RoleLastUsed": {
      "Region": "us-east-1",
      "LastUsedDate": "2019-11-13T17:14:00Z"
    },
    "Arn": "arn:aws:iam::123456789012:role/Test-Role"
  }
}
```

Perintah menampilkan kebijakan kepercayaan yang dilampirkan pada peran tersebut. Untuk mencantumkan kebijakan izin yang dilampirkan ke peran, gunakan `list-role-policies` perintah.

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetRole](#) di Referensi AWS CLI Perintah.

**get-saml-provider**

Contoh kode berikut menunjukkan cara menggunakan `get-saml-provider`.

**AWS CLI**

Untuk mengambil metadocument penyedia SAFL

Contoh ini mengambil rincian tentang penyedia SAFL 2.0 yang ARM.

`arn:aws:iam::123456789012:saml-provider/SAMLADFS` Responsnya mencakup dokumen metadata yang Anda dapatkan dari penyedia identitas untuk membuat entitas penyedia AWS SAML serta tanggal pembuatan dan kedaluwarsa.

```
aws iam get-saml-provider \
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFS
```

**Output:**

```
{
  "SAMLMetadataDocument": "...SAMLMetadataDocument-XML...",
  "CreateDate": "2017-03-06T22:29:46+00:00",
  "ValidUntil": "2117-03-06T22:29:46.433000+00:00",
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",
      "Value": "Accounting"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas IAM SALL](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetSamlProvider](#) di Referensi AWS CLI Perintah.

**get-server-certificate**

Contoh kode berikut menunjukkan cara menggunakan `get-server-certificate`.

**AWS CLI**

Untuk mendapatkan detail tentang sertifikat server di AWS akun Anda

`get-server-certificate` Perintah berikut mengambil semua detail tentang sertifikat server yang ditentukan di AWS akun Anda.

```
aws iam get-server-certificate \
  --server-certificate-name myUpdatedServerCertificate
```

**Output:**

```
{
  "ServerCertificate": {
    "ServerCertificateMetadata": {
```

```

    "Path": "/",
    "ServerCertificateName": "myUpdatedServerCertificate",
    "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:server-certificate/
myUpdatedServerCertificate",
    "UploadDate": "2019-04-22T21:13:44+00:00",
    "Expiration": "2019-10-15T22:23:16+00:00"
  },
  "CertificateBody": "-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMaKGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAwTC01BTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWVxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGfYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMaKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAwTC01BTSBDb25z
b2x1MRIwEAYDVQQDEwLUZXN0Q21sYWVxHzAdBgkqhkiG9w0BCQEWEG5vb251QGfY
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhdLQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJIIJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStb
NYiytVbZPQUQ5Yaxu2jXnimvrszlaEXAMPLE=-----END CERTIFICATE-----",
  "CertificateChain": "-----BEGIN CERTIFICATE-----\nMIICiTCCAfICCQD6md
7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMaKGA1UEBhMCMVVMxCzAJBgNVBAGT
AldBMRAwDgYDVQQHEwdTZWF0drGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAw
TC01BTSBDb25zb2x1MRIwEAYDVQsQQDEwLUZXN0Q21sYWVxHzAdBgkqhkiG9w0BCQ
jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTIwNDI0MjA0NTIxWjCBiDELMaKGA1UEBh
MCMVVMxCzAJBgNVBAGTAldBMRAwDgsYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBb
WF6b24xFDASBgNVBAwTC01BTSBDb2d5zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWVx
HzAdBgkqhkiG9w0BCQEWEG5vb251QGfYXpvbi5jb20wZ8wDQYJKoZIhvcNAQE
BBQADgY0AMIGJAoGBAMaK0dn+a4GmWIGWJ21uUSfwfEvySWtC2XADZ4nB+BLyGVI
k60CpiwsZ3G93vUEI03IyNoH/f0wYK8mh9TrDHudUZg3qX4waLG5M43q7Wgc/MbQ
ITx0USQv7c7ugFFDzQGBzZswY6786m86gjpEIbb30hjZnzcvcQAaRHhdLQWIMm2nr
AgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCku4nUHVxYUntneD9+h8Mg9q6q+auN
KyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0F1kbFFbjvSfpJIIJ00zbhNYS5f6Guo
EDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjS;TbNYiytVbZPQUQ5Yaxu2jXnimvw
3rrszlaEWEG5vb251QGftsYXpvbiEXAMPLE=\n-----END CERTIFICATE-----"
}
}

```

Untuk membuat daftar sertifikat server yang tersedia di AWS akun Anda, gunakan `list-server-certificates` perintah.

Untuk informasi selengkapnya, lihat [Mengelola sertifikat server di IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetServerCertificate](#) di Referensi AWS CLI Perintah.

## **get-service-last-accessed-details-with-entities**

Contoh kode berikut menunjukkan cara menggunakan `get-service-last-accessed-details-with-entities`.

### AWS CLI

Untuk mengambil laporan akses layanan dengan rincian untuk layanan

`get-service-last-accessed-details-with-entities` Contoh berikut mengambil laporan yang berisi rincian tentang pengguna IAM dan entitas lain yang mengakses layanan yang ditentukan. Untuk menghasilkan laporan, gunakan `generate-service-last-accessed-details` perintah. Untuk mendapatkan daftar layanan yang diakses dengan ruang nama, gunakan `get-service-last-accessed-details`

```
aws iam get-service-last-accessed-details-with-entities \
  --job-id 78b6c2ba-d09e-6xmp-7039-ecde30b26916 \
  --service-namespace lambda
```

### Output:

```
{
  "JobStatus": "COMPLETED",
  "JobCreationDate": "2019-10-01T03:55:41.756Z",
  "JobCompletionDate": "2019-10-01T03:55:42.533Z",
  "EntityDetailsList": [
    {
      "EntityInfo": {
        "Arn": "arn:aws:iam::123456789012:user/admin",
        "Name": "admin",
        "Type": "USER",
        "Id": "AIDAI02XMPLNQEXAMPLE",
        "Path": "/"
      },
      "LastAuthenticated": "2019-09-30T23:02:00Z"
    },
    {
```

```

    "EntityInfo": {
      "Arn": "arn:aws:iam::123456789012:user/developer",
      "Name": "developer",
      "Type": "USER",
      "Id": "AIDAIBEYXMPL2YEXAMPLE",
      "Path": "/"
    },
    "LastAuthenticated": "2019-09-16T19:34:00Z"
  ]
}

```

Untuk informasi selengkapnya, lihat [Menyempurnakan izin dalam AWS menggunakan informasi yang terakhir diakses](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetServiceLastAccessedDetailsWithEntities](#) di Referensi AWS CLI Perintah.

## get-service-last-accessed-details

Contoh kode berikut menunjukkan cara menggunakan `get-service-last-accessed-details`.

### AWS CLI

Untuk mengambil laporan akses layanan

`get-service-last-accessed-details` Contoh berikut mengambil laporan yang dihasilkan sebelumnya yang mencantumkan layanan yang diakses oleh entitas IAM. Untuk menghasilkan laporan, gunakan `generate-service-last-accessed-details` perintah.

```

aws iam get-service-last-accessed-details \
  --job-id 2eb6c2b8-7b4c-3xmp-3c13-03b72c8cdfdc

```

Output:

```

{
  "JobStatus": "COMPLETED",
  "JobCreationDate": "2019-10-01T03:50:35.929Z",
  "ServicesLastAccessed": [
    ...
    {
      "ServiceName": "AWS Lambda",

```

```
        "LastAuthenticated": "2019-09-30T23:02:00Z",
        "ServiceNamespace": "lambda",
        "LastAuthenticatedEntity": "arn:aws:iam::123456789012:user/admin",
        "TotalAuthenticatedEntities": 6
    },
]
}
```

Untuk informasi selengkapnya, lihat [Menyempurnakan izin dalam AWS menggunakan informasi yang terakhir diakses](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetServiceLastAccessedDetails](#) di Referensi AWS CLI Perintah.

## get-service-linked-role-deletion-status

Contoh kode berikut menunjukkan cara menggunakan `get-service-linked-role-deletion-status`.

### AWS CLI

Untuk memeriksa status permintaan untuk menghapus peran terkait layanan

`get-service-linked-role-deletion-status` Contoh berikut menampilkan status permintaan sebelumnya untuk menghapus peran terkait layanan. Operasi hapus terjadi secara asinkron. Ketika Anda membuat permintaan, Anda mendapatkan `DeletionTaskId` nilai yang Anda berikan sebagai parameter untuk perintah ini.

```
aws iam get-service-linked-role-deletion-status \
    --deletion-task-id task/aws-service-role/lex.amazonaws.com/
AWSServiceRoleForLexBots/1a2b3c4d-1234-abcd-7890-abcdeEXAMPLE
```

### Output:

```
{
  "Status": "SUCCEEDED"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan peran terkait layanan](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetServiceLinkedRoleDeletionStatus](#) di Referensi AWS CLI Perintah.



## get-ssh-public-key

Contoh kode berikut menunjukkan cara menggunakan `get-ssh-public-key`.

### AWS CLI

Contoh 1: Untuk mengambil kunci publik SSH yang dilampirkan ke pengguna IAM dalam bentuk yang dikodekan SSH

`get-ssh-public-key` Perintah berikut mengambil kunci publik SSH yang ditentukan dari pengguna IAM. `sofia` Outputnya dalam pengkodean SSH.

```
aws iam get-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA123456789EXAMPLE \  
  --encoding SSH
```

Output:

```
{  
  "SSHPublicKey": {  
    "UserName": "sofia",  
    "SSHPublicKeyId": "APKA123456789EXAMPLE",  
    "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",  
    "SSHPublicKeyBody": "ssh-rsa <<long encoded SSH string>>",  
    "Status": "Inactive",  
    "UploadDate": "2019-04-18T17:04:49+00:00"  
  }  
}
```

Contoh 2: Untuk mengambil kunci publik SSH yang dilampirkan ke pengguna IAM dalam bentuk yang dikodekan PEM

`get-ssh-public-key` Perintah berikut mengambil kunci publik SSH yang ditentukan dari pengguna IAM. `sofia` Outputnya dalam pengkodean PEM.

```
aws iam get-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA123456789EXAMPLE \  
  --encoding PEM
```

Output:

```
{
  "SSHPublicKey": {
    "UserName": "sofia",
    "SSHPublicKeyId": "APKA123456789EXAMPLE",
    "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
    "SSHPublicKeyBody": ""-----BEGIN PUBLIC KEY-----\n<<long encoded PEM
string>>\n-----END PUBLIC KEY-----\n"",
    "Status": "Inactive",
    "UploadDate": "2019-04-18T17:04:49+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Menggunakan kunci SSH dan SSH dengan CodeCommit](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetSshPublicKey](#) di Referensi AWS CLI Perintah.

## get-user-policy

Contoh kode berikut menunjukkan cara menggunakan `get-user-policy`.

### AWS CLI

Untuk mencantumkan detail kebijakan untuk pengguna IAM

`get-user-policy` Perintah berikut mencantumkan rincian kebijakan tertentu yang dilampirkan ke pengguna IAM bernama Bob.

```
aws iam get-user-policy \
  --user-name Bob \
  --policy-name ExamplePolicy
```

### Output:

```
{
  "UserName": "Bob",
  "PolicyName": "ExamplePolicy",
  "PolicyDocument": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "*",
```

```
        "Resource": "*",
        "Effect": "Allow"
      }
    ]
  }
}
```

Untuk mendapatkan daftar kebijakan untuk pengguna IAM, gunakan `list-user-policies` perintah.

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetUserPolicy](#) di Referensi AWS CLI Perintah.

## get-user

Contoh kode berikut menunjukkan cara menggunakan `get-user`.

### AWS CLI

Untuk mendapatkan informasi tentang pengguna IAM

`get-user` Perintah berikut mendapatkan informasi tentang nama Paulo pengguna IAM.

```
aws iam get-user \
  --user-name Paulo
```

Output:

```
{
  "User": {
    "UserName": "Paulo",
    "Path": "/",
    "CreateDate": "2019-09-21T23:03:13Z",
    "UserId": "AIDA123456789EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/Paulo"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetUser](#) di Referensi AWS CLI Perintah.

## list-access-keys

Contoh kode berikut menunjukkan cara menggunakan `list-access-keys`.

### AWS CLI

Untuk mencantumkan ID kunci akses untuk pengguna IAM

`list-access-keys` Perintah berikut mencantumkan ID kunci akses untuk pengguna IAM bernama Bob.

```
aws iam list-access-keys \
  --user-name Bob
```

Output:

```
{
  "AccessKeyMetadata": [
    {
      "UserName": "Bob",
      "Status": "Active",
      "CreateDate": "2013-06-04T18:17:34Z",
      "AccessKeyId": "AKIAIOSFODNN7EXAMPLE"
    },
    {
      "UserName": "Bob",
      "Status": "Inactive",
      "CreateDate": "2013-06-06T20:42:26Z",
      "AccessKeyId": "AKIAI44QH8DHBEXAMPLE"
    }
  ]
}
```

Anda tidak dapat mencantumkan kunci akses rahasia untuk pengguna IAM. Jika kunci akses rahasia hilang, Anda harus membuat kunci akses baru menggunakan `create-access-keys` perintah.

Untuk informasi lebih lanjut, lihat [Mengelola access key untuk pengguna IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListAccessKeys](#) di Referensi AWS CLI Perintah.

## **list-account-aliases**

Contoh kode berikut menunjukkan cara menggunakan `list-account-aliases`.

### AWS CLI

Untuk daftar alias akun

`list-account-aliases` Perintah berikut mencantumkan alias untuk akun saat ini.

```
aws iam list-account-aliases
```

Output:

```
{
  "AccountAliases": [
    "mycompany"
  ]
}
```

Untuk informasi selengkapnya, lihat [ID AWS akun Anda dan aliasnya](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListAccountAliases](#) di Referensi AWS CLI Perintah.

## **list-attached-group-policies**

Contoh kode berikut menunjukkan cara menggunakan `list-attached-group-policies`.

### AWS CLI

Untuk mencantumkan semua kebijakan terkelola yang dilampirkan ke grup yang ditentukan

Contoh ini mengembalikan nama dan ARN dari kebijakan terkelola yang dilampirkan ke grup IAM yang disebutkan Admins di AWS akun.

```
aws iam list-attached-group-policies \
  --group-name Admins
```

Output:

```
{
  "AttachedPolicies": [
    {
      "PolicyName": "AdministratorAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
    },
    {
      "PolicyName": "SecurityAudit",
      "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
    }
  ],
  "IsTruncated": false
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListAttachedGroupPolicies](#) di Referensi AWS CLI Perintah.

## list-attached-role-policies

Contoh kode berikut menunjukkan cara menggunakan `list-attached-role-policies`.

### AWS CLI

Untuk mencantumkan semua kebijakan terkelola yang dilampirkan pada peran yang ditentukan

Perintah ini mengembalikan nama dan ARN dari kebijakan terkelola yang dilampirkan pada peran IAM yang disebutkan `SecurityAuditRole` di AWS akun.

```
aws iam list-attached-role-policies \
  --role-name SecurityAuditRole
```

Output:

```
{
  "AttachedPolicies": [
    {
      "PolicyName": "SecurityAudit",
      "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
    }
  ],
}
```

```
"IsTruncated": false
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListAttachedRolePolicies](#) di Referensi AWS CLI Perintah.

## list-attached-user-policies

Contoh kode berikut menunjukkan cara menggunakan `list-attached-user-policies`.

### AWS CLI

Untuk mencantumkan semua kebijakan terkelola yang dilampirkan ke pengguna yang ditentukan

Perintah ini mengembalikan nama dan ARN dari kebijakan terkelola untuk pengguna IAM yang disebutkan Bob di AWS akun.

```
aws iam list-attached-user-policies \
  --user-name Bob
```

### Output:

```
{
  "AttachedPolicies": [
    {
      "PolicyName": "AdministratorAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
    },
    {
      "PolicyName": "SecurityAudit",
      "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
    }
  ],
  "IsTruncated": false
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListAttachedUserPolicies](#) di Referensi AWS CLI Perintah.

## list-entities-for-policy

Contoh kode berikut menunjukkan cara menggunakan `list-entities-for-policy`.

### AWS CLI

Untuk mencantumkan semua pengguna, grup, dan peran yang dilampirkan oleh kebijakan terkelola yang ditentukan

Contoh ini menampilkan daftar grup, peran, dan pengguna IAM yang memiliki kebijakan `arn:aws:iam::123456789012:policy/TestPolicy` terlampir.

```
aws iam list-entities-for-policy \
  --policy-arn arn:aws:iam::123456789012:policy/TestPolicy
```

### Output:

```
{
  "PolicyGroups": [
    {
      "GroupName": "Admins",
      "GroupId": "AGPACKCEVSQ6C2EXAMPLE"
    }
  ],
  "PolicyUsers": [
    {
      "UserName": "Alice",
      "UserId": "AIDACKCEVSQ6C2EXAMPLE"
    }
  ],
  "PolicyRoles": [
    {
      "RoleName": "DevRole",
      "RoleId": "AR0ADBQP57FF2AEXAMPLE"
    }
  ],
  "IsTruncated": false
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListEntitiesForPolicy](#) di Referensi AWS CLI Perintah.



## list-group-policies

Contoh kode berikut menunjukkan cara menggunakan `list-group-policies`.

### AWS CLI

Untuk mencantumkan semua kebijakan inline yang dilampirkan ke grup yang ditentukan

`list-group-policies` Perintah berikut mencantumkan nama-nama kebijakan inline yang dilampirkan ke grup IAM yang disebutkan Admins di akun saat ini.

```
aws iam list-group-policies \  
  --group-name Admins
```

Output:

```
{  
  "PolicyNames": [  
    "AdminRoot",  
    "ExamplePolicy"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola kebijakan IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListGroupPolicies](#) di Referensi AWS CLI Perintah.

## list-groups-for-user

Contoh kode berikut menunjukkan cara menggunakan `list-groups-for-user`.

### AWS CLI

Untuk membuat daftar grup yang dimiliki pengguna IAM

`list-groups-for-user` Perintah berikut menampilkan kelompok-kelompok yang bernama pengguna IAM Bob milik.

```
aws iam list-groups-for-user \  
  --user-name Bob
```

Output:

```
{
  "Groups": [
    {
      "Path": "/",
      "CreateDate": "2013-05-06T01:18:08Z",
      "GroupId": "AKIAIOSFODNN7EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/Admin",
      "GroupName": "Admin"
    },
    {
      "Path": "/",
      "CreateDate": "2013-05-06T01:37:28Z",
      "GroupId": "AKIAI44QH8DHBEXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/s3-Users",
      "GroupName": "s3-Users"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListGroupsWithUsers](#) di Referensi AWS CLI Perintah.

## list-groups

Contoh kode berikut menunjukkan cara menggunakan `list-groups`.

### AWS CLI

Untuk membuat daftar grup IAM untuk akun saat ini

`list-groups` Perintah berikut mencantumkan grup IAM di akun saat ini.

```
aws iam list-groups
```

### Output:

```
{
  "Groups": [
    {
      "Path": "/",
```

```

    "CreateDate": "2013-06-04T20:27:27.972Z",
    "GroupId": "AIDACKCEVSQ6C2EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/Admins",
    "GroupName": "Admins"
  },
  {
    "Path": "/",
    "CreateDate": "2013-04-16T20:30:42Z",
    "GroupId": "AIDGPMS9R04H3FEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/S3-Admins",
    "GroupName": "S3-Admins"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListGroups](#) di Referensi AWS CLI Perintah.

## list-instance-profile-tags

Contoh kode berikut menunjukkan cara menggunakan `list-instance-profile-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke profil instance

`list-instance-profile-tags` Perintah berikut mengambil daftar tag yang terkait dengan profil contoh yang ditentukan.

```
aws iam list-instance-profile-tags \
  --instance-profile-name deployment-role
```

Output:

```

{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {

```

```
        "Key": "Department",
        "Value": "Accounting"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListInstanceProfileTags](#) di Referensi AWS CLI Perintah.

## list-instance-profiles-for-role

Contoh kode berikut menunjukkan cara menggunakan `list-instance-profiles-for-role`.

### AWS CLI

Untuk membuat daftar profil instance untuk peran IAM

`list-instance-profiles-for-role` Perintah berikut mencantumkan profil instance yang terkait dengan peran tersebut `Test-Role`.

```
aws iam list-instance-profiles-for-role \
  --role-name Test-Role
```

### Output:

```
{
  "InstanceProfiles": [
    {
      "InstanceId": "AIDGPM9R04H3FEXAMPLE",
      "Roles": [
        {
          "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
          "RoleId": "AIDACKCEVSQ6C2EXAMPLE",
          "CreateDate": "2013-06-07T20:42:15Z",
          "RoleName": "Test-Role",
          "Path": "/",
          "Arn": "arn:aws:iam::123456789012:role/Test-Role"
        }
      ],
      "CreateDate": "2013-06-07T21:05:24Z",
      "InstanceProfileName": "ExampleInstanceProfile",
    }
  ]
}
```

```
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:instance-profile/
ExampleInstanceProfile"
  }
]
}
```

Untuk informasi lebih lanjut, lihat [Menggunakan profil instance](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListInstanceProfilesForRole](#) di Referensi AWS CLI Perintah.

## list-instance-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-instance-profiles`.

### AWS CLI

Untuk mencantumkan profil instance untuk akun

`list-instance-profiles` Perintah berikut mencantumkan profil instance yang terkait dengan akun saat ini.

```
aws iam list-instance-profiles
```

Output:

```
{
  "InstanceProfiles": [
    {
      "Path": "/",
      "InstanceProfileName": "example-dev-role",
      "InstanceProfileId": "AIPAIXEU4NUHUPEXAMPLE",
      "Arn": "arn:aws:iam::123456789012:instance-profile/example-dev-role",
      "CreateDate": "2023-09-21T18:17:41+00:00",
      "Roles": [
        {
          "Path": "/",
          "RoleName": "example-dev-role",
          "RoleId": "AR0AJ520TH4H7LEXAMPLE",
          "Arn": "arn:aws:iam::123456789012:role/example-dev-role",
          "CreateDate": "2023-09-21T18:17:40+00:00",
          "AssumeRolePolicyDocument": {
```

```
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Principal": {
                    "Service": "ec2.amazonaws.com"
                },
                "Action": "sts:AssumeRole"
            }
        ]
    },
    {
        "Path": "/",
        "InstanceProfileName": "example-s3-role",
        "InstanceProfileId": "AIPAJVJVNRIQFREXAMPLE",
        "Arn": "arn:aws:iam::123456789012:instance-profile/example-s3-role",
        "CreateDate": "2023-09-21T18:18:50+00:00",
        "Roles": [
            {
                "Path": "/",
                "RoleName": "example-s3-role",
                "RoleId": "AROAINUBC507XLEXAMPLE",
                "Arn": "arn:aws:iam::123456789012:role/example-s3-role",
                "CreateDate": "2023-09-21T18:18:49+00:00",
                "AssumeRolePolicyDocument": {
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Effect": "Allow",
                            "Principal": {
                                "Service": "ec2.amazonaws.com"
                            },
                            "Action": "sts:AssumeRole"
                        }
                    ]
                }
            }
        ]
    }
]
```

```
}
```

Untuk informasi lebih lanjut, lihat [Menggunakan profil instance](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListInstanceProfiles](#) di Referensi AWS CLI Perintah.

## **list-mfa-device-tags**

Contoh kode berikut menunjukkan cara menggunakan `list-mfa-device-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke perangkat MFA

`list-mfa-device-tags` Perintah berikut mengambil daftar tag yang terkait dengan perangkat MFA yang ditentukan.

```
aws iam list-mfa-device-tags \
  --serial-number arn:aws:iam::123456789012:mfa/alice
```

Output:

```
{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",
      "Value": "Accounting"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListMfaDeviceTags](#) di Referensi AWS CLI Perintah.

## list-mfa-devices

Contoh kode berikut menunjukkan cara menggunakan `list-mfa-devices`.

### AWS CLI

Untuk mencantumkan semua perangkat MFA untuk pengguna tertentu

Contoh ini mengembalikan rincian tentang perangkat MFA yang ditetapkan untuk pengguna IAM. Bob

```
aws iam list-mfa-devices \
  --user-name Bob
```

### Output:

```
{
  "MFADevices": [
    {
      "UserName": "Bob",
      "SerialNumber": "arn:aws:iam::123456789012:mfa/Bob",
      "EnableDate": "2019-10-28T20:37:09+00:00"
    },
    {
      "UserName": "Bob",
      "SerialNumber": "GAKT12345678",
      "EnableDate": "2023-02-18T21:44:42+00:00"
    },
    {
      "UserName": "Bob",
      "SerialNumber": "arn:aws:iam::123456789012:u2f/user/Bob/
fidosecuritykey1-7XNL7NFNLZ123456789EXAMPLE",
      "EnableDate": "2023-09-19T02:25:35+00:00"
    },
    {
      "UserName": "Bob",
      "SerialNumber": "arn:aws:iam::123456789012:u2f/user/Bob/
fidosecuritykey2-VDRQTDBBN5123456789EXAMPLE",
      "EnableDate": "2023-09-19T01:49:18+00:00"
    }
  ]
}
```



Untuk informasi selengkapnya, lihat [Menggunakan autentikasi multifaktor \(MFA\) dalam AWS](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListMfaDevices](#) di Referensi AWS CLI Perintah.

## **list-open-id-connect-provider-tags**

Contoh kode berikut menunjukkan cara menggunakan `list-open-id-connect-provider-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke penyedia identitas yang kompatibel dengan OpenID Connect (OIDC)

`list-open-id-connect-provider-tags` Perintah berikut mengambil daftar tag yang terkait dengan penyedia identitas OIDC tertentu.

```
aws iam list-open-id-connect-provider-tags \
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com
```

### Output:

```
{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",
      "Value": "Accounting"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListOpenIdConnectProviderTags](#) di Referensi AWS CLI Perintah.

## list-open-id-connect-providers

Contoh kode berikut menunjukkan cara menggunakan `list-open-id-connect-providers`.

### AWS CLI

Untuk mencantumkan informasi tentang penyedia OpenID Connect di akun AWS

Contoh ini mengembalikan daftar ARNS dari semua penyedia OpenID Connect yang didefinisikan dalam akun AWS saat ini.

```
aws iam list-open-id-connect-providers
```

Output:

```
{
  "OpenIDConnectProviderList": [
    {
      "Arn": "arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [ListOpenIdConnectProviders](#) di Referensi AWS CLI Perintah.

## list-policies-granting-service-access

Contoh kode berikut menunjukkan cara menggunakan `list-policies-granting-service-access`.

### AWS CLI

Untuk membuat daftar kebijakan yang memberikan akses utama ke layanan yang ditentukan

`list-policies-granting-service-access` Contoh berikut mengambil daftar kebijakan yang memberikan akses pengguna IAM ke AWS CodeCommit layanan.

```
aws iam list-policies-granting-service-access \  
  --arn arn:aws:iam::123456789012:user/sofia \  
  --service-namespaces codecommit
```

Output:

```
{  
  "PoliciesGrantingServiceAccess": [  
    {  
      "ServiceNamespace": "codecommit",  
      "Policies": [  
        {  
          "PolicyName": "Grant-Sofia-Access-To-CodeCommit",  
          "PolicyType": "INLINE",  
          "EntityType": "USER",  
          "EntityName": "sofia"  
        }  
      ]  
    }  
  ],  
  "IsTruncated": false  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan IAM dengan CodeCommit: Kredensial Git, kunci SSH, dan kunci AWS akses](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [ListPoliciesGrantingServiceAccess](#) di Referensi AWS CLI Perintah.

## list-policies

Contoh kode berikut menunjukkan cara menggunakan `list-policies`.

### AWS CLI

Untuk mencantumkan kebijakan terkelola yang tersedia untuk AWS akun Anda

Contoh ini mengembalikan kumpulan dari dua kebijakan terkelola pertama yang tersedia di AWS akun saat ini.

```
aws iam list-policies \  
  --max-items 3
```

## Output:

```
{
  "Policies": [
    {
      "PolicyName": "AWSCloudTrailAccessPolicy",
      "PolicyId": "ANPAXQE2B5PJ7YEXAMPLE",
      "Arn": "arn:aws:iam::123456789012:policy/AWSCloudTrailAccessPolicy",
      "Path": "/",
      "DefaultVersionId": "v1",
      "AttachmentCount": 0,
      "PermissionsBoundaryUsageCount": 0,
      "IsAttachable": true,
      "CreateDate": "2019-09-04T17:43:42+00:00",
      "UpdateDate": "2019-09-04T17:43:42+00:00"
    },
    {
      "PolicyName": "AdministratorAccess",
      "PolicyId": "ANPAIWMBCKSKIEE64ZLYK",
      "Arn": "arn:aws:iam::aws:policy/AdministratorAccess",
      "Path": "/",
      "DefaultVersionId": "v1",
      "AttachmentCount": 6,
      "PermissionsBoundaryUsageCount": 0,
      "IsAttachable": true,
      "CreateDate": "2015-02-06T18:39:46+00:00",
      "UpdateDate": "2015-02-06T18:39:46+00:00"
    },
    {
      "PolicyName": "PowerUserAccess",
      "PolicyId": "ANPAJYRXTHIB4F0VS3ZXS",
      "Arn": "arn:aws:iam::aws:policy/PowerUserAccess",
      "Path": "/",
      "DefaultVersionId": "v5",
      "AttachmentCount": 1,
      "PermissionsBoundaryUsageCount": 0,
      "IsAttachable": true,
      "CreateDate": "2015-02-06T18:39:47+00:00",
      "UpdateDate": "2023-07-06T22:04:00+00:00"
    }
  ],
  "NextToken": "EXAMPLErZXIi0iBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQi0iA4fQ=="
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListPolicies](#) di Referensi AWS CLI Perintah.

## list-policy-tags

Contoh kode berikut menunjukkan cara menggunakan `list-policy-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan pada kebijakan terkelola

`list-policy-tags` Perintah berikut mengambil daftar tag yang terkait dengan kebijakan terkelola yang ditentukan.

```
aws iam list-policy-tags \  
  --policy-arn arn:aws:iam::123456789012:policy/billing-access
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "DeptID",  
      "Value": "123456"  
    },  
    {  
      "Key": "Department",  
      "Value": "Accounting"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListPolicyTags](#) di Referensi AWS CLI Perintah.

## list-policy-versions

Contoh kode berikut menunjukkan cara menggunakan `list-policy-versions`.

## AWS CLI

Untuk mencantumkan informasi tentang versi kebijakan terkelola yang ditentukan

Contoh ini mengembalikan daftar versi kebijakan yang tersedia yang ARN-nya.

`arn:aws:iam::123456789012:policy/MySamplePolicy`

```
aws iam list-policy-versions \  
  --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Output:

```
{  
  "IsTruncated": false,  
  "Versions": [  
    {  
      "VersionId": "v2",  
      "IsDefaultVersion": true,  
      "CreateDate": "2015-06-02T23:19:44Z"  
    },  
    {  
      "VersionId": "v1",  
      "IsDefaultVersion": false,  
      "CreateDate": "2015-06-02T22:30:47Z"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ListPolicyVersions](#) di Referensi AWS CLI Perintah.

## **list-role-policies**

Contoh kode berikut menunjukkan cara menggunakan `list-role-policies`.

## AWS CLI

Untuk mencantumkan kebijakan yang dilampirkan pada peran IAM

`list-role-policies` Perintah berikut mencantumkan nama kebijakan izin untuk peran IAM yang ditentukan.

```
aws iam list-role-policies \  
  --role-name Test-Role
```

Output:

```
{  
  "PolicyNames": [  
    "ExamplePolicy"  
  ]  
}
```

Untuk melihat kebijakan kepercayaan yang dilampirkan pada peran, gunakan `get-role` perintah. Untuk melihat detail kebijakan izin, gunakan `get-role-policy` perintah.

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListRolePolicies](#) di Referensi AWS CLI Perintah.

## list-role-tags

Contoh kode berikut menunjukkan cara menggunakan `list-role-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke peran

`list-role-tags` Perintah berikut mengambil daftar tag yang terkait dengan peran yang ditentukan.

```
aws iam list-role-tags \  
  --role-name production-role
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Department",  
      "Value": "Accounting"  
    },  
    {
```

```
        "Key": "DeptID",
        "Value": "12345"
    }
],
"IsTruncated": false
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListRoleTags](#) di Referensi AWS CLI Perintah.

## list-roles

Contoh kode berikut menunjukkan cara menggunakan `list-roles`.

### AWS CLI

Untuk mencantumkan peran IAM untuk akun saat ini

`list-roles` Perintah berikut mencantumkan peran IAM untuk akun saat ini.

```
aws iam list-roles
```

Output:

```
{
  "Roles": [
    {
      "Path": "/",
      "RoleName": "ExampleRole",
      "RoleId": "AR0AJ520TH4H7LEXAMPLE",
      "Arn": "arn:aws:iam::123456789012:role/ExampleRole",
      "CreateDate": "2017-09-12T19:23:36+00:00",
      "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
              "Service": "ec2.amazonaws.com"
            }
          }
        ]
      }
    }
  ]
}
```



```

        },
        "Action": "sts:AssumeRole"
    }
]
},
"MaxSessionDuration": 3600
},
{
    "Path": "/example_path/",
    "RoleName": "ExampleRoleWithPath",
    "RoleId": "AROAI4QRP7UFT7EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:role/example_path/
ExampleRoleWithPath",
    "CreateDate": "2023-09-21T20:29:38+00:00",
    "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "",
                "Effect": "Allow",
                "Principal": {
                    "Service": "ec2.amazonaws.com"
                },
                "Action": "sts:AssumeRole"
            }
        ]
    },
    "MaxSessionDuration": 3600
}
]
}

```

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListRoles](#) di Referensi AWS CLI Perintah.

## list-saml-provider-tags

Contoh kode berikut menunjukkan cara menggunakan `list-saml-provider-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke penyedia SAFL

`list-saml-provider-tags` Perintah berikut mengambil daftar tag yang terkait dengan penyedia SAFL tertentu.

```
aws iam list-saml-provider-tags \
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS
```

Output:

```
{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",
      "Value": "Accounting"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListSamlProviderTags](#) di Referensi AWS CLI Perintah.

## `list-saml-providers`

Contoh kode berikut menunjukkan cara menggunakan `list-saml-providers`.

AWS CLI

Untuk membuat daftar penyedia SAFL di akun AWS

Contoh ini mengambil daftar penyedia SAFL 2.0 yang dibuat di akun saat ini AWS .

```
aws iam list-saml-providers
```

Output:

```
{
```

```
"SAMLProviderList": [  
  {  
    "Arn": "arn:aws:iam::123456789012:saml-provider/SAML-ADFS",  
    "ValidUntil": "2015-06-05T22:45:14Z",  
    "CreateDate": "2015-06-05T22:45:14Z"  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas IAM SALL](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListSamlProviders](#) di AWS CLI Referensi Perintah.

## list-server-certificate-tags

Contoh kode berikut menunjukkan cara menggunakan `list-server-certificate-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke sertifikat server

`list-server-certificate-tags` Perintah berikut mengambil daftar tag yang terkait dengan sertifikat server yang ditentukan.

```
aws iam list-server-certificate-tags \  
  --server-certificate-name ExampleCertificate
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "DeptID",  
      "Value": "123456"  
    },  
    {  
      "Key": "Department",  
      "Value": "Accounting"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListServerCertificateTags](#) di Referensi AWS CLI Perintah.

## list-server-certificates

Contoh kode berikut menunjukkan cara menggunakan `list-server-certificates`.

### AWS CLI

Untuk mencantumkan sertifikat server di AWS akun Anda

`list-server-certificates` Perintah berikut mencantumkan semua sertifikat server yang disimpan dan tersedia untuk digunakan di AWS akun Anda.

```
aws iam list-server-certificates
```

Output:

```
{
  "ServerCertificateMetadataList": [
    {
      "Path": "/",
      "ServerCertificateName": "myUpdatedServerCertificate",
      "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:server-certificate/myUpdatedServerCertificate",
      "UploadDate": "2019-04-22T21:13:44+00:00",
      "Expiration": "2019-10-15T22:23:16+00:00"
    },
    {
      "Path": "/cloudfront/",
      "ServerCertificateName": "MyTestCert",
      "ServerCertificateId": "ASCAEXAMPLE456EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:server-certificate/Org1/Org2/MyTestCert",
      "UploadDate": "2015-04-21T18:14:16+00:00",
      "Expiration": "2018-01-14T17:52:36+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola sertifikat server di IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListServerCertificates](#) di Referensi AWS CLI Perintah.

## **list-service-specific-credential**

Contoh kode berikut menunjukkan cara menggunakan `list-service-specific-credential`.

### AWS CLI

Contoh 1: Buat daftar kredensi khusus layanan untuk pengguna

`list-service-specific-credentials` Contoh berikut menampilkan semua kredensi khusus layanan yang ditetapkan untuk pengguna tertentu. Kata sandi tidak termasuk dalam respons.

```
aws iam list-service-specific-credentials \  
  --user-name sofia
```

Output:

```
{  
  "ServiceSpecificCredential": {  
    "CreateDate": "2019-04-18T20:45:36+00:00",  
    "ServiceName": "codecommit.amazonaws.com",  
    "ServiceUserName": "sofia-at-123456789012",  
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",  
    "UserName": "sofia",  
    "Status": "Active"  
  }  
}
```

Contoh 2: Buat daftar kredensi khusus layanan untuk pengguna yang difilter ke layanan tertentu

`list-service-specific-credentials` Contoh berikut menampilkan kredensi khusus layanan yang ditetapkan kepada pengguna yang membuat permintaan. Daftar ini disaring untuk menyertakan hanya kredensi tersebut untuk layanan yang ditentukan. Kata sandi tidak termasuk dalam respons.

```
aws iam list-service-specific-credentials \  
  --service-name codecommit.amazonaws.com
```

**Output:**

```
{
  "ServiceSpecificCredential": {
    "CreateDate": "2019-04-18T20:45:36+00:00",
    "ServiceName": "codecommit.amazonaws.com",
    "ServiceUserName": "sofia-at-123456789012",
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
    "UserName": "sofia",
    "Status": "Active"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat kredensi Git untuk koneksi HTTPS CodeCommit](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [ListServiceSpecificCredential](#) di Referensi AWS CLI Perintah.

**list-service-specific-credentials**

Contoh kode berikut menunjukkan cara menggunakan `list-service-specific-credentials`.

**AWS CLI**

Untuk mengambil daftar kredensial

`list-service-specific-credentials` Contoh berikut mencantumkan kredensial yang dihasilkan untuk akses HTTPS ke AWS CodeCommit repositori untuk pengguna bernama `developer`

```
aws iam list-service-specific-credentials \
  --user-name developer \
  --service-name codecommit.amazonaws.com
```

**Output:**

```
{
  "ServiceSpecificCredentials": [
    {
      "UserName": "developer",
      "Status": "Inactive",
    }
  ]
}
```

```

    "ServiceUserName": "developer-at-123456789012",
    "CreateDate": "2019-10-01T04:31:41Z",
    "ServiceSpecificCredentialId": "ACCAQF0DXMPL4YFHP7DZE",
    "ServiceName": "codecommit.amazonaws.com"
  },
  {
    "UserName": "developer",
    "Status": "Active",
    "ServiceUserName": "developer+1-at-123456789012",
    "CreateDate": "2019-10-01T04:31:45Z",
    "ServiceSpecificCredentialId": "ACCAQF0XMPL6VW57M7AJP",
    "ServiceName": "codecommit.amazonaws.com"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat kredensi Git untuk koneksi HTTPS CodeCommit](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [ListServiceSpecificCredentials](#) di Referensi AWS CLI Perintah.

## list-signing-certificates

Contoh kode berikut menunjukkan cara menggunakan `list-signing-certificates`.

### AWS CLI

Untuk mencantumkan sertifikat penandatanganan untuk pengguna IAM

`list-signing-certificates` Perintah berikut mencantumkan sertifikat penandatanganan untuk pengguna IAM bernama Bob.

```
aws iam list-signing-certificates \
  --user-name Bob
```

Output:

```

{
  "Certificates": [
    {
      "UserName": "Bob",
      "Status": "Inactive",

```

```

        "CertificateBody": "-----BEGIN CERTIFICATE-----<certificate-body>-----
END CERTIFICATE-----",
        "CertificateId": "TA7SMP42TDN5Z260BPJE7EXAMPLE",
        "UploadDate": "2013-06-06T21:40:08Z"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola sertifikat penandatanganan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [ListSigningCertificates](#) di Referensi AWS CLI Perintah.

## list-ssh-public-keys

Contoh kode berikut menunjukkan cara menggunakan `list-ssh-public-keys`.

### AWS CLI

Untuk mencantumkan kunci publik SSH yang dilampirkan ke pengguna IAM

`list-ssh-public-keys` Contoh berikut mencantumkan kunci publik SSH yang dilampirkan ke pengguna IAM. `sofia`

```
aws iam list-ssh-public-keys \
  --user-name sofia
```

Output:

```
{
  "SSHPublicKeys": [
    {
      "UserName": "sofia",
      "SSHPublicKeyId": "APKA1234567890EXAMPLE",
      "Status": "Inactive",
      "UploadDate": "2019-04-18T17:04:49+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan kunci SSH dan SSH dengan CodeCommit](#) dalam Panduan Pengguna AWS IAM



- Untuk detail API, lihat [ListSshPublicKeys](#) di Referensi AWS CLI Perintah.

## **list-user-policies**

Contoh kode berikut menunjukkan cara menggunakan `list-user-policies`.

### AWS CLI

Untuk membuat daftar kebijakan untuk pengguna IAM

`list-user-policies` Perintah berikut mencantumkan kebijakan yang dilampirkan ke nama Bob pengguna IAM.

```
aws iam list-user-policies \  
  --user-name Bob
```

Output:

```
{  
  "PolicyNames": [  
    "ExamplePolicy",  
    "TestPolicy"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di AWS akun Anda](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListUserPolicies](#) di Referensi AWS CLI Perintah.

## **list-user-tags**

Contoh kode berikut menunjukkan cara menggunakan `list-user-tags`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke pengguna

`list-user-tags` Perintah berikut mengambil daftar tag yang terkait dengan pengguna IAM tertentu.

```
aws iam list-user-tags \  
  --user-name alice
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "Department",  
      "Value": "Accounting"  
    },  
    {  
      "Key": "DeptID",  
      "Value": "12345"  
    }  
  ],  
  "IsTruncated": false  
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListUserTags](#) di Referensi AWS CLI Perintah.

## list-users

Contoh kode berikut menunjukkan cara menggunakan `list-users`.

### AWS CLI

Untuk daftar pengguna IAM

`list-users` Perintah berikut mencantumkan pengguna IAM di akun saat ini.

```
aws iam list-users
```

Output:

```
{  
  "Users": [  
    {  
      "UserName": "alice"  
    }  
  ]  
}
```

```
{
  "UserName": "Adele",
  "Path": "/",
  "CreateDate": "2013-03-07T05:14:48Z",
  "UserId": "AKIAI44QH8DHBEXAMPLE",
  "Arn": "arn:aws:iam::123456789012:user/Adele"
},
{
  "UserName": "Bob",
  "Path": "/",
  "CreateDate": "2012-09-21T23:03:13Z",
  "UserId": "AKIAIOSFODNN7EXAMPLE",
  "Arn": "arn:aws:iam::123456789012:user/Bob"
}
]
```

Untuk informasi selengkapnya, lihat [Daftar pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListUsers](#) di Referensi AWS CLI Perintah.

## list-virtual-mfa-devices

Contoh kode berikut menunjukkan cara menggunakan `list-virtual-mfa-devices`.

### AWS CLI

Untuk daftar perangkat MFA virtual

`list-virtual-mfa-devices` Perintah berikut mencantumkan perangkat MFA virtual yang telah dikonfigurasi untuk akun saat ini.

```
aws iam list-virtual-mfa-devices
```

Output:

```
{
  "VirtualMFADevices": [
    {
      "SerialNumber": "arn:aws:iam::123456789012:mfa/ExampleMFADevice"
    },
    {
```

```
        "SerialNumber": "arn:aws:iam::123456789012:mfa/Fred"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan perangkat autentikasi multi-faktor virtual \(MFA\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [ListVirtualMfaDevices](#) di Referensi AWS CLI Perintah.

## put-group-policy

Contoh kode berikut menunjukkan cara menggunakan `put-group-policy`.

### AWS CLI

Untuk menambahkan kebijakan ke grup

`put-group-policy` Perintah berikut menambahkan kebijakan ke grup IAM bernama `Admins`.

```
aws iam put-group-policy \
  --group-name Admins \
  --policy-document file://AdminPolicy.json \
  --policy-name AdminRoot
```

Perintah ini tidak menghasilkan output.

Kebijakan didefinisikan sebagai dokumen JSON dalam `AdminPolicyfile.json`. (Nama file dan ekstensi tidak memiliki signifikansi.)

Untuk informasi selengkapnya, lihat [Mengelola kebijakan IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [PutGroupPolicy](#) di Referensi AWS CLI Perintah.

## put-role-permissions-boundary

Contoh kode berikut menunjukkan cara menggunakan `put-role-permissions-boundary`.

### AWS CLI

Contoh 1: Untuk menerapkan batas izin berdasarkan kebijakan kustom ke peran IAM

`put-role-permissions-boundary` Contoh berikut menerapkan kebijakan kustom yang dinamai `intern-boundary` sebagai batas izin untuk peran IAM yang ditentukan.

```
aws iam put-role-permissions-boundary \  
  --permissions-boundary arn:aws:iam::123456789012:policy/intern-boundary \  
  --role-name lambda-application-role
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menerapkan batas izin berdasarkan kebijakan AWS terkelola ke peran IAM

`put-role-permissions-boundary` Contoh berikut menerapkan `PowerUserAccess` kebijakan AWS terkelola sebagai batas izin untuk peran IAM yang ditentukan.

```
aws iam put-role-permissions-boundary \  
  --permissions-boundary arn:aws:iam::aws:policy/PowerUserAccess \  
  --role-name x-account-admin
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memodifikasi peran](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [PutRolePermissionsBoundary](#) di Referensi AWS CLI Perintah.

## **put-role-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-role-policy`.

### AWS CLI

Untuk melampirkan kebijakan izin ke peran IAM

`put-role-policy` Perintah berikut menambahkan kebijakan izin ke peran bernama `Test-Role`.

```
aws iam put-role-policy \  
  --role-name Test-Role \  
  --policy-name ExamplePolicy \  
  --policy-document file://AdminPolicy.json
```

Perintah ini tidak menghasilkan output.

Kebijakan didefinisikan sebagai dokumen JSON dalam AdminPolicyfile.json. (Nama file dan ekstensi tidak memiliki signifikansi.)

Untuk melampirkan kebijakan kepercayaan ke peran, gunakan `update-assume-role-policy` perintah.

Untuk informasi selengkapnya, lihat [Memodifikasi peran](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [PutRolePolicy](#) di Referensi AWS CLI Perintah.

## **put-user-permissions-boundary**

Contoh kode berikut menunjukkan cara menggunakan `put-user-permissions-boundary`.

### AWS CLI

Contoh 1: Untuk menerapkan batas izin berdasarkan kebijakan kustom ke pengguna IAM

`put-user-permissions-boundary` Contoh berikut menerapkan kebijakan kustom bernama `intern-boundary` sebagai batas izin untuk pengguna IAM yang ditentukan.

```
aws iam put-user-permissions-boundary \  
  --permissions-boundary arn:aws:iam::123456789012:policy/intern-boundary \  
  --user-name intern
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menerapkan batas izin berdasarkan kebijakan AWS terkelola ke pengguna IAM

`put-user-permissions-boundary` Contoh berikut menerapkan policy AWS terkelola bernama `PowerUserAccess` sebagai batas izin untuk pengguna IAM yang ditentukan.

```
aws iam put-user-permissions-boundary \  
  --permissions-boundary arn:aws:iam::aws:policy/PowerUserAccess \  
  --user-name developer
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menambahkan dan menghapus izin identitas IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [PutUserPermissionsBoundary](#) di Referensi AWS CLI Perintah.

## **put-user-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-user-policy`.

### AWS CLI

Untuk melampirkan kebijakan ke pengguna IAM

`put-user-policy` Perintah berikut melampirkan kebijakan ke pengguna IAM bernama. Bob

```
aws iam put-user-policy \  
  --user-name Bob \  
  --policy-name ExamplePolicy \  
  --policy-document file://AdminPolicy.json
```

Perintah ini tidak menghasilkan output.

Kebijakan didefinisikan sebagai dokumen JSON dalam `AdminPolicyfile.json`. (Nama file dan ekstensi tidak memiliki signifikansi.)

Untuk informasi lebih lanjut, lihat [Menambahkan dan menghapus izin identitas IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [PutUserPolicy](#) di Referensi AWS CLI Perintah.

## **remove-client-id-from-open-id-connect-provider**

Contoh kode berikut menunjukkan cara menggunakan `remove-client-id-from-open-id-connect-provider`.

### AWS CLI

Untuk menghapus ID klien yang ditentukan dari daftar ID klien yang terdaftar untuk penyedia IAM OpenID Connect yang ditentukan

Contoh ini menghapus ID klien `My-TestApp-3` dari daftar ID klien yang terkait dengan penyedia IAM OIDC yang ARN. `arn:aws:iam::123456789012:oidc-provider/example.oidcprovider.com`

```
aws iam remove-client-id-from-open-id-connect-provider
  --client-id My-TestApp-3 \
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [RemoveClientIdFromOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## **remove-role-from-instance-profile**

Contoh kode berikut menunjukkan cara menggunakan `remove-role-from-instance-profile`.

### AWS CLI

Untuk menghapus peran dari profil instance

`remove-role-from-instance-profile` Perintah berikut menghapus peran bernama `Test-Role` dari profil instance bernama `ExampleInstanceProfile`.

```
aws iam remove-role-from-instance-profile \
  --instance-profile-name ExampleInstanceProfile \
  --role-name Test-Role
```

Untuk informasi lebih lanjut, lihat [Menggunakan profil instance](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [RemoveRoleFromInstanceProfile](#) di Referensi AWS CLI Perintah.

## **remove-user-from-group**

Contoh kode berikut menunjukkan cara menggunakan `remove-user-from-group`.

### AWS CLI

Untuk menghapus pengguna dari grup IAM



`remove-user-from-group` Perintah berikut menghapus nama pengguna Bob dari grup IAM bernama Admins.

```
aws iam remove-user-from-group \  
  --user-name Bob \  
  --group-name Admins
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus pengguna dalam grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [RemoveUserFromGroup](#) di Referensi AWS CLI Perintah.

## **reset-service-specific-credential**

Contoh kode berikut menunjukkan cara menggunakan `reset-service-specific-credential`.

### AWS CLI

Contoh 1: Setel ulang kata sandi untuk kredensi khusus layanan yang dilampirkan pada pengguna yang membuat permintaan

`reset-service-specific-credential` Contoh berikut menghasilkan kata sandi baru yang kuat secara kriptografis untuk kredensi khusus layanan tertentu yang dilampirkan pada pengguna yang membuat permintaan.

```
aws iam reset-service-specific-credential \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Output:

```
{  
  "ServiceSpecificCredential": {  
    "CreateDate": "2019-04-18T20:45:36+00:00",  
    "ServiceName": "codecommit.amazonaws.com",  
    "ServiceUserName": "sofia-at-123456789012",  
    "ServicePassword": "+oaFsNk7tLco+C/obP9Ghhc0zGcK0ayTmE3LnAmAmH4=",  
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",  
    "UserName": "sofia",  
    "Status": "Active"  
  }  
}
```

```
}
```

Contoh 2: Setel ulang kata sandi untuk kredensi khusus layanan yang dilampirkan ke pengguna tertentu

`reset-service-specific-credential` Contoh berikut menghasilkan kata sandi baru yang kuat secara kriptografis untuk kredensi khusus layanan yang dilampirkan pada pengguna tertentu.

```
aws iam reset-service-specific-credential \  
  --user-name sofia \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Output:

```
{  
  "ServiceSpecificCredential": {  
    "CreateDate": "2019-04-18T20:45:36+00:00",  
    "ServiceName": "codecommit.amazonaws.com",  
    "ServiceUserName": "sofia-at-123456789012",  
    "ServicePassword": "+oaFsNk7tLco+C/obP9GhhcOzGcK0ayTmE3LnAmAmH4=",  
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",  
    "UserName": "sofia",  
    "Status": "Active"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat kredensi Git untuk koneksi HTTPS CodeCommit](#) di AWS CodeCommit Panduan Pengguna.

- Untuk detail API, lihat [ResetServiceSpecificCredential](#) di Referensi AWS CLI Perintah.

## **resync-mfa-device**

Contoh kode berikut menunjukkan cara menggunakan `resync-mfa-device`.

### AWS CLI

Untuk menyinkronkan perangkat MFA

`resync-mfa-device` Contoh berikut menyinkronkan perangkat MFA yang terkait dengan pengguna IAM Bob dan ARN-nya dengan program autentikator yang `arn:aws:iam::123456789012:mfa/BobsMFADevice` menyediakan dua kode otentikasi.

```
aws iam resync-mfa-device \  
  --user-name Bob \  
  --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice \  
  --authentication-code1 123456 \  
  --authentication-code2 987654
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan autentikasi multifaktor \(MFA\) dalam AWS](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [ResyncMfaDevice](#) di Referensi AWS CLI Perintah.

## set-default-policy-version

Contoh kode berikut menunjukkan cara menggunakan `set-default-policy-version`.

### AWS CLI

Untuk menetapkan versi tertentu dari kebijakan yang ditentukan sebagai versi default kebijakan.

Contoh ini menetapkan v2 versi kebijakan yang ARN-nya `arn:aws:iam::123456789012:policy/MyPolicy` sebagai versi aktif default.

```
aws iam set-default-policy-version \  
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \  
  --version-id v2
```

Untuk informasi selengkapnya, lihat [Kebijakan dan izin di IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [SetDefaultPolicyVersion](#) di Referensi AWS CLI Perintah.

## set-security-token-service-preferences

Contoh kode berikut menunjukkan cara menggunakan `set-security-token-service-preferences`.

### AWS CLI

Untuk mengatur versi token titik akhir global

`set-security-token-service-preferences` Contoh berikut mengonfigurasi Amazon STS untuk menggunakan token versi 2 saat Anda mengautentikasi terhadap titik akhir global.

```
aws iam set-security-token-service-preferences \
  --global-endpoint-token-version v2Token
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola AWS STS di AWS Wilayah](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [SetSecurityTokenServicePreferences](#) di Referensi AWS CLI Perintah.

## **simulate-custom-policy**

Contoh kode berikut menunjukkan cara menggunakan `simulate-custom-policy`.

### AWS CLI

Contoh 1: Untuk mensimulasikan efek dari semua kebijakan IAM yang terkait dengan pengguna atau peran IAM

Berikut ini `simulate-custom-policy` menunjukkan cara menyediakan kebijakan dan menentukan nilai variabel dan mensimulasikan panggilan API untuk melihat apakah itu diizinkan atau ditolak. Contoh berikut menunjukkan kebijakan yang memungkinkan akses database hanya setelah tanggal dan waktu tertentu. Simulasi berhasil karena tindakan simulasi dan `aws:CurrentTime` variabel yang ditentukan semuanya sesuai dengan persyaratan kebijakan.

```
aws iam simulate-custom-policy \
  --policy-input-list '{"Version":"2012-10-17","Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"*","Condition":
{"DateGreaterThan":{"aws:CurrentTime":"2018-08-16T12:00:00Z"}}}' \
  --action-names dynamodb>CreateBackup \
  --context-entries
"ContextKeyName='aws:CurrentTime',ContextKeyValues='2019-04-25T11:00:00Z',ContextKeyType=da
```

Output:

```
{
  "EvaluationResults": [
```

```

    {
      "EvalActionName": "dynamodb:CreateBackup",
      "EvalResourceName": "*",
      "EvalDecision": "allowed",
      "MatchedStatements": [
        {
          "SourcePolicyId": "PolicyInputList.1",
          "StartPosition": {
            "Line": 1,
            "Column": 38
          },
          "EndPosition": {
            "Line": 1,
            "Column": 167
          }
        }
      ],
      "MissingContextValues": []
    }
  ]
}

```

Contoh 2: Untuk mensimulasikan perintah yang dilarang oleh kebijakan

`simulate-custom-policy` Contoh berikut menunjukkan hasil simulasi perintah yang dilarang oleh kebijakan. Dalam contoh ini, tanggal yang diberikan adalah sebelum yang disyaratkan oleh kondisi polis.

```

aws iam simulate-custom-policy \
  --policy-input-list '{"Version":"2012-10-17","Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"*","Condition":
{"DateGreaterThan":{"aws:CurrentTime":"2018-08-16T12:00:00Z"}}}' \
  --action-names dynamodb:CreateBackup \
  --context-entries
"ContextKeyName='aws:CurrentTime',ContextKeyValues='2014-04-25T11:00:00Z',ContextKeyType=da

```

Output:

```

{
  "EvaluationResults": [
    {
      "EvalActionName": "dynamodb:CreateBackup",

```

```

        "EvalResourceName": "*",
        "EvalDecision": "implicitDeny",
        "MatchedStatements": [],
        "MissingContextValues": []
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menguji kebijakan IAM dengan simulator kebijakan IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [SimulateCustomPolicy](#) di Referensi AWS CLI Perintah.

## **simulate-principal-policy**

Contoh kode berikut menunjukkan cara menggunakan `simulate-principal-policy`.

### AWS CLI

Contoh 1: Untuk mensimulasikan efek dari kebijakan IAM yang sewenang-wenang

Berikut ini `simulate-principal-policy` menunjukkan cara mensimulasikan pengguna yang memanggil tindakan API dan menentukan apakah kebijakan yang terkait dengan pengguna tersebut mengizinkan atau menolak tindakan tersebut. Dalam contoh berikut, pengguna memiliki kebijakan yang hanya mengizinkan `codecommit:ListRepositories` tindakan.

```

aws iam simulate-principal-policy \
  --policy-source-arn arn:aws:iam::123456789012:user/alejandro \
  --action-names codecommit:ListRepositories

```

### Output:

```

{
  "EvaluationResults": [
    {
      "EvalActionName": "codecommit:ListRepositories",
      "EvalResourceName": "*",
      "EvalDecision": "allowed",
      "MatchedStatements": [
        {
          "SourcePolicyId": "Grant-Access-To-CodeCommit-ListRepo",
          "StartPosition": {

```

```

        "Line": 3,
        "Column": 19
      },
      "EndPosition": {
        "Line": 9,
        "Column": 10
      }
    }
  ],
  "MissingContextValues": []
}
]
}

```

Contoh 2: Untuk mensimulasikan efek dari perintah terlarang

`simulate-custom-policy` Contoh berikut menunjukkan hasil simulasi perintah yang dilarang oleh salah satu kebijakan pengguna. Dalam contoh berikut, pengguna memiliki kebijakan yang mengizinkan akses ke database DynamoDB hanya setelah tanggal dan waktu tertentu. Simulasi memiliki pengguna yang mencoba mengakses database dengan `aws:CurrentTime` nilai yang lebih awal dari izin kondisi kebijakan.

```

aws iam simulate-principal-policy \
  --policy-source-arn arn:aws:iam::123456789012:user/alejandro \
  --action-names dynamodb:CreateBackup \
  --context-entries
  "ContextKeyName='aws:CurrentTime',ContextKeyValues='2018-04-25T11:00:00Z',ContextKeyType=da

```

Output:

```

{
  "EvaluationResults": [
    {
      "EvalActionName": "dynamodb:CreateBackup",
      "EvalResourceName": "*",
      "EvalDecision": "implicitDeny",
      "MatchedStatements": [],
      "MissingContextValues": []
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Menguji kebijakan IAM dengan simulator kebijakan IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [SimulatePrincipalPolicy](#) di Referensi AWS CLI Perintah.

## tag-instance-profile

Contoh kode berikut menunjukkan cara menggunakan `tag-instance-profile`.

### AWS CLI

Untuk menambahkan tag ke profil instance

`tag-instance-profile` Perintah berikut menambahkan tag dengan nama Departemen ke profil instance tertentu.

```
aws iam tag-instance-profile \  
  --instance-profile-name deployment-role \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagInstanceProfile](#) di Referensi AWS CLI Perintah.

## tag-mfa-device

Contoh kode berikut menunjukkan cara menggunakan `tag-mfa-device`.

### AWS CLI

Untuk menambahkan tag ke perangkat MFA

`tag-mfa-device` Perintah berikut menambahkan tag dengan nama Departemen ke perangkat MFA yang ditentukan.

```
aws iam tag-mfa-device \  
  --serial-number arn:aws:iam::123456789012:mfa/alice \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```



Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagMfaDevice](#) di Referensi AWS CLI Perintah.

## **tag-open-id-connect-provider**

Contoh kode berikut menunjukkan cara menggunakan `tag-open-id-connect-provider`.

### AWS CLI

Untuk menambahkan tag ke penyedia identitas yang kompatibel dengan OpenID Connect (OIDC)

`tag-open-id-connect-provider` Perintah berikut menambahkan tag dengan nama Departemen ke penyedia identitas OIDC tertentu.

```
aws iam tag-open-id-connect-provider \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
server.example.com \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## **tag-policy**

Contoh kode berikut menunjukkan cara menggunakan `tag-policy`.

### AWS CLI

Untuk menambahkan tag ke kebijakan terkelola pelanggan

`tag-policy` Perintah berikut menambahkan tag dengan nama Departemen ke kebijakan terkelola pelanggan yang ditentukan.

```
aws iam tag-policy \  
  --policy-arn arn:aws:iam::123456789012:policy/billing-access \  

```

```
--tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagPolicy](#) di Referensi AWS CLI Perintah.

## tag-role

Contoh kode berikut menunjukkan cara menggunakan `tag-role`.

### AWS CLI

Untuk menambahkan tag ke peran

`tag-role` Perintah berikut menambahkan tag dengan nama Departemen ke peran yang ditentukan.

```
aws iam tag-role --role-name my-role \  
  --tags '{"Key": "Department", "Value": "Accounting"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagRole](#) di Referensi AWS CLI Perintah.

## tag-saml-provider

Contoh kode berikut menunjukkan cara menggunakan `tag-saml-provider`.

### AWS CLI

Untuk menambahkan tag ke penyedia SAFL

`tag-saml-provider` Perintah berikut menambahkan tag dengan nama Departemen ke penyedia SAFL tertentu.

```
aws iam tag-saml-provider \  
  --role-name my-role \  
  --tags '{"Key": "Department", "Value": "Accounting"}'
```

```
--saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS \  
--tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagSamlProvider](#) di Referensi AWS CLI Perintah.

## tag-server-certificate

Contoh kode berikut menunjukkan cara menggunakan `tag-server-certificate`.

### AWS CLI

Untuk menambahkan tag ke sertifikat server

`tag-saml-provider` Perintah berikut menambahkan tag dengan nama Departemen ke sertifikat sever yang ditentukan.

```
aws iam tag-server-certificate \  
--server-certificate-name ExampleCertificate \  
--tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagServerCertificate](#) di Referensi AWS CLI Perintah.

## tag-user

Contoh kode berikut menunjukkan cara menggunakan `tag-user`.

### AWS CLI

Untuk menambahkan tag ke pengguna

`tag-user` Perintah berikut menambahkan tag dengan Departemen terkait untuk pengguna tertentu.

```
aws iam tag-user \  
  --user-name alice \  
  --tags '{"Key": "Department", "Value": "Accounting"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagUser](#) di Referensi AWS CLI Perintah.

## untag-instance-profile

Contoh kode berikut menunjukkan cara menggunakan `untag-instance-profile`.

### AWS CLI

Untuk menghapus tag dari profil instance

`untag-instance-profile` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari profil instance yang ditentukan.

```
aws iam untag-instance-profile \  
  --instance-profile-name deployment-role \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagInstanceProfile](#) di Referensi AWS CLI Perintah.

## untag-mfa-device

Contoh kode berikut menunjukkan cara menggunakan `untag-mfa-device`.

### AWS CLI

Untuk menghapus tag dari perangkat MFA

`untag-mfa-device` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari perangkat MFA yang ditentukan.

```
aws iam untag-mfa-device \  
  --serial-number arn:aws:iam::123456789012:mfa/alice \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagMfaDevice](#) di Referensi AWS CLI Perintah.

## **untag-open-id-connect-provider**

Contoh kode berikut menunjukkan cara menggunakan `untag-open-id-connect-provider`.

### AWS CLI

Untuk menghapus tag dari penyedia identitas OIDC

`untag-open-id-connect-provider` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari penyedia identitas OIDC yang ditentukan.

```
aws iam untag-open-id-connect-provider \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
server.example.com \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagOpenIdConnectProvider](#) di Referensi AWS CLI Perintah.

## **untag-policy**

Contoh kode berikut menunjukkan cara menggunakan `untag-policy`.

## AWS CLI

Untuk menghapus tag dari kebijakan terkelola pelanggan

`untag-policy` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari kebijakan terkelola pelanggan yang ditentukan.

```
aws iam untag-policy \  
  --policy-arn arn:aws:iam::452925170507:policy/billing-access \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagPolicy](#) di Referensi AWS CLI Perintah.

## **untag-role**

Contoh kode berikut menunjukkan cara menggunakan `untag-role`.

## AWS CLI

Untuk menghapus tag dari peran

`untag-role` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari peran yang ditentukan.

```
aws iam untag-role \  
  --role-name my-role \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagRole](#) di Referensi AWS CLI Perintah.

## **untag-saml-provider**

Contoh kode berikut menunjukkan cara menggunakan `untag-saml-provider`.

## AWS CLI

Untuk menghapus tag dari penyedia SAFL

`untag-saml-provider` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari profil instance yang ditentukan.

```
aws iam untag-saml-provider \  
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagSamlProvider](#) di Referensi AWS CLI Perintah.

## **untag-server-certificate**

Contoh kode berikut menunjukkan cara menggunakan `untag-server-certificate`.

## AWS CLI

Untuk menghapus tag dari sertifikat server

`untag-server-certificate` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari sertifikat server yang ditentukan.

```
aws iam untag-server-certificate \  
  --server-certificate-name ExampleCertificate \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagServerCertificate](#) di Referensi AWS CLI Perintah.

## **untag-user**

Contoh kode berikut menunjukkan cara menggunakan `untag-user`.

## AWS CLI

Untuk menghapus tag dari pengguna

`untag-user` Perintah berikut menghapus tag apa pun dengan nama kunci 'Departemen' dari pengguna yang ditentukan.

```
aws iam untag-user \  
  --user-name alice \  
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagUser](#) di Referensi AWS CLI Perintah.

## update-access-key

Contoh kode berikut menunjukkan cara menggunakan `update-access-key`.

## AWS CLI

Untuk mengaktifkan atau menonaktifkan kunci akses untuk pengguna IAM

`update-access-key` Perintah berikut menonaktifkan kunci akses yang ditentukan (ID kunci akses dan kunci akses rahasia) untuk pengguna IAM bernama. Bob

```
aws iam update-access-key \  
  --access-key-id AKIAIOSFODNN7EXAMPLE \  
  --status Inactive \  
  --user-name Bob
```

Perintah ini tidak menghasilkan output.

Menonaktifkan kunci berarti tidak dapat digunakan untuk akses terprogram ke. AWS Namun, kuncinya masih tersedia dan dapat diaktifkan kembali.

Untuk informasi lebih lanjut, lihat [Mengelola access key untuk pengguna IAM](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [UpdateAccessKey](#) di Referensi AWS CLI Perintah.



## update-account-password-policy

Contoh kode berikut menunjukkan cara menggunakan `update-account-password-policy`.

### AWS CLI

Untuk mengatur atau mengubah kebijakan kata sandi akun saat ini

`update-account-password-policy` Perintah berikut menetapkan kebijakan kata sandi untuk memerlukan panjang minimum delapan karakter dan memerlukan satu atau lebih angka dalam kata sandi.

```
aws iam update-account-password-policy \  
  --minimum-password-length 8 \  
  --require-numbers
```

Perintah ini tidak menghasilkan output.

Perubahan pada kebijakan kata sandi akun memengaruhi kata sandi baru apa pun yang dibuat untuk pengguna IAM di akun. Perubahan kebijakan kata sandi tidak memengaruhi kata sandi yang ada.

Untuk informasi selengkapnya, lihat [Menyetel kebijakan kata sandi akun untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateAccountPasswordPolicy](#) di Referensi AWS CLI Perintah.

## update-assume-role-policy

Contoh kode berikut menunjukkan cara menggunakan `update-assume-role-policy`.

### AWS CLI

Untuk memperbarui kebijakan kepercayaan untuk peran IAM

`update-assume-role-policy` Perintah berikut memperbarui kebijakan kepercayaan untuk peran bernama `Test-Role`.

```
aws iam update-assume-role-policy \  
  --role-name Test-Role \  
  --policy-document file://Test-Role-Trust-Policy.json
```

Perintah ini tidak menghasilkan output.

Kebijakan kepercayaan didefinisikan sebagai dokumen JSON dalam file `test-role-trust-policy.json`. (Nama file dan ekstensi tidak memiliki signifikansi.) Kebijakan kepercayaan harus menentukan kepala sekolah.

Untuk memperbarui kebijakan izin untuk peran, gunakan `put-role-policy` perintah.

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateAssumeRolePolicy](#) di Referensi AWS CLI Perintah.

## update-group

Contoh kode berikut menunjukkan cara menggunakan `update-group`.

### AWS CLI

Untuk mengganti nama grup IAM

`update-group` Perintah berikut mengubah nama grup IAM `Test` menjadi `Test-1`.

```
aws iam update-group \  
  --group-name Test \  
  --new-group-name Test-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti nama grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateGroup](#) di Referensi AWS CLI Perintah.

## update-login-profile

Contoh kode berikut menunjukkan cara menggunakan `update-login-profile`.

### AWS CLI

Untuk memperbarui kata sandi untuk pengguna IAM

`update-login-profile` Perintah berikut membuat password baru untuk pengguna IAM bernama `Bob`.

```
aws iam update-login-profile \  
  --user-name Bob \  
  --password <password>
```

Perintah ini tidak menghasilkan output.

Untuk menetapkan kebijakan kata sandi untuk akun, gunakan `update-account-password-policy` perintah. Jika kata sandi baru melanggar kebijakan kata sandi akun, perintah mengembalikan `PasswordPolicyViolation` kesalahan.

Jika kebijakan kata sandi akun memungkinkan mereka, pengguna IAM dapat mengubah kata sandi mereka sendiri menggunakan `change-password` perintah.

Simpan kata sandi di tempat yang aman. Jika kata sandi hilang, itu tidak dapat dipulihkan, dan Anda harus membuat yang baru menggunakan `create-login-profile` perintah.

Untuk informasi selengkapnya, lihat [Mengelola kata sandi untuk pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateLoginProfile](#) di Referensi AWS CLI Perintah.

## **update-open-id-connect-provider-thumbprint**

Contoh kode berikut menunjukkan cara menggunakan `update-open-id-connect-provider-thumbprint`.

### AWS CLI

Untuk mengganti daftar sidik jari sertifikat server yang ada dengan daftar baru

Contoh ini memperbarui daftar cap jempol sertifikat untuk penyedia OIDC yang ARN-nya menggunakan sidik jari baru. `arn:aws:iam::123456789012:oidc-provider/example.oidcprovider.com`

```
aws iam update-open-id-connect-provider-thumbprint \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
example.oidcprovider.com \  
  --thumbprint-list 7359755EXAMPLEabc3060bce3EXAMPLEec4542a3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas OpenID Connect \(OIDC\)](#) di Panduan Pengguna IAM.AWS

- Untuk detail API, lihat [UpdateOpenIdConnectProviderThumbprint](#) di Referensi AWS CLI Perintah.

## update-role-description

Contoh kode berikut menunjukkan cara menggunakan `update-role-description`.

### AWS CLI

Untuk mengubah deskripsi peran IAM

`update-role` Perintah berikut mengubah deskripsi peran IAM `production-role` menjadi `Main production role`.

```
aws iam update-role-description \  
  --role-name production-role \  
  --description 'Main production role'
```

Output:

```
{  
  "Role": {  
    "Path": "/",  
    "RoleName": "production-role",  
    "RoleId": "AROAI234567890EXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:role/production-role",  
    "CreateDate": "2017-12-06T17:16:37+00:00",  
    "AssumeRolePolicyDocument": {  
      "Version": "2012-10-17",  
      "Statement": [  
        {  
          "Effect": "Allow",  
          "Principal": {  
            "AWS": "arn:aws:iam::123456789012:root"  
          },  
          "Action": "sts:AssumeRole",  
          "Condition": {}  
        }  
      ]  
    }  
  }  
}
```

```
    },  
    "Description": "Main production role"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi peran](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateRoleDescription](#) di Referensi AWS CLI Perintah.

## update-role

Contoh kode berikut menunjukkan cara menggunakan `update-role`.

### AWS CLI

Untuk mengubah deskripsi peran IAM atau durasi sesi

`update-role` Perintah berikut mengubah deskripsi peran IAM `production-role` menjadi `Main production role` dan menetapkan durasi sesi maksimum menjadi 12 jam.

```
aws iam update-role \  
  --role-name production-role \  
  --description 'Main production role' \  
  --max-session-duration 43200
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memodifikasi peran](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateRole](#) di Referensi AWS CLI Perintah.

## update-saml-provider

Contoh kode berikut menunjukkan cara menggunakan `update-saml-provider`.

### AWS CLI

Untuk memperbarui dokumen metadata untuk penyedia SAFL yang ada

Contoh ini memperbarui penyedia SAFL di IAM yang ARN-nya

`arn:aws:iam::123456789012:saml-provider/SAMLADFS` dengan dokumen metadata SAMB baru dari file. `SAMLMetaData.xml`

```
aws iam update-saml-provider \  
  --saml-metadata-document file://SAMLMetaData.xml \  
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFS
```

Output:

```
{  
  "SAMLProviderArn": "arn:aws:iam::123456789012:saml-provider/SAMLADFS"  
}
```

Untuk informasi selengkapnya, lihat [Membuat penyedia identitas IAM SALL](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateSamlProvider](#) di Referensi AWS CLI Perintah.

## update-server-certificate

Contoh kode berikut menunjukkan cara menggunakan `update-server-certificate`.

### AWS CLI

Untuk mengubah jalur atau nama sertifikat server di AWS akun Anda

`update-server-certificate` Perintah berikut mengubah nama sertifikat dari `myServerCertificate` menjadi `myUpdatedServerCertificate`. Itu juga mengubah jalur ke `/cloudfront/` sehingga dapat diakses oleh CloudFront layanan Amazon. Perintah ini tidak menghasilkan output. Anda dapat melihat hasil pembaruan dengan menjalankan `list-server-certificates` perintah.

```
aws-iam update-server-certificate \  
  --server-certificate-name myServerCertificate \  
  --new-server-certificate-name myUpdatedServerCertificate \  
  --new-path /cloudfront/
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola sertifikat server di IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateServerCertificate](#) di Referensi AWS CLI Perintah.

## update-service-specific-credential

Contoh kode berikut menunjukkan cara menggunakan `update-service-specific-credential`.

### AWS CLI

Contoh 1: Untuk memperbarui status kredensi khusus layanan pengguna yang meminta

`update-service-specific-credential` Contoh berikut mengubah status kredensi yang ditentukan untuk pengguna yang membuat Inactive permintaan.

```
aws iam update-service-specific-credential \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE \  
  --status Inactive
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk memperbarui status kredensi khusus layanan pengguna tertentu

`update-service-specific-credential` Contoh berikut mengubah status kredensi pengguna yang ditentukan menjadi Tidak Aktif.

```
aws iam update-service-specific-credential \  
  --user-name sofia \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE \  
  --status Inactive
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Kredensial Git untuk Koneksi HTTPS CodeCommit](#) di Panduan Pengguna AWS CodeCommit

- Untuk detail API, lihat [UpdateServiceSpecificCredential](#) di Referensi AWS CLI Perintah.

## update-signing-certificate

Contoh kode berikut menunjukkan cara menggunakan `update-signing-certificate`.

### AWS CLI

Untuk mengaktifkan atau menonaktifkan sertifikat penandatanganan untuk pengguna IAM

`update-signing-certificate` Perintah berikut menonaktifkan sertifikat penandatanganan yang ditentukan untuk pengguna IAM bernama. Bob

```
aws iam update-signing-certificate \  
  --certificate-id TA7SMP42TDN5Z260BPJE7EXAMPLE \  
  --status Inactive \  
  --user-name Bob
```

Untuk mendapatkan ID untuk sertifikat penandatanganan, gunakan `list-signing-certificates` perintah.

Untuk informasi selengkapnya, lihat [Mengelola sertifikat penandatanganan](#) di Panduan Pengguna Amazon EC2.

- Untuk detail API, lihat [UpdateSigningCertificate](#) di Referensi AWS CLI Perintah.

## update-ssh-public-key

Contoh kode berikut menunjukkan cara menggunakan `update-ssh-public-key`.

### AWS CLI

Untuk mengubah status kunci publik SSH

`update-ssh-public-key` Perintah berikut mengubah status kunci publik yang ditentukan menjadi `Inactive`.

```
aws iam update-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA1234567890EXAMPLE \  
  --status Inactive
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan kunci SSH dan SSH dengan CodeCommit](#) dalam Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateSshPublicKey](#) di Referensi AWS CLI Perintah.

## update-user

Contoh kode berikut menunjukkan cara menggunakan `update-user`.



## AWS CLI

Untuk mengubah nama pengguna IAM

`update-user` Perintah berikut mengubah nama pengguna IAM Bob menjadi Robert.

```
aws iam update-user \  
  --user-name Bob \  
  --new-user-name Robert
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengganti nama grup pengguna IAM](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateUser](#) di Referensi AWS CLI Perintah.

## upload-server-certificate

Contoh kode berikut menunjukkan cara menggunakan `upload-server-certificate`.

### AWS CLI

Untuk mengunggah sertifikat server ke AWS akun Anda

`upload-server-certificate` Perintah berikut mengunggah sertifikat server ke AWS akun Anda. Dalam contoh ini, sertifikat ada di file `public_key_cert_file.pem`, kunci pribadi terkait ada di file `my_private_key.pem`, dan rantai sertifikat yang disediakan oleh otoritas sertifikat (CA) ada di `my_certificate_chain_file.pem` file. Ketika file telah selesai diunggah, itu tersedia di bawah nama `myServerCertificate`. Parameter yang dimulai dengan `file://` memberi tahu perintah untuk membaca isi file dan menggunakannya sebagai nilai parameter alih-alih nama file itu sendiri.

```
aws iam upload-server-certificate \  
  --server-certificate-name myServerCertificate \  
  --certificate-body file://public_key_cert_file.pem \  
  --private-key file://my_private_key.pem \  
  --certificate-chain file://my_certificate_chain_file.pem
```

Output:

```
{
  "ServerCertificateMetadata": {
    "Path": "/",
    "ServerCertificateName": "myServerCertificate",
    "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
    "Arn": "arn:aws:iam::1234567989012:server-certificate/myServerCertificate",
    "UploadDate": "2019-04-22T21:13:44+00:00",
    "Expiration": "2019-10-15T22:23:16+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat Membuat, Mengunggah, dan Menghapus Sertifikat Server dalam panduan Menggunakan IAM.

- Untuk detail API, lihat [UploadServerCertificate](#) di Referensi AWS CLI Perintah.

## upload-signing-certificate

Contoh kode berikut menunjukkan cara menggunakan `upload-signing-certificate`.

### AWS CLI

Untuk mengunggah sertifikat penandatanganan untuk pengguna IAM

`upload-signing-certificate` Perintah berikut mengunggah sertifikat penandatanganan untuk pengguna IAM bernama Bob

```
aws iam upload-signing-certificate \
  --user-name Bob \
  --certificate-body file://certificate.pem
```

### Output:

```
{
  "Certificate": {
    "UserName": "Bob",
    "Status": "Active",
    "CertificateBody": "-----BEGIN CERTIFICATE-----<certificate-body>-----END
CERTIFICATE-----",
    "CertificateId": "TA7SMP42TDN5Z260BPJE7EXAMPLE",
    "UploadDate": "2013-06-06T21:40:08.121Z"
  }
}
```

```
}
```

Sertifikat dalam file bernama `certificate.pem` dalam format PEM.

Untuk informasi selengkapnya, lihat [Membuat dan Mengunggah Sertifikat Penandatanganan Pengguna](#) di panduan [Menggunakan IAM](#).

- Untuk detail API, lihat [UploadSigningCertificate](#) di Referensi AWS CLI Perintah.

## upload-ssh-public-key

Contoh kode berikut menunjukkan cara menggunakan `upload-ssh-public-key`.

### AWS CLI

Untuk mengunggah kunci publik SSH dan mengaitkannya dengan pengguna

`upload-ssh-public-key` Perintah berikut mengunggah kunci publik yang ditemukan dalam file `sshkey.pub` dan melampirkannya ke pengguna `sofia`

```
aws iam upload-ssh-public-key \
  --user-name sofia \
  --ssh-public-key-body file://sshkey.pub
```

### Output:

```
{
  "SSHPublicKey": {
    "UserName": "sofia",
    "SSHPublicKeyId": "APKA1234567890EXAMPLE",
    "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
    "SSHPublicKeyBody": "ssh-rsa <<long string generated by ssh-keygen
command>>",
    "Status": "Active",
    "UploadDate": "2019-04-18T17:04:49+00:00"
  }
}
```

Untuk informasi selengkapnya tentang cara membuat kunci dalam format yang sesuai untuk perintah ini, lihat [SSH dan Linux, macOS, atau Unix: Mengatur kunci publik dan pribadi untuk Git](#) dan [atau SSH CodeCommit dan Windows: Mengatur kunci publik dan pribadi untuk Git CodeCommit dan di](#) Panduan Pengguna.AWS CodeCommit

- Untuk detail API, lihat [UploadSshPublicKey](#) di Referensi AWS CLI Perintah.

## Contoh IAM Access Analyzer menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface dengan IAM Access Analyzer.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **apply-archive-rule**

Contoh kode berikut menunjukkan cara menggunakan `apply-archive-rule`.

### AWS CLI

Untuk menerapkan aturan arsip pada temuan yang ada yang memenuhi kriteria aturan arsip

`apply-archive-rule` Contoh berikut menerapkan aturan arsip untuk temuan yang ada yang memenuhi kriteria aturan arsip.

```
aws accessanalyzer apply-archive-rule \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyArchiveRule
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Aturan arsip](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ApplyArchiveRule](#) di Referensi AWS CLI Perintah.

## cancel-policy-generation

Contoh kode berikut menunjukkan cara menggunakan `cancel-policy-generation`.

### AWS CLI

Untuk membatalkan pembuatan kebijakan yang diminta

`cancel-policy-generation` Contoh berikut membatalkan id pekerjaan pembuatan kebijakan yang diminta.

```
aws accessanalyzer cancel-policy-generation \  
  --job-id 923a56b0-ebb8-4e80-8a3c-a11ccfbcd6f2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [pembuatan kebijakan IAM Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CancelPolicyGeneration](#) di Referensi AWS CLI Perintah.

## check-access-not-granted

Contoh kode berikut menunjukkan cara menggunakan `check-access-not-granted`.

### AWS CLI

Untuk memeriksa apakah akses yang ditentukan tidak diizinkan oleh kebijakan

`check-access-not-granted` Contoh berikut memeriksa apakah akses yang ditentukan tidak diizinkan oleh kebijakan.

```
aws accessanalyzer check-access-not-granted \  
  --policy-document file://myfile.json \  
  --access actions="s3:DeleteBucket","s3:GetBucketLocation" \  
  --policy-type IDENTITY_POLICY
```

Isi dari `myfile.json`:

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Resource": [
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    ]
  }
]
```

Output:

```
{
  "result": "PASS",
  "message": "The policy document does not grant access to perform the listed
actions."
}
```

Untuk informasi selengkapnya, lihat [Mempratinjau akses dengan API Penganalisis Akses IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [CheckAccessNotGranted](#) di Referensi AWS CLI Perintah.

## check-no-new-access

Contoh kode berikut menunjukkan cara menggunakan `check-no-new-access`.

### AWS CLI

Untuk memeriksa apakah akses baru diizinkan untuk kebijakan yang diperbarui jika dibandingkan dengan kebijakan yang ada

`check-no-new-access` Contoh berikut memeriksa apakah akses baru diizinkan untuk kebijakan yang diperbarui jika dibandingkan dengan kebijakan yang ada.

```
aws accessanalyzer check-no-new-access \
```

```
--existing-policy-document file://existing-policy.json \  
--new-policy-document file://new-policy.json \  
--policy-type IDENTITY_POLICY
```

Isi dari existing-policy.json:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetObject",  
        "s3:ListBucket"  
      ],  
      "Resource": [  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"  
      ]  
    }  
  ]  
}
```

Isi dari new-policy.json:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetObject",  
        "s3:GetObjectAcl",  
        "s3:ListBucket"  
      ],  
      "Resource": [  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"  
      ]  
    }  
  ]  
}
```

**Output:**

```
{
  "result": "FAIL",
  "message": "The modified permissions grant new access compared to your existing
policy.",
  "reasons": [
    {
      "description": "New access in the statement with index: 0.",
      "statementIndex": 0
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mempratinjau akses dengan API Penganalisis Akses IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [CheckNoNewAccess](#) di Referensi AWS CLI Perintah.

**create-access-preview**

Contoh kode berikut menunjukkan cara menggunakan `create-access-preview`.

**AWS CLI**

Untuk membuat pratinjau akses yang memungkinkan Anda melihat pratinjau temuan IAM Access Analyzer untuk sumber daya Anda sebelum menerapkan izin sumber daya

`create-access-preview` Contoh berikut membuat pratinjau akses yang memungkinkan Anda melihat pratinjau temuan IAM Access Analyzer untuk sumber daya Anda sebelum menerapkan izin sumber daya di akun Anda. AWS

```
aws accessanalyzer create-access-preview \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
  --configurations file://myfile.json
```

Isi dari `myfile.json`:

```
{
  "arn:aws:s3:::DOC-EXAMPLE-BUCKET": {
```



```

    "s3Bucket": {
      "bucketPolicy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"
      \"Allow\",\"Principal\":{\"AWS\":[\"arn:aws:iam::111122223333:root\"]},\"Action\":"
      \"s3:PutObject\",\"s3:PutObjectAcl\"],\"Resource\":"arn:aws:s3::DOC-EXAMPLE-
      BUCKET/*\"}]}",
      "bucketPublicAccessBlock": {
        "ignorePublicAcls": true,
        "restrictPublicBuckets": true
      },
      "bucketAclGrants": [
        {
          "grantee": {
            "id":
            "79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"
          },
          "permission": "READ"
        }
      ]
    }
  }
}

```

Output:

```

{
  "id": "3c65eb13-6ef9-4629-8919-a32043619e6b"
}

```

Untuk informasi selengkapnya, lihat [Mempratinjau akses dengan API Penganalisis Akses IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [CreateAccessPreview](#) di Referensi AWS CLI Perintah.

## create-analyzer

Contoh kode berikut menunjukkan cara menggunakan `create-analyzer`.

### AWS CLI

Untuk membuat analyzer

`create-analyzer` Contoh berikut membuat analyzer di AWS akun Anda.

```
aws accessanalyzer create-analyzer \  
  --analyzer-name example \  
  --type ACCOUNT
```

Output:

```
{  
  "arn": "arn:aws:access-analyzer:us-east-2:111122223333:analyzer/example"  
}
```

Untuk informasi selengkapnya, lihat [Memulai temuan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateAnalyzer](#) di Referensi AWS CLI Perintah.

## create-archive-rule

Contoh kode berikut menunjukkan cara menggunakan `create-archive-rule`.

### AWS CLI

Untuk membuat aturan arsip untuk analyzer yang ditentukan

`create-archive-rule` Contoh berikut membuat aturan arsip untuk analyzer yang ditentukan di AWS akun Anda.

```
aws accessanalyzer create-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyRule \  
  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":  
  ["AWS::IAM::Role"]}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Aturan arsip](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [CreateArchiveRule](#) di Referensi AWS CLI Perintah.

## delete-analyzer

Contoh kode berikut menunjukkan cara menggunakan `delete-analyzer`.

## AWS CLI

Untuk menghapus analyzer yang ditentukan

`delete-analyzer` Contoh berikut menghapus analyzer yang ditentukan di akun Anda AWS .

```
aws accessanalyzer delete-analyzer \  
  --analyzer-name example
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Aturan arsip](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteAnalyzer](#) di Referensi AWS CLI Perintah.

## `delete-archive-rule`

Contoh kode berikut menunjukkan cara menggunakan `delete-archive-rule`.

## AWS CLI

Untuk menghapus aturan arsip yang ditentukan

`delete-archive-rule` Contoh berikut menghapus aturan arsip yang ditentukan di AWS akun Anda.

```
aws accessanalyzer delete-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyRule
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Aturan arsip](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DeleteArchiveRule](#) di Referensi AWS CLI Perintah.

## `get-access-preview`

Contoh kode berikut menunjukkan cara menggunakan `get-access-preview`.

## AWS CLI

Untuk mengambil informasi tentang pratinjau akses untuk penganalisis yang ditentukan

`get-access-preview` Contoh berikut mengambil informasi tentang pratinjau akses untuk penganalisis yang ditentukan di akun Anda AWS .

```
aws accessanalyzer get-access-preview \
  --access-preview-id 3c65eb13-6ef9-4629-8919-a32043619e6b \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account
```

Output:

```
{
  "accessPreview": {
    "id": "3c65eb13-6ef9-4629-8919-a32043619e6b",
    "analyzerArn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
    "configurations": {
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET": {
        "s3Bucket": {
          "bucketPolicy": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\n\"Effect\":\n\"Allow\", \"Principal\":{\n\"AWS\":[\n\"arn:aws:iam::111122223333:root\"
]}\n\"Action\":[\n\"s3:PutObject\", \"s3:PutObjectAcl\"], \"Resource\":\n\"arn:aws:s3:::DOC-
EXAMPLE-BUCKET/*\"}]}",
          "bucketAclGrants": [
            {
              "permission": "READ",
              "grantee": {
                "id":
"79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"
              }
            }
          ],
          "bucketPublicAccessBlock": {
            "ignorePublicAcls": true,
            "restrictPublicBuckets": true
          }
        }
      }
    },
    "createdAt": "2024-02-17T00:18:44+00:00",
    "status": "COMPLETED"
  }
}
```

Untuk informasi selengkapnya, lihat [Mempratinjau akses dengan API Penganalisis Akses IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [GetAccessPreview](#) di Referensi AWS CLI Perintah.

## get-analyzed-resource

Contoh kode berikut menunjukkan cara menggunakan `get-analyzed-resource`.

### AWS CLI

Untuk mengambil informasi tentang sumber daya yang dianalisis

`get-analyzed-resource` Contoh berikut mengambil informasi tentang sumber daya yang dianalisis di AWS akun Anda.

```
aws accessanalyzer get-analyzed-resource \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
  ConsoleAnalyzer-account \
  --resource-arn arn:aws:s3:::DOC-EXAMPLE-BUCKET
```

Output:

```
{
  "resource": {
    "analyzedAt": "2024-02-15T18:01:53.002000+00:00",
    "isPublic": false,
    "resourceArn": "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::S3::Bucket"
  }
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetAnalyzedResource](#) di Referensi AWS CLI Perintah.

## get-analyzer

Contoh kode berikut menunjukkan cara menggunakan `get-analyzer`.

## AWS CLI

Untuk mengambil informasi tentang analyzer yang ditentukan

`get-analyzer` Contoh berikut mengambil informasi tentang analyzer yang ditentukan di akun Anda AWS .

```
aws accessanalyzer get-analyzer \  
  --analyzer-name ConsoleAnalyzer-account
```

Output:

```
{  
  "analyzer": {  
    "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account",  
    "createdAt": "2019-12-03T07:28:17+00:00",  
    "lastResourceAnalyzed": "arn:aws:sns:us-west-2:111122223333:config-topic",  
    "lastResourceAnalyzedAt": "2024-02-15T18:01:53.003000+00:00",  
    "name": "ConsoleAnalyzer-account",  
    "status": "ACTIVE",  
    "tags": {  
      "auto-delete": "no"  
    },  
    "type": "ACCOUNT"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetAnalyzer](#) di Referensi AWS CLI Perintah.

## `get-archive-rule`

Contoh kode berikut menunjukkan cara menggunakan `get-archive-rule`.

## AWS CLI

Untuk mengambil informasi tentang aturan arsip

`get-archive-rule` Contoh berikut mengambil informasi tentang aturan arsip di AWS akun Anda.

```
aws accessanalyzer get-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyArchiveRule
```

Output:

```
{  
  "archiveRule": {  
    "createdAt": "2024-02-15T00:49:27+00:00",  
    "filter": {  
      "resource": {  
        "contains": [  
          "Cognito"  
        ]  
      },  
      "resourceType": {  
        "eq": [  
          "AWS::IAM::Role"  
        ]  
      }  
    },  
    "ruleName": "MyArchiveRule",  
    "updatedAt": "2024-02-15T00:49:27+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Aturan arsip](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetArchiveRule](#) di Referensi AWS CLI Perintah.

## get-finding-v2

Contoh kode berikut menunjukkan cara menggunakan `get-finding-v2`.

### AWS CLI

Untuk mengambil informasi tentang temuan yang ditentukan

`get-finding-v2` Contoh berikut etrieves informasi tentang temuan yang ditentukan di akun Anda AWS .

```
aws accessanalyzer get-finding-v2 \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyArchiveRule
```

```
--analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization \
--id 0910eedb-381e-4e95-adda-0d25c19e6e90
```

## Output:

```
{
  "findingDetails": [
    {
      "externalAccessDetails": {
        "action": [
          "sts:AssumeRoleWithWebIdentity"
        ],
        "condition": {
          "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
        },
        "isPublic": false,
        "principal": {
          "Federated": "cognito-identity.amazonaws.com"
        }
      }
    }
  ],
  "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
  "status": "ACTIVE",
  "error": null,
  "createdAt": "2021-02-26T21:17:50.905000+00:00",
  "resourceType": "AWS::IAM::Role",
  "findingType": "ExternalAccess",
  "resourceOwnerAccount": "111122223333",
  "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
  "id": "0910eedb-381e-4e95-adda-0d25c19e6e90",
  "updatedAt": "2021-02-26T21:17:50.905000+00:00"
}
```

Untuk informasi selengkapnya, lihat [Meninjau temuan](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetFindingV2](#) di Referensi AWS CLI Perintah.

## get-finding

Contoh kode berikut menunjukkan cara menggunakan `get-finding`.



## AWS CLI

Untuk mengambil informasi tentang temuan yang ditentukan

`get-finding` Contoh berikut retrieves informasi tentang temuan yang ditentukan di akun Anda AWS .

```
aws accessanalyzer get-finding \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-organization \  
  --id 0910eedb-381e-4e95-adda-0d25c19e6e90
```

Output:

```
{  
  "finding": {  
    "id": "0910eedb-381e-4e95-adda-0d25c19e6e90",  
    "principal": {  
      "Federated": "cognito-identity.amazonaws.com"  
    },  
    "action": [  
      "sts:AssumeRoleWithWebIdentity"  
    ],  
    "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",  
    "isPublic": false,  
    "resourceType": "AWS::IAM::Role",  
    "condition": {  
      "cognito-identity.amazonaws.com:aud": "us-  
west-2:EXAMPLE0-0000-0000-0000-000000000000"  
    },  
    "createdAt": "2021-02-26T21:17:50.905000+00:00",  
    "analyzedAt": "2024-02-16T18:17:47.888000+00:00",  
    "updatedAt": "2021-02-26T21:17:50.905000+00:00",  
    "status": "ACTIVE",  
    "resourceOwnerAccount": "111122223333"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Meninjau temuan](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetFinding](#) di Referensi AWS CLI Perintah.

## get-generated-policy

Contoh kode berikut menunjukkan cara menggunakan `get-generated-policy`.

### AWS CLI

Untuk mengambil kebijakan yang dibuat menggunakan `StartPolicyGeneration` API

`get-generated-policy` Contoh berikut mengambil kebijakan yang dibuat menggunakan `StartPolicyGeneration` API di AWS akun Anda.

```
aws accessanalyzer get-generated-policy \
  --job-id c557dc4a-0338-4489-95dd-739014860ff9
```

### Output:

```
{
  "generatedPolicyResult": {
    "generatedPolicies": [
      {
        "policy": "{\n\"Version\": \"2012-10-17\", \"Statement\":\n[\n{\n\"Sid\": \"SupportedServiceSid0\", \"Effect\": \"Allow\", \"Action\":\n[\n\"access-analyzer:GetAnalyzer\", \"access-analyzer:ListAnalyzers\",\n\"access-analyzer:ListArchiveRules\", \"access-analyzer:ListFindings\n\", \"cloudtrail:DescribeTrails\", \"cloudtrail:GetEventDataStore\",\n\"cloudtrail:GetEventSelectors\", \"cloudtrail:GetInsightSelectors\n\", \"cloudtrail:GetTrailStatus\", \"cloudtrail:ListChannels\",\n\"cloudtrail:ListEventDataStores\", \"cloudtrail:ListQueries\", \"cloudtrail:ListTags\n\", \"cloudtrail:LookupEvents\", \"ec2:DescribeRegions\", \"iam:GetAccountSummary\n\", \"iam:GetOpenIDConnectProvider\", \"iam:GetRole\", \"iam:ListAccessKeys\",\n\"iam:ListAccountAliases\", \"iam:ListOpenIDConnectProviders\", \"iam:ListRoles\n\", \"iam:ListSAMLProviders\", \"kms:ListAliases\", \"s3:GetBucketLocation\",\n\"s3:ListAllMyBuckets\"]\n\", \"Resource\": \"*\"]}"
      }
    ],
    "properties": {
      "cloudTrailProperties": {
        "endTime": "2024-02-14T22:44:40+00:00",
        "startTime": "2024-02-13T00:30:00+00:00",
        "trailProperties": [
          {
            "allRegions": true,

```

```

        "cloudTrailArn": "arn:aws:cloudtrail:us-
west-2:111122223333:trail/my-trail",
        "regions": []
    }
]
},
"principalArn": "arn:aws:iam::111122223333:role/Admin"
}
},
"jobDetails": {
    "completedOn": "2024-02-14T22:47:01+00:00",
    "jobId": "c557dc4a-0338-4489-95dd-739014860ff9",
    "startedOn": "2024-02-14T22:44:41+00:00",
    "status": "SUCCEEDED"
}
}
}

```

Untuk informasi selengkapnya, lihat [pembuatan kebijakan IAM Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [GetGeneratedPolicy](#) di Referensi AWS CLI Perintah.

## list-access-preview-findings

Contoh kode berikut menunjukkan cara menggunakan `list-access-preview-findings`.

### AWS CLI

Untuk mengambil daftar temuan pratinjau akses yang dihasilkan oleh pratinjau akses yang ditentukan

`list-access-preview-findings` Contoh berikut mengambil daftar temuan pratinjau akses yang dihasilkan oleh pratinjau akses yang ditentukan di AWS akun Anda.

```

aws accessanalyzer list-access-preview-findings \
  --access-preview-id 3c65eb13-6ef9-4629-8919-a32043619e6b \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account

```

### Output:

```
{
```

```

"findings": [
  {
    "id": "e22fc158-1c87-4c32-9464-e7f405ce8d74",
    "principal": {
      "AWS": "111122223333"
    },
    "action": [
      "s3:PutObject",
      "s3:PutObjectAcl"
    ],
    "condition": {},
    "resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
    "isPublic": false,
    "resourceType": "AWS::S3::Bucket",
    "createdAt": "2024-02-17T00:18:46+00:00",
    "changeType": "NEW",
    "status": "ACTIVE",
    "resourceOwnerAccount": "111122223333",
    "sources": [
      {
        "type": "POLICY"
      }
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mempratinjau akses dengan API Penganalisis Akses IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [ListAccessPreviewFindings](#) di Referensi AWS CLI Perintah.

## list-access-previews

Contoh kode berikut menunjukkan cara menggunakan `list-access-previews`.

### AWS CLI

Untuk mengambil daftar pratinjau akses untuk analyzer yang ditentukan

`list-access-previews` Contoh berikut mengambil daftar pratinjau akses untuk analyzer yang ditentukan di akun Anda. AWS

```
aws accessanalyzer list-access-previews \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
  ConsoleAnalyzer-account
```

Output:

```
{
  "accessPreviews": [
    {
      "id": "3c65eb13-6ef9-4629-8919-a32043619e6b",
      "analyzerArn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
      ConsoleAnalyzer-account",
      "createdAt": "2024-02-17T00:18:44+00:00",
      "status": "COMPLETED"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mempratinjau akses dengan API Penganalisis Akses IAM di Panduan Pengguna AWS IAM](#).

- Untuk detail API, lihat [ListAccessPreviews](#) di Referensi AWS CLI Perintah.

## list-analyzed-resources

Contoh kode berikut menunjukkan cara menggunakan `list-analyzed-resources`.

AWS CLI

Untuk daftar widget yang tersedia

`list-analyzed-resources` Contoh berikut mencantumkan widget yang tersedia di AWS akun Anda.

```
aws accessanalyzer list-analyzed-resources \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
  ConsoleAnalyzer-account \
  --resource-type AWS::IAM::Role
```

Output:

```
{
```

```
"analyzedResources": [
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:Validation-Email",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  },
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:admin-alerts",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  },
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:config-topic",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  },
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:inspector-topic",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  }
]
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListAnalyzedResources](#) di Referensi AWS CLI Perintah.

## list-analyzers

Contoh kode berikut menunjukkan cara menggunakan `list-analyzers`.

### AWS CLI

Untuk mengambil daftar analisa

`list-analyzers` Contoh berikut mengambil daftar analisa di akun Anda AWS .

```
aws accessanalyzer list-analyzers
```

Output:

```
{
  "analyzers": [
    {
      "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
UnusedAccess-ConsoleAnalyzer-organization",
      "createdAt": "2024-02-15T00:46:40+00:00",
      "name": "UnusedAccess-ConsoleAnalyzer-organization",
      "status": "ACTIVE",
      "tags": {
        "auto-delete": "no"
      },
      "type": "ORGANIZATION_UNUSED_ACCESS"
    },
    {
      "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization",
      "createdAt": "2020-04-25T07:43:28+00:00",
      "lastResourceAnalyzed": "arn:aws:s3::DOC-EXAMPLE-BUCKET",
      "lastResourceAnalyzedAt": "2024-02-15T21:51:56.517000+00:00",
      "name": "ConsoleAnalyzer-organization",
      "status": "ACTIVE",
      "tags": {
        "auto-delete": "no"
      },
      "type": "ORGANIZATION"
    },
    {
      "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
      "createdAt": "2019-12-03T07:28:17+00:00",
      "lastResourceAnalyzed": "arn:aws:sns:us-west-2:111122223333:config-
topic",
      "lastResourceAnalyzedAt": "2024-02-15T18:01:53.003000+00:00",
      "name": "ConsoleAnalyzer-account",
      "status": "ACTIVE",
      "tags": {
        "auto-delete": "no"
      },
      "type": "ACCOUNT"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListAnalyzers](#) di Referensi AWS CLI Perintah.

## list-archive-rules

Contoh kode berikut menunjukkan cara menggunakan `list-archive-rules`.

### AWS CLI

Untuk mengambil daftar aturan arsip yang dibuat untuk analyzer yang ditentukan

`list-archive-rules` Contoh berikut mengambil daftar aturan arsip yang dibuat untuk analyzer yang ditentukan di akun Anda AWS .

```
aws accessanalyzer list-archive-rules \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization
```

### Output:

```
{  
  "archiveRules": [  
    {  
      "createdAt": "2024-02-15T00:49:27+00:00",  
      "filter": {  
        "resource": {  
          "contains": [  
            "Cognito"  
          ]  
        },  
        "resourceType": {  
          "eq": [  
            "AWS::IAM::Role"  
          ]  
        }  
      },  
      "ruleName": "MyArchiveRule",  
      "updatedAt": "2024-02-15T00:49:27+00:00"  
    },  
    {  
      "createdAt": "2024-02-15T23:27:45+00:00",  
      "filter": {
```



```

        "findingType": {
            "eq": [
                "UnusedIAMUserAccessKey"
            ]
        },
        "ruleName": "ArchiveRule-56125a39-e517-4ff8-afb1-ef06f58db612",
        "updatedAt": "2024-02-15T23:27:45+00:00"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListArchiveRules](#) di Referensi AWS CLI Perintah.

## list-findings-v2

Contoh kode berikut menunjukkan cara menggunakan `list-findings-v2`.

### AWS CLI

Untuk mengambil daftar temuan yang dihasilkan oleh penganalisis yang ditentukan

`list-findings-v2` Contoh berikut mengambil daftar temuan yang dihasilkan oleh penganalisis yang ditentukan di akun Anda AWS . Contoh ini memfilter hasil untuk menyertakan hanya peran IAM yang namanya berisi `Cognito`.

```

aws accessanalyzer list-findings-v2 \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
  ConsoleAnalyzer-account \
  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":
  ["AWS::IAM::Role"]}}'

```

### Output:

```

{
  "findings": [
    {
      "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
      "createdAt": "2021-02-26T21:17:24.710000+00:00",

```

```

    "id": "597f3bc2-3adc-4c18-9879-5c4b23485e46",
    "resource": "arn:aws:iam::111122223333:role/
Cognito_testpoolUnauth_Role",
    "resourceType": "AWS::IAM::Role",
    "resourceOwnerAccount": "111122223333",
    "status": "ACTIVE",
    "updatedAt": "2021-02-26T21:17:24.710000+00:00",
    "findingType": "ExternalAccess"
  },
  {
    "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
    "createdAt": "2021-02-26T21:17:50.905000+00:00",
    "id": "ce0e221a-85b9-4d52-91ff-d7678075442f",
    "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
    "resourceType": "AWS::IAM::Role",
    "resourceOwnerAccount": "111122223333",
    "status": "ACTIVE",
    "updatedAt": "2021-02-26T21:17:50.905000+00:00",
    "findingType": "ExternalAccess"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListFindingsV2](#) di Referensi AWS CLI Perintah.

## list-findings

Contoh kode berikut menunjukkan cara menggunakan `list-findings`.

### AWS CLI

Untuk mengambil daftar temuan yang dihasilkan oleh penganalisis yang ditentukan

`list-findings` Contoh berikut mengambil daftar temuan yang dihasilkan oleh penganalisis yang ditentukan di akun Anda AWS . Contoh ini memfilter hasil untuk menyertakan hanya peran IAM yang namanya berisi `Cognito`.

```

aws accessanalyzer list-findings \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \

```

```
--filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq": ["AWS::IAM::Role"]}]'
```

## Output:

```
{
  "findings": [
    {
      "id": "597f3bc2-3adc-4c18-9879-5c4b23485e46",
      "principal": {
        "Federated": "cognito-identity.amazonaws.com"
      },
      "action": [
        "sts:AssumeRoleWithWebIdentity"
      ],
      "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolUnauth_Role",
      "isPublic": false,
      "resourceType": "AWS::IAM::Role",
      "condition": {
        "cognito-identity.amazonaws.com:aud": "us-west-2:EXAMPLE0-0000-0000-0000-000000000000"
      },
      "createdAt": "2021-02-26T21:17:24.710000+00:00",
      "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
      "updatedAt": "2021-02-26T21:17:24.710000+00:00",
      "status": "ACTIVE",
      "resourceOwnerAccount": "111122223333"
    },
    {
      "id": "ce0e221a-85b9-4d52-91ff-d7678075442f",
      "principal": {
        "Federated": "cognito-identity.amazonaws.com"
      },
      "action": [
        "sts:AssumeRoleWithWebIdentity"
      ],
      "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
      "isPublic": false,
      "resourceType": "AWS::IAM::Role",
      "condition": {
        "cognito-identity.amazonaws.com:aud": "us-west-2:EXAMPLE0-0000-0000-0000-000000000000"
      }
    }
  ]
}
```

```
    },
    "createdAt": "2021-02-26T21:17:50.905000+00:00",
    "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
    "updatedAt": "2021-02-26T21:17:50.905000+00:00",
    "status": "ACTIVE",
    "resourceOwnerAccount": "111122223333"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListFindings](#) di Referensi AWS CLI Perintah.

## list-policy-generations

Contoh kode berikut menunjukkan cara menggunakan `list-policy-generations`.

### AWS CLI

Untuk membuat daftar semua generasi kebijakan yang diminta dalam tujuh hari terakhir

`list-policy-generations` Contoh berikut mencantumkan semua generasi kebijakan yang diminta dalam tujuh hari terakhir di AWS akun Anda.

```
aws accessanalyzer list-policy-generations
```

Output:

```
{
  "policyGenerations": [
    {
      "completedOn": "2024-02-14T23:43:38+00:00",
      "jobId": "923a56b0-ebb8-4e80-8a3c-a11ccfbcd6f2",
      "principalArn": "arn:aws:iam::111122223333:role/Admin",
      "startedOn": "2024-02-14T23:43:02+00:00",
      "status": "CANCELED"
    },
    {
      "completedOn": "2024-02-14T22:47:01+00:00",
```

```
    "jobId": "c557dc4a-0338-4489-95dd-739014860ff9",
    "principalArn": "arn:aws:iam::111122223333:role/Admin",
    "startedOn": "2024-02-14T22:44:41+00:00",
    "status": "SUCCEEDED"
  }
]
```

Untuk informasi selengkapnya, lihat [pembuatan kebijakan IAM Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListPolicyGenerations](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mengambil daftar tag yang diterapkan ke sumber daya yang ditentukan

`list-tags-for-resource` Contoh berikut mengambil daftar tag yang diterapkan ke sumber daya tertentu di AWS akun Anda.

```
aws accessanalyzer list-tags-for-resource \
  --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
  ConsoleAnalyzer-account
```

### Output:

```
{
  "tags": {
    "Zone-of-trust": "Account",
    "Name": "ConsoleAnalyzer"
  }
}
```

Untuk informasi selengkapnya, lihat [pembuatan kebijakan IAM Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## start-policy-generation

Contoh kode berikut menunjukkan cara menggunakan `start-policy-generation`.

### AWS CLI

Untuk memulai permintaan pembuatan kebijakan

`start-policy-generation` Contoh berikut memulai permintaan pembuatan kebijakan di AWS akun Anda.

```
aws accessanalyzer start-policy-generation \  
  --policy-generation-details '{"principalArn":"arn:aws:iam::111122223333:role/  
Admin"}' \  
  --cloud-trail-details file://myfile.json
```

Isi dari `myfile.json`:

```
{  
  "accessRole": "arn:aws:iam::111122223333:role/service-role/  
AccessAnalyzerMonitorServiceRole",  
  "startTime": "2024-02-13T00:30:00Z",  
  "trails": [  
    {  
      "allRegions": true,  
      "cloudTrailArn": "arn:aws:cloudtrail:us-west-2:111122223333:trail/my-  
trail"  
    }  
  ]  
}
```

Output:

```
{  
  "jobId": "c557dc4a-0338-4489-95dd-739014860ff9"  
}
```

Untuk informasi selengkapnya, lihat [pembuatan kebijakan IAM Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [StartPolicyGeneration](#) di Referensi AWS CLI Perintah.

## start-resource-scan

Contoh kode berikut menunjukkan cara menggunakan `start-resource-scan`.

### AWS CLI

Untuk segera memulai pemindaian kebijakan yang diterapkan ke sumber daya yang ditentukan

`start-resource-scan` Contoh berikut mmediately memulai pemindaian kebijakan yang diterapkan ke sumber daya yang ditentukan di akun Anda AWS .

```
aws accessanalyzer start-resource-scan \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --resource-arn arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [pembuatan kebijakan IAM Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [StartResourceScan](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber daya yang ditentukan

`tag-resource` Contoh berikut menambahkan tag ke sumber daya yang ditentukan di AWS akun Anda.

```
aws accessanalyzer tag-resource \  
  --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --tags Environment=dev,Purpose=testing
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya yang ditentukan

`untag-resource` Contoh berikut menghapus tag dari sumber daya yang ditentukan di AWS akun Anda.

```
aws accessanalyzer untag-resource \  
  --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --tag-keys Environment Purpose
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-archive-rule

Contoh kode berikut menunjukkan cara menggunakan `update-archive-rule`.

### AWS CLI

Untuk memperbarui kriteria dan nilai untuk aturan arsip yang ditentukan

`update-archive-rule` Contoh berikut memperbarui kriteria dan nilai untuk aturan arsip yang ditentukan di AWS akun Anda.

```
aws accessanalyzer update-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyArchiveRule \  
  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":  
["AWS::IAM::Role"]}]'
```



Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Aturan arsip](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateArchiveRule](#) di Referensi AWS CLI Perintah.

## update-findings

Contoh kode berikut menunjukkan cara menggunakan `update-findings`.

### AWS CLI

Untuk memperbarui status temuan yang ditentukan

`update-findings` Contoh berikut memperbarui status untuk temuan yang ditentukan di AWS akun Anda.

```
aws accessanalyzer update-findings \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
UnusedAccess-ConsoleAnalyzer-organization \
  --ids 4f319ac3-2e0c-4dc4-bf51-7013a086b6ae 780d586a-2cce-4f72-aff6-359d450e7500
  \
  --status ARCHIVED
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS Identity and Access Management Access Analyzer](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [UpdateFindings](#) di Referensi AWS CLI Perintah.

## validate-policy

Contoh kode berikut menunjukkan cara menggunakan `validate-policy`.

### AWS CLI

Untuk meminta validasi kebijakan dan mengembalikan daftar temuan

`validate-policy` Contoh berikut meminta validasi kebijakan dan mengembalikan daftar temuan. Kebijakan dalam contoh adalah kebijakan kepercayaan peran untuk peran Amazon Cognito yang digunakan untuk federasi identitas web. Temuan yang

dihasilkan dari kebijakan kepercayaan berhubungan dengan nilai Sid elemen kosong dan prinsip kebijakan yang tidak cocok karena tindakan peran asumsi yang salah digunakan,. sts:AssumeRole Tindakan peran asumsi yang benar untuk digunakan dengan Cognito adalah. sts:AssumeRoleWithWebIdentity

```
aws accessanalyzer validate-policy \  
  --policy-document file://myfile.json \  
  --policy-type RESOURCE_POLICY
```

Isi dari myfile.json:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "",  
      "Effect": "Allow",  
      "Principal": {  
        "Federated": "cognito-identity.amazonaws.com"  
      },  
      "Action": [  
        "sts:AssumeRole",  
        "sts:TagSession"  
      ],  
      "Condition": {  
        "StringEquals": {  
          "cognito-identity.amazonaws.com:aud": "us-west-2_EXAMPLE"  
        }  
      }  
    }  
  ]  
}
```

Output:

```
{  
  "findings": [  
    {  
      "findingDetails": "Add a value to the empty string in the Sid element.",  
      "findingType": "SUGGESTION",  
      "issueCode": "EMPTY_SID_VALUE",  
    }  
  ]  
}
```

```

    "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-suggestion-empty-sid-value",
    "locations": [
        {
            "path": [
                {
                    "value": "Statement"
                },
                {
                    "index": 0
                },
                {
                    "value": "Sid"
                }
            ],
            "span": {
                "end": {
                    "column": 21,
                    "line": 5,
                    "offset": 81
                },
                "start": {
                    "column": 19,
                    "line": 5,
                    "offset": 79
                }
            }
        }
    ],
},
{
    "findingDetails": "The sts:AssumeRole action is invalid with the
following principal(s): cognito-identity.amazonaws.com. Use a SAML provider
principal with the sts:AssumeRoleWithSAML action or use an OIDC provider principal
with the sts:AssumeRoleWithWebIdentity action. Ensure the provider is Federated if
you use either of the two options.",
    "findingType": "ERROR",
    "issueCode": "MISMATCHED_ACTION_FOR_PRINCIPAL",
    "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-error-mismatched-action-for-principal",
    "locations": [
        {

```

```
    "path": [  
      {  
        "value": "Statement"  
      },  
      {  
        "index": 0  
      },  
      {  
        "value": "Action"  
      },  
      {  
        "index": 0  
      }  
    ],  
    "span": {  
      "end": {  
        "column": 32,  
        "line": 11,  
        "offset": 274  
      },  
      "start": {  
        "column": 16,  
        "line": 11,  
        "offset": 258  
      }  
    }  
  },  
  {  
    "path": [  
      {  
        "value": "Statement"  
      },  
      {  
        "index": 0  
      },  
      {  
        "value": "Principal"  
      },  
      {  
        "value": "Federated"  
      }  
    ],  
    "span": {  
      "end": {
```

```

        "column": 61,
        "line": 8,
        "offset": 202
      },
      "start": {
        "column": 29,
        "line": 8,
        "offset": 170
      }
    }
  ]
},
{
  "findingDetails": "The following actions: sts:TagSession are not
supported by the condition key cognito-identity.amazonaws.com:aud. The condition
will not be evaluated for these actions. We recommend that you move these actions
to a different statement without this condition key.",
  "findingType": "ERROR",
  "issueCode": "UNSUPPORTED_ACTION_FOR_CONDITION_KEY",
  "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-error-unsupported-action-for-condition-key",
  "locations": [
    {
      "path": [
        {
          "value": "Statement"
        },
        {
          "index": 0
        },
        {
          "value": "Action"
        },
        {
          "index": 1
        }
      ],
      "span": {
        "end": {
          "column": 32,
          "line": 12,
          "offset": 308
        }
      }
    }
  ]
}

```

```
    },
    "start": {
      "column": 16,
      "line": 12,
      "offset": 292
    }
  }
},
{
  "path": [
    {
      "value": "Statement"
    },
    {
      "index": 0
    },
    {
      "value": "Condition"
    },
    {
      "value": "StringEquals"
    },
    {
      "value": "cognito-identity.amazonaws.com:aud"
    }
  ],
  "span": {
    "end": {
      "column": 79,
      "line": 16,
      "offset": 464
    },
    "start": {
      "column": 58,
      "line": 16,
      "offset": 443
    }
  }
}
]
}
]
}
```

Untuk informasi selengkapnya, lihat [Memeriksa untuk memvalidasi kebijakan](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [ValidatePolicy](#) di Referensi AWS CLI Perintah.

## Contoh Image Builder menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Image Builder.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-component**

Contoh kode berikut menunjukkan cara menggunakan `create-component`.

AWS CLI

Untuk membuat komponen

`create-component` Contoh berikut membuat komponen yang menggunakan file dokumen JSON dan mereferensikan dokumen komponen dalam format YAMAL yang diunggah ke bucket Amazon S3.

```
aws imagebuilder create-component \  
  --cli-input-json file://create-component.json
```

Isi dari `create-component.json`:

```
{
  "name": "MyExampleComponent",
  "semanticVersion": "2019.12.02",
  "description": "An example component that builds, validates and tests an image",
  "changeDescription": "Initial version.",
  "platform": "Windows",
  "uri": "s3://s3-bucket-name/s3-bucket-path/component.yaml"
}
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "componentBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/examplecomponent/2019.12.02/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [CreateComponent](#) di Referensi AWS CLI Perintah.

## create-distribution-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-distribution-configuration`.

### AWS CLI

Untuk membuat konfigurasi distribusi

`create-distribution-configuration` Contoh berikut membuat konfigurasi distribusi menggunakan file JSON.

```
aws imagebuilder create-distribution-configuration \
  --cli-input-json file:/create-distribution-configuration.json
```

Isi dari `create-distribution-configuration.json`:

```
{
  "name": "MyExampleDistribution",
```



```

"description": "Copies AMI to eu-west-1",
"distributions": [
  {
    "region": "us-west-2",
    "amiDistributionConfiguration": {
      "name": "Name {{imagebuilder:buildDate}}",
      "description": "An example image name with parameter references",
      "amiTags": {
        "KeyName": "{{ssm:parameter_name}}"
      },
      "launchPermission": {
        "userIds": [
          "123456789012"
        ]
      }
    },
  },
  {
    "region": "eu-west-1",
    "amiDistributionConfiguration": {
      "name": "My {{imagebuilder:buildVersion}} image
{{imagebuilder:buildDate}}",
      "amiTags": {
        "KeyName": "Value"
      },
      "launchPermission": {
        "userIds": [
          "123456789012"
        ]
      }
    }
  }
]
}

```

### Output:

```

{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution"
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [CreateDistributionConfiguration](#) di Referensi AWS CLI Perintah.

## create-image-pipeline

Contoh kode berikut menunjukkan cara menggunakan `create-image-pipeline`.

### AWS CLI

Untuk membuat pipeline gambar

`create-image-pipeline` Contoh berikut membuat pipeline gambar menggunakan file JSON.

```
aws imagebuilder create-image-pipeline \  
  --cli-input-json file://create-image-pipeline.json
```

Isi dari `create-image-pipeline.json`:

```
{  
  "name": "MyWindows2016Pipeline",  
  "description": "Builds Windows 2016 Images",  
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
mybasicrecipe/2019.12.03",  
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",  
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:distribution-configuration/myexampledistribution",  
  "imageTestsConfiguration": {  
    "imageTestsEnabled": true,  
    "timeoutMinutes": 60  
  },  
  "schedule": {  
    "scheduleExpression": "cron(0 0 * * SUN)",  
    "pipelineExecutionStartCondition":  
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"  
  },  
  "status": "ENABLED"  
}
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/mywindows2016pipeline"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [CreateImagePipeline](#) di Referensi AWS CLI Perintah.

## create-image-recipe

Contoh kode berikut menunjukkan cara menggunakan `create-image-recipe`.

### AWS CLI

Untuk membuat resep

`create-image-recipe` Contoh berikut membuat resep gambar menggunakan file JSON. Komponen dipasang sesuai urutan yang ditentukan.

```
aws imagebuilder create-image-recipe \
  --cli-input-json file://create-image-recipe.json
```

Isi dari `create-image-recipe.json`:

```
{
  "name": "MyBasicRecipe",
  "description": "This example image recipe creates a Windows 2016 image.",
  "semanticVersion": "2019.12.03",
  "components":
  [
    {
      "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/myexamplecomponent/2019.12.02/1"
    },
    {
      "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/myimportedcomponent/1.0.0/1"
    }
  ]
}
```

```

    }
  ],
  "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-
english-full-base-x86/xxxx.x.x"
}

```

**Output:**

```

{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03"
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [CreateImageRecipe](#) di Referensi AWS CLI Perintah.

**create-image**

Contoh kode berikut menunjukkan cara menggunakan `create-image`.

**AWS CLI**

Untuk membuat gambar

`create-image` Contoh berikut membuat gambar.

```

aws imagebuilder create-image \
  --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03 \
  --infrastructure-configuration-arn arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure

```

**Output:**

```

{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",

```

```
"imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [CreateImage](#) di Referensi AWS CLI Perintah.

## create-infrastructure-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-infrastructure-configuration`.

### AWS CLI

Untuk membuat konfigurasi infrastruktur

`create-infrastructure-configuration` Contoh berikut membuat konfigurasi infrastruktur menggunakan file JSON.

```
aws imagebuilder create-infrastructure-configuration \
  --cli-input-json file://create-infrastructure-configuration.json
```

Isi dari `create-infrastructure-configuration.json`:

```
{
  "name": "MyExampleInfrastructure",
  "description": "An example that will retain instances of failed builds",
  "instanceTypes": [
    "m5.large", "m5.xlarge"
  ],
  "instanceProfileName": "EC2InstanceProfileForImageBuilder",
  "securityGroupIds": [
    "sg-a1b2c3d4"
  ],
  "subnetId": "subnet-a1b2c3d4",
  "logging": {
    "s3Logs": {
      "s3BucketName": "bucket-name",
      "s3KeyPrefix": "bucket-path"
    }
  }
}
```

```
    }
  },
  "keyPair": "key-pair-name",
  "terminateInstanceOnFailure": false,
  "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-topic-name"
}
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [CreateInfrastructureConfiguration](#) di Referensi AWS CLI Perintah.

## delete-component

Contoh kode berikut menunjukkan cara menggunakan `delete-component`.

### AWS CLI

Untuk menghapus komponen

`delete-component` Contoh berikut menghapus versi build komponen dengan menentukan ARN-nya.

```
aws imagebuilder delete-component \
  --component-build-version-arn arn:aws:imagebuilder:us-
west-2:123456789012:component/myexamplecomponent/2019.12.02/1
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```

```
"componentBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/myexamplecomponent/2019.12.02/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [DeleteComponent](#) di Referensi AWS CLI Perintah.

## delete-image-pipeline

Contoh kode berikut menunjukkan cara menggunakan `delete-image-pipeline`.

### AWS CLI

Untuk menghapus pipeline gambar

`delete-image-pipeline` Contoh berikut menghapus pipeline gambar dengan menentukan ARN nya.

```
aws imagebuilder delete-image-pipeline \
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
  my-example-pipeline
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
  mywindows2016pipeline"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [DeleteImagePipeline](#) di Referensi AWS CLI Perintah.

## delete-image-recipe

Contoh kode berikut menunjukkan cara menggunakan `delete-image-recipe`.

## AWS CLI

Untuk menghapus resep gambar

`delete-image-recipe` Contoh berikut menghapus resep gambar dengan menentukan ARN nya.

```
aws imagebuilder delete-image-recipe \  
  --image-recipe-arn arn:aws:imagebuilder:us-east-1:123456789012:image-recipe/  
mybasicrecipe/2019.12.03
```

Output:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
mybasicrecipe/2019.12.03"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [DeleteImageRecipe](#) di Referensi AWS CLI Perintah.

## `delete-image`

Contoh kode berikut menunjukkan cara menggunakan `delete-image`.

## AWS CLI

Untuk menghapus gambar

`delete-image` Contoh berikut menghapus versi build gambar dengan menentukan ARN nya.

```
aws imagebuilder delete-image \  
  --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-  
example-image/2019.12.02/1
```

Output:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```



```
"imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [DeleteImage](#) di Referensi AWS CLI Perintah.

## **delete-infrastructure-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-infrastructure-configuration`.

### AWS CLI

Untuk menghapus konfigurasi infrastruktur

`delete-infrastructure-configuration` Contoh berikut menghapus pipeline gambar dengan menentukan ARN nya.

```
aws imagebuilder delete-infrastructure-configuration \
  --infrastructure-configuration-arn arn:aws:imagebuilder:us-
  east-1:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
  west-2:123456789012:infrastructure-configuration/myexampleinfrastructure"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [DeleteInfrastructureConfiguration](#) di Referensi AWS CLI Perintah.

## **get-component-policy**

Contoh kode berikut menunjukkan cara menggunakan `get-component-policy`.

## AWS CLI

Untuk mendapatkan detail kebijakan komponen

`get-component-policy` Contoh berikut mencantumkan rincian kebijakan komponen dengan menentukan ARN nya.

```
aws imagebuilder get-component-policy \  
  --component-arn arn:aws:imagebuilder:us-west-2:123456789012:component/my-  
example-component/2019.12.03/1
```

Output:

```
{  
  "Policy": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":  
\"Allow\", \"Principal\": { \"AWS\": [ \"123456789012\" ] }, \"Action\":  
[ \"imagebuilder:GetComponent\", \"imagebuilder:ListComponents\" ], \"Resource\":  
[ \"arn:aws:imagebuilder:us-west-2:123456789012:component/my-example-  
component/2019.12.03/1\" ] } ] }\"  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](https://docs.aws.amazon.com/imagebuilder/latest/userguide/) < [managing-image-builder-cli https://docs.aws.amazon.com/imagebuilder/latest/userguide/](https://docs.aws.amazon.com/imagebuilder/latest/userguide/) .html>`\_\_ di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetComponentPolicy](#) di Referensi AWS CLI Perintah.

## `get-component`

Contoh kode berikut menunjukkan cara menggunakan `get-component`.

## AWS CLI

Untuk mendapatkan detail komponen

`get-component` Contoh berikut mencantumkan rincian komponen dengan menentukan ARN nya.

```
aws imagebuilder get-component \  
  --component-build-version-arn arn:aws:imagebuilder:us-  
west-2:123456789012:component/component-name/1.0.0/1
```

**Output:**

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "component": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/component-
name/1.0.0/1",
    "name": "component-name",
    "version": "1.0.0",
    "type": "TEST",
    "platform": "Linux",
    "owner": "123456789012",
    "data": "name: HelloWorldTestingDocument\ndescription: This is hello world
testing document.\nschemaVersion: 1.0\n\nphases:\n - name: test\n  steps:\n
- name: HelloWorldStep\n      action: ExecuteBash\n      inputs:\n
commands:\n          - echo \"Hello World! Test.\\\"\\n\",
    "encrypted": true,
    "dateCreated": "2020-01-27T20:43:30.306Z",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetComponent](#) di Referensi AWS CLI Perintah.

**get-distribution-configuration**

Contoh kode berikut menunjukkan cara menggunakan `get-distribution-configuration`.

**AWS CLI**

Untuk mendapatkan detail konfigurasi distribusi

`get-distribution-configuration` Contoh berikut menampilkan rincian konfigurasi distribusi dengan menentukan ARN nya.

```
aws imagebuilder get-distribution-configuration \
  --distribution-configuration-arn arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution
```

## Output:

```

{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "distributionConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-
configuration/myexampledistribution",
    "name": "MyExampleDistribution",
    "description": "Copies AMI to eu-west-1 and exports to S3",
    "distributions": [
      {
        "region": "us-west-2",
        "amiDistributionConfiguration": {
          "name": "Name {{imagebuilder:buildDate}}",
          "description": "An example image name with parameter
references",
          "amiTags": {
            "KeyName": "{{ssm:parameter_name}}"
          },
          "launchPermission": {
            "userIds": [
              "123456789012"
            ]
          }
        }
      },
      {
        "region": "eu-west-1",
        "amiDistributionConfiguration": {
          "name": "My {{imagebuilder:buildVersion}} image
{{imagebuilder:buildDate}}",
          "amiTags": {
            "KeyName": "Value"
          },
          "launchPermission": {
            "userIds": [
              "123456789012"
            ]
          }
        }
      }
    ],
    "dateCreated": "2020-02-19T18:40:10.529Z",
    "tags": {}
  }
}

```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetDistributionConfiguration](#) di Referensi AWS CLI Perintah.

## get-image-pipeline

Contoh kode berikut menunjukkan cara menggunakan `get-image-pipeline`.

### AWS CLI

Untuk mendapatkan detail pipa gambar

`get-image-pipeline` Contoh berikut mencantumkan rincian pipeline gambar dengan menentukan ARN nya.

```
aws imagebuilder get-image-pipeline \  
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
  mywindows2016pipeline
```

### Output:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imagePipeline": {  
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline",  
    "name": "MyWindows2016Pipeline",  
    "description": "Builds Windows 2016 Images",  
    "platform": "Windows",  
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
mybasicrecipe/2019.12.03",  
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",  
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:distribution-configuration/myexampledistribution",  
    "imageTestsConfiguration": {  
      "imageTestsEnabled": true,  
      "timeoutMinutes": 60
```

```

    },
    "schedule": {
      "scheduleExpression": "cron(0 0 * * SUN)",
      "pipelineExecutionStartCondition":
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
    },
    "status": "ENABLED",
    "dateCreated": "2020-02-19T19:04:01.253Z",
    "dateUpdated": "2020-02-19T19:04:01.253Z",
    "tags": {}
  }
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetImagePipeline](#) di Referensi AWS CLI Perintah.

## get-image-policy

Contoh kode berikut menunjukkan cara menggunakan `get-image-policy`.

### AWS CLI

Untuk mendapatkan detail kebijakan gambar

`get-image-policy` Contoh berikut mencantumkan rincian kebijakan gambar dengan menentukan ARN-nya.

```

aws imagebuilder get-image-policy \
  --image-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-
image/2019.12.03/1

```

Output:

```

{
  "Policy": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\",
\"Principal\": { \"AWS\": [ \"123456789012\" ] }, \"Action\": [ \"imagebuilder:GetImage\",
\"imagebuilder:ListImages\" ], \"Resource\": [ \"arn:aws:imagebuilder:us-
west-2:123456789012:image/my-example-image/2019.12.03/1\" ] } ] }"
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetImagePolicy](#) di Referensi AWS CLI Perintah.

## get-image-recipe-policy

Contoh kode berikut menunjukkan cara menggunakan `get-image-recipe-policy`.

### AWS CLI

Untuk mendapatkan detail kebijakan resep gambar

`get-image-recipe-policy` Contoh berikut mencantumkan rincian kebijakan resep gambar dengan menentukan ARN-nya.

```
aws imagebuilder get-image-recipe-policy \
  --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-
  example-image-recipe/2019.12.03/1
```

Output:

```
{
  "Policy": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
  \"Allow\", \"Principal\": { \"AWS\": [ \"123456789012\" ] }, \"Action\":
  [ \"imagebuilder:GetImageRecipe\", \"imagebuilder:ListImageRecipes\" ], \"Resource\":
  [ \"arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-
  recipe/2019.12.03/1\" ] } ] }"
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetImageRecipePolicy](#) di Referensi AWS CLI Perintah.

## get-image

Contoh kode berikut menunjukkan cara menggunakan `get-image`.

### AWS CLI

Untuk mendapatkan detail gambar

`get-image` Contoh berikut mencantumkan detail gambar dengan menentukan ARN-nya.

```
aws imagebuilder get-image \  
  --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/  
mybasicrecipe/2019.12.03/1
```

Output:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "image": {  
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/  
mybasicrecipe/2019.12.03/1",  
    "name": "MyBasicRecipe",  
    "version": "2019.12.03/1",  
    "platform": "Windows",  
    "state": {  
      "status": "BUILDING"  
    },  
    "imageRecipe": {  
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
mybasicrecipe/2019.12.03",  
      "name": "MyBasicRecipe",  
      "description": "This example image recipe creates a Windows 2016  
image.",  
      "platform": "Windows",  
      "version": "2019.12.03",  
      "components": [  
        {  
          "componentArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:component/myexamplecomponent/2019.12.02/1"  
        },  
        {  
          "componentArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:component/myimportedcomponent/1.0.0/1"  
        }  
      ],  
      "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-  
server-2016-english-full-base-x86/2019.12.17/1",  
      "dateCreated": "2020-02-14T19:46:16.904Z",  
      "tags": {}  
    },  
    "infrastructureConfiguration": {
```



```
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/myexampleinfrastructure",
    "name": "MyExampleInfrastructure",
    "description": "An example that will retain instances of failed builds",
    "instanceTypes": [
        "m5.large",
        "m5.xlarge"
    ],
    "instanceProfileName": "EC2InstanceProfileForImageFactory",
    "securityGroupIds": [
        "sg-a1b2c3d4"
    ],
    "subnetId": "subnet-a1b2c3d4",
    "logging": {
        "s3Logs": {
            "s3BucketName": "bucket-name",
            "s3KeyPrefix": "bucket-path"
        }
    },
    "keyPair": "Sam",
    "terminateInstanceOnFailure": false,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name",
    "dateCreated": "2020-02-14T21:21:05.098Z",
    "tags": {}
},
"imageTestsConfiguration": {
    "imageTestsEnabled": true,
    "timeoutMinutes": 720
},
"dateCreated": "2020-02-14T23:14:13.597Z",
"outputResources": {
    "amis": []
},
"tags": {}
}
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetImage](#) di Referensi AWS CLI Perintah.

## get-infrastructure-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-infrastructure-configuration`.

### AWS CLI

Untuk mendapatkan detail konfigurasi infrastruktur

`get-infrastructure-configuration` Contoh berikut mencantumkan rincian konfigurasi infrastruktur dengan menentukan ARN nya.

```
aws imagebuilder get-infrastructure-configuration \
  --infrastructure-configuration-arn arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure
```

### Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "infrastructureConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/myexampleinfrastructure",
    "name": "MyExampleInfrastructure",
    "description": "An example that will retain instances of failed builds",
    "instanceTypes": [
      "m5.large",
      "m5.xlarge"
    ],
    "instanceProfileName": "EC2InstanceProfileForImageBuilder",
    "securityGroupIds": [
      "sg-a48c95ef"
    ],
    "subnetId": "subnet-a48c95ef",
    "logging": {
      "s3Logs": {
        "s3BucketName": "bucket-name",
        "s3KeyPrefix": "bucket-path"
      }
    },
    "keyPair": "Name",
    "terminateInstanceOnFailure": false,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name",
    "dateCreated": "2020-02-19T19:11:51.858Z",
```

```
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [GetInfrastructureConfiguration](#) di Referensi AWS CLI Perintah.

## import-component

Contoh kode berikut menunjukkan cara menggunakan `import-component`.

### AWS CLI

Untuk mengimpor komponen

`import-component` Contoh berikut mengimpor skrip yang sudah ada sebelumnya menggunakan file JSON.

```
aws imagebuilder import-component \
  --cli-input-json file://import-component.json
```

Isi dari `import-component.json`:

```
{
  "name": "MyImportedComponent",
  "semanticVersion": "1.0.0",
  "description": "An example of how to import a component",
  "changeDescription": "First commit message.",
  "format": "SHELL",
  "platform": "Windows",
  "type": "BUILD",
  "uri": "s3://s3-bucket-name/s3-bucket-path/component.yaml"
}
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
}
```

```
"clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
"componentBuildVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myimportedcomponent/1.0.0/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ImportComponent](#) di Referensi AWS CLI Perintah.

## list-component-build-versions

Contoh kode berikut menunjukkan cara menggunakan `list-component-build-versions`.

### AWS CLI

Untuk membuat daftar versi pembuatan komponen

`list-component-build-versions` Contoh berikut mencantumkan versi build komponen dengan versi semantik tertentu.

```
aws imagebuilder list-component-build-versions --component-version-arn
arn:aws:imagebuilder:us-west-2:123456789012:component/myexamplecomponent/2019.12.02
```

### Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "componentSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
myexamplecomponent/2019.12.02/1",
      "name": "MyExampleComponent",
      "version": "2019.12.02",
      "platform": "Windows",
      "type": "BUILD",
      "owner": "123456789012",
      "description": "An example component that builds, validates and tests an
image",
      "changeDescription": "Initial version.",
      "dateCreated": "2020-02-19T18:53:45.940Z",
      "tags": {
```

```
        "KeyName": "KeyValue"
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListComponentBuildVersions](#) di Referensi AWS CLI Perintah.

## list-components

Contoh kode berikut menunjukkan cara menggunakan `list-components`.

### AWS CLI

Untuk daftar semua versi semantik komponen

`list-components` Contoh berikut mencantumkan semua versi semantik komponen yang dapat Anda akses. Anda dapat secara opsional memfilter apakah akan mencantumkan komponen yang dimiliki oleh Anda, oleh Amazon, atau yang telah dibagikan dengan Anda oleh akun lain.

```
aws imagebuilder list-components
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "componentVersionList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/component-name/1.0.0",
      "name": "component-name",
      "version": "1.0.0",
      "platform": "Linux",
      "type": "TEST",
      "owner": "123456789012",
      "dateCreated": "2020-01-27T20:43:30.306Z"
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListComponents](#) di Referensi AWS CLI Perintah.

## list-distribution-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-distribution-configurations`.

### AWS CLI

Untuk daftar distribusi

`list-distribution-configurations` Contoh berikut mencantumkan semua distribusi Anda.

```
aws imagebuilder list-distribution-configurations
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "distributionConfigurationSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/myexampledistribution",
      "name": "MyExampleDistribution",
      "description": "Copies AMI to eu-west-1 and exports to S3",
      "dateCreated": "2020-02-19T18:40:10.529Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListDistributionConfigurations](#) di Referensi AWS CLI Perintah.

## list-image-build-versions

Contoh kode berikut menunjukkan cara menggunakan `list-image-build-versions`.

### AWS CLI

Untuk membuat daftar versi pembuatan gambar

`list-image-build-versions` Contoh berikut mencantumkan semua versi build image dengan versi semantik.

```
aws imagebuilder list-image-build-versions \
  --image-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03
```

### Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/7",
      "name": "MyBasicRecipe",
      "version": "2019.12.03/7",
      "platform": "Windows",
      "state": {
        "status": "FAILED",
        "reason": "Can't start SSM Automation for arn
arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/7 during
building. Parameter \"iamInstanceProfileName\" has a null value."
      },
      "owner": "123456789012",
      "dateCreated": "2020-02-19T18:56:11.511Z",
      "outputResources": {
        "amis": []
      },
      "tags": {}
    },
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/6",
      "name": "MyBasicRecipe",
```

```

    "version": "2019.12.03/6",
    "platform": "Windows",
    "state": {
      "status": "FAILED",
      "reason": "An internal error has occurred."
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-18T22:49:08.142Z",
    "outputResources": {
      "amis": [
        {
          "region": "us-west-2",
          "image": "ami-a1b2c3d4567890ab",
          "name": "MyBasicRecipe 2020-02-18T22-49-38.704Z",
          "description": "This example image recipe creates a Windows
2016 image."
        },
        {
          "region": "us-west-2",
          "image": "ami-a1b2c3d4567890ab",
          "name": "Name 2020-02-18T22-49-08.131Z",
          "description": "Copies AMI to eu-west-2 and exports to S3"
        },
        {
          "region": "eu-west-2",
          "image": "ami-a1b2c3d4567890ab",
          "name": "My 6 image 2020-02-18T22-49-08.131Z",
          "description": "Copies AMI to eu-west-2 and exports to S3"
        }
      ]
    },
    "tags": {}
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/5",
    "name": "MyBasicRecipe",
    "version": "2019.12.03/5",
    "platform": "Windows",
    "state": {
      "status": "AVAILABLE"
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-18T16:51:48.403Z",

```



```

    "outputResources": {
      "amis": [
        {
          "region": "us-west-2",
          "image": "ami-a1b2c3d4567890ab",
          "name": "MyBasicRecipe 2020-02-18T16-52-18.965Z",
          "description": "This example image recipe creates a Windows
2016 image."
        }
      ]
    },
    "tags": {}
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/4",
    "name": "MyBasicRecipe",
    "version": "2019.12.03/4",
    "platform": "Windows",
    "state": {
      "status": "AVAILABLE"
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-18T16:50:01.827Z",
    "outputResources": {
      "amis": [
        {
          "region": "us-west-2",
          "image": "ami-a1b2c3d4567890ab",
          "name": "MyBasicRecipe 2020-02-18T16-50-32.280Z",
          "description": "This example image recipe creates a Windows
2016 image."
        }
      ]
    },
    "tags": {}
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/3",
    "name": "MyBasicRecipe",
    "version": "2019.12.03/3",
    "platform": "Windows",
    "state": {

```

```

        "status": "AVAILABLE"
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-14T23:14:13.597Z",
    "outputResources": {
        "amis": [
            {
                "region": "us-west-2",
                "image": "ami-a1b2c3d4567890ab",
                "name": "MyBasicRecipe 2020-02-14T23-14-44.243Z",
                "description": "This example image recipe creates a Windows
2016 image."
            }
        ]
    },
    "tags": {}
},
{
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/2",
    "name": "MyBasicRecipe",
    "version": "2019.12.03/2",
    "platform": "Windows",
    "state": {
        "status": "FAILED",
        "reason": "SSM execution 'a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'
failed with status = 'Failed' and failure message = 'Step fails when it is
verifying the command has completed. Command a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
returns unexpected invocation result: \n{Status=[Failed], ResponseCode=[1],
Output=[\n-----ERROR-----\nfailed to run commands: exit status 1],
OutputPayload=[{\"Status\": \"Failed\", \"ResponseCode\": 1, \"Output\": \"\
\n-----ERROR-----\nfailed to run commands: exit status 1\", \"CommandId\":
\"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111\"}], CommandId=[a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111]}. Please refer to Automation Service Troubleshooting Guide for more
diagnosis details.'"
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-14T22:57:42.593Z",
    "outputResources": {
        "amis": []
    },
    "tags": {}
}
]

```

```
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListImageBuildVersions](#) di Referensi AWS CLI Perintah.

## list-image-pipeline-images

Contoh kode berikut menunjukkan cara menggunakan `list-image-pipeline-images`.

### AWS CLI

Untuk mencantumkan gambar pipa pipa gambar

`list-image-pipeline-images` Contoh berikut mencantumkan semua gambar yang dibuat oleh pipeline gambar tertentu.

```
aws imagebuilder list-image-pipeline-images \
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

### Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imagePipelineList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
      "name": "MyWindows2016Pipeline",
      "description": "Builds Windows 2016 Images",
      "platform": "Windows",
      "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
recipe/mybasicrecipe/2019.12.03",
      "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
      "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
      "imageTestsConfiguration": {
        "imageTestsEnabled": true,
        "timeoutMinutes": 60
      },
    },
  ],
}
```

```

    "schedule": {
      "scheduleExpression": "cron(0 0 * * SUN)",
      "pipelineExecutionStartCondition":
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
    },
    "status": "ENABLED",
    "dateCreated": "2020-02-19T19:04:01.253Z",
    "dateUpdated": "2020-02-19T19:04:01.253Z",
    "tags": {
      "KeyName": "KeyValue"
    }
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/sam",
    "name": "PipelineName",
    "platform": "Linux",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
recipe/recipe-name-a1b2c3d45678/1.0.0",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/infrastructureconfiguration-name-
a1b2c3d45678",
    "imageTestsConfiguration": {
      "imageTestsEnabled": true,
      "timeoutMinutes": 720
    },
    "status": "ENABLED",
    "dateCreated": "2019-12-16T18:19:02.068Z",
    "dateUpdated": "2019-12-16T18:19:02.068Z",
    "tags": {
      "KeyName": "KeyValue"
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListImagePipelineImages](#) di Referensi AWS CLI Perintah.

## list-image-recipes

Contoh kode berikut menunjukkan cara menggunakan `list-image-recipes`.

## AWS CLI

Untuk daftar resep gambar

`list-image-recipes` Contoh berikut mencantumkan semua resep gambar Anda.

```
aws imagebuilder list-image-recipes
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageRecipeSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/mybasicrecipe/2019.12.03",
      "name": "MyBasicRecipe",
      "platform": "Windows",
      "owner": "123456789012",
      "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-english-full-base-x86/2019.x.x",
      "dateCreated": "2020-02-19T18:54:25.975Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    },
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/recipe-name-a1b2c3d45678/1.0.0",
      "name": "recipe-name-a1b2c3d45678",
      "platform": "Linux",
      "owner": "123456789012",
      "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/amazon-linux-2-x86/2019.11.21",
      "dateCreated": "2019-12-16T18:19:00.120Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListImageRecipes](#) di Referensi AWS CLI Perintah.

## list-images

Contoh kode berikut menunjukkan cara menggunakan `list-images`.

### AWS CLI

Untuk membuat daftar gambar

`list-images` Contoh berikut mencantumkan semua versi semantik yang dapat Anda akses.

```
aws imagebuilder list-images
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageVersionList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03",
      "name": "MyBasicRecipe",
      "version": "2019.12.03",
      "platform": "Windows",
      "owner": "123456789012",
      "dateCreated": "2020-02-14T21:29:18.810Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListImages](#) di Referensi AWS CLI Perintah.

## list-infrastructure-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-infrastructure-configurations`.

## AWS CLI

Untuk membuat daftar konfigurasi infrastruktur

`list-infrastructure-configurations` Contoh berikut mencantumkan semua konfigurasi infrastruktur Anda.

```
aws imagebuilder list-infrastructure-configurations
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "infrastructureConfigurationSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
      "name": "MyExampleInfrastructure",
      "description": "An example that will retain instances of failed builds",
      "dateCreated": "2020-02-19T19:11:51.858Z",
      "tags": {}
    },
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/infrastructureconfiguration-name-a1b2c3d45678",
      "name": "infrastructureConfiguration-name-a1b2c3d45678",
      "dateCreated": "2019-12-16T18:19:01.038Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListInfrastructureConfigurations](#) di Referensi AWS CLI Perintah.

### **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk membuat daftar tag untuk sumber daya tertentu

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk sumber daya tertentu.

```
aws imagebuilder list-tags-for-resource \  
  --resource-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
  mywindows2016pipeline
```

Output:

```
{  
  "tags": {  
    "KeyName": "KeyValue"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **put-component-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-component-policy`.

## AWS CLI

Untuk menerapkan kebijakan sumber daya ke komponen

`put-component-policy` Perintah berikut menerapkan kebijakan sumber daya ke komponen build untuk mengaktifkan berbagi komponen build lintas akun. Kami sarankan Anda menggunakan perintah RAM CLI `create-resource-share` Jika Anda menggunakan perintah EC2 Image Builder CLI, Anda juga harus menggunakan `put-component-policy` perintah RAM CLI `promote-resource-share-create-from-policy` agar sumber daya dapat dilihat oleh semua kepala sekolah dengan siapa sumber daya dibagikan.

```
aws imagebuilder put-component-policy \  
  --resource-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
  mywindows2016pipeline
```



```
--component-arn arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/2019.12.02/1 \
--policy '{ "Version": "2012-10-17", "Statement": [ { "Effect":
"Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
[ "imagebuilder:GetComponent", "imagebuilder:ListComponents" ],
"Resource": [ "arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/2019.12.02/1" ] } ] }'
```

### Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/2019.12.02/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [PutComponentPolicy](#) di Referensi AWS CLI Perintah.

## put-image-policy

Contoh kode berikut menunjukkan cara menggunakan `put-image-policy`.

### AWS CLI

Untuk menerapkan kebijakan sumber daya ke gambar

`put-image-policy` Perintah berikut menerapkan kebijakan sumber daya ke gambar untuk mengaktifkan berbagi gambar lintas akun. Kami sarankan Anda menggunakan perintah RAM CLI `create-resource-share` Jika Anda menggunakan perintah EC2 Image Builder CLI, Anda juga harus menggunakan `put-image-policy` perintah RAM CLI `promote-resource-share-create -from-policy` agar sumber daya dapat dilihat oleh semua prinsipal dengan siapa sumber daya dibagikan.

```
aws imagebuilder put-image-policy \
--image-arn arn:aws:imagebuilder:us-west-2:123456789012:image/example-
image/2019.12.02/1 \
--policy '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow",
"Principal": { "AWS": [ "123456789012" ] }, "Action": [ "imagebuilder:GetImage",
"imagebuilder:ListImages" ], "Resource": [ "arn:aws:imagebuilder:us-
west-2:123456789012:image/example-image/2019.12.02/1" ] } ] }'
```

**Output:**

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/example-
image/2019.12.02/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [PutImagePolicy](#) di Referensi AWS CLI Perintah.

**put-image-recipe-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-image-recipe-policy`.

**AWS CLI**

Untuk menerapkan kebijakan sumber daya ke resep gambar

`put-image-recipe-policy` Perintah berikut menerapkan kebijakan sumber daya ke resep gambar untuk mengaktifkan berbagi resep gambar lintas akun. Kami menyarankan Anda menggunakan perintah RAM CLI `create-resource-share` Jika Anda menggunakan perintah EC2 Image Builder CLI, Anda juga harus menggunakan `put-image-recipe-policy` perintah RAM CLI `promote-resource-share-create-from-policy` agar sumber daya dapat dilihat oleh semua kepala sekolah dengan siapa sumber daya dibagikan.

```
aws imagebuilder put-image-recipe-policy \
  --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
example-image-recipe/2019.12.02 \
  --policy '{ "Version": "2012-10-17", "Statement": [ { "Effect":
"Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
[ "imagebuilder:GetImageRecipe", "imagebuilder:ListImageRecipes" ], "Resource":
[ "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/example-image-
recipe/2019.12.02" ] } ] }'
```

**Output:**

```
{
```

```
"requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
example-image-recipe/2019.12.02/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [PutImageRecipePolicy](#) di Referensi AWS CLI Perintah.

## start-image-pipeline-execution

Contoh kode berikut menunjukkan cara menggunakan `start-image-pipeline-execution`.

### AWS CLI

Untuk memulai pipeline gambar secara manual

`start-image-pipeline-execution` Contoh berikut secara manual memulai pipeline gambar.

```
aws imagebuilder start-image-pipeline-execution \
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/1"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [StartImagePipelineExecution](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menandai sumber daya

`tag-resource` Contoh berikut menambahkan dan menandai sumber daya ke EC2 Image Builder menggunakan file JSON.

```
aws imagebuilder tag-resource \  
  --cli-input-json file://tag-resource.json
```

Isi dari `tag-resource.json`:

```
{  
  "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline",  
  "tags": {  
    "KeyName": "KeyValue"  
  }  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dari sumber daya menggunakan file JSON.

```
aws imagebuilder untag-resource \  
  --cli-input-json file://tag-resource.json
```

Isi dari `untag-resource.json`:

```
{
  "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
  "tagKeys": [
    "KeyName"
  ]
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-distribution-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-distribution-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi distribusi

`update-distribution-configuration` Contoh berikut memperbarui konfigurasi distribusi menggunakan file JSON.

```
aws imagebuilder update-distribution-configuration \
  --cli-input-json file://update-distribution-configuration.json
```

Isi dari `update-distribution-configuration.json`:

```
{
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexempldistribution",
  "description": "Copies AMI to eu-west-2 and exports to S3",
  "distributions": [
    {
      "region": "us-west-2",
      "amiDistributionConfiguration": {
        "name": "Name {{imagebuilder:buildDate}}",
        "description": "An example image name with parameter references"
      }
    }
  ]
}
```

```

    },
    {
      "region": "eu-west-2",
      "amiDistributionConfiguration": {
        "name": "My {{imagebuilder:buildVersion}} image
        {{imagebuilder:buildDate}}"
      }
    }
  ]
}

```

Output:

```

{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}

```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [UpdateDistributionConfiguration](#) di Referensi AWS CLI Perintah.

## update-image-pipeline

Contoh kode berikut menunjukkan cara menggunakan `update-image-pipeline`.

### AWS CLI

Untuk memperbarui pipeline gambar

`update-image-pipeline` Contoh berikut memperbarui pipeline gambar menggunakan file JSON.

```

aws imagebuilder update-image-pipeline \
  --cli-input-json file://update-image-pipeline.json

```

Isi dari `update-image-pipeline.json`:

```

{
  "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
  mywindows2016pipeline",

```

```
"imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/mybasicrecipe/2019.12.03",
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/myexampledistribution",
  "imageTestsConfiguration": {
    "imageTestsEnabled": true,
    "timeoutMinutes": 120
  },
  "schedule": {
    "scheduleExpression": "cron(0 0 * * MON)",
    "pipelineExecutionStartCondition":
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
  },
  "status": "DISABLED"
}
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [UpdateImagePipeline](#) di Referensi AWS CLI Perintah.

## update-infrastructure-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-infrastructure-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi infrastruktur

`update-infrastructure-configuration` Contoh berikut memperbarui konfigurasi infrastruktur menggunakan file JSON.

```
aws imagebuilder update-infrastructure-configuration \
```

```
--cli-input-json file:/update-infrastructure-configuration.json
```

Isi dari `update-infrastructure-configuration.json`:

```
{
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
  "description": "An example that will terminate instances of failed builds",
  "instanceTypes": [
    "m5.large", "m5.2xlarge"
  ],
  "instanceProfileName": "EC2InstanceProfileForImageFactory",
  "securityGroupIds": [
    "sg-a48c95ef"
  ],
  "subnetId": "subnet-a48c95ef",
  "logging": {
    "s3Logs": {
      "s3BucketName": "bucket-name",
      "s3KeyPrefix": "bucket-path"
    }
  },
  "terminateInstanceOnFailure": true,
  "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name"
}
```

Output:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan dan Mengelola Pipeline Image Builder EC2 Menggunakan AWS CLI](#) di Panduan Pengguna EC2 Image Builder.

- Untuk detail API, lihat [UpdateInfrastructureConfiguration](#) di Referensi AWS CLI Perintah.

## Contoh Manajer Insiden menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Incident Manager.



Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-replication-set**

Contoh kode berikut menunjukkan cara menggunakan `create-replication-set`.

AWS CLI

Untuk membuat set replikasi

`create-replication-set` Contoh berikut membuat kumpulan replikasi yang digunakan Incident Manager untuk mereplikasi dan mengenkripsi data di akun Amazon Web Services Anda. Contoh ini menggunakan Wilayah `us-east-1` dan `us-east-2` saat membuat kumpulan replikasi.

```
aws ssm-incidents create-replication-set \
  --regions '{"us-east-1": {"sseKmsKeyId": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"}, "us-east-2":
{"sseKmsKeyId": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"}}'
```

Output:

```
{
  "replicationSetArns": [
    "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
bb3f-413c-08df53673b57"
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan replikasi Manajer Insiden yang diatur](#) dalam Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [CreateReplicationSet](#) di Referensi AWS CLI Perintah.

## create-response-plan

Contoh kode berikut menunjukkan cara menggunakan `create-response-plan`.

### AWS CLI

Untuk membuat rencana respons

`create-response-plan` Contoh berikut membuat rencana respons dengan rincian yang ditentukan.

```
aws ssm-incidents create-response-plan \
  --chat-channel '{"chatbotSns": ["arn:aws:sns:us-east-1:111122223333:Standard_User"]}' \
  --display-name "Example response plan" \
  --incident-template '{"impact": 5, "title": "example-incident"}' \
  --name "example-response" \
  --actions '[{"ssmAutomation": {"documentName": "AWSIncidents-CriticalIncidentRunbookTemplate", "documentVersion": "$DEFAULT", "roleArn": "arn:aws:iam::111122223333:role/aws-service-role/ssm-incidents.amazonaws.com/AWSServiceRoleForIncidentManager", "targetAccount": "RESPONSE_PLAN_OWNER_ACCOUNT"}}]' \
  --engagements '["arn:aws:ssm-contacts:us-east-1:111122223333:contact/example"]'
```

Output:

```
{
  "arn": "arn:aws:ssm-incidents::111122223333:response-plan/example-response"
}
```

Untuk informasi selengkapnya, lihat [Persiapan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [CreateResponsePlan](#) di Referensi AWS CLI Perintah.

## create-timeline-event

Contoh kode berikut menunjukkan cara menggunakan `create-timeline-event`.

## AWS CLI

### Contoh 1: Untuk membuat acara timeline kustom

`create-timeline-event` Contoh berikut membuat peristiwa timeline kustom pada waktu yang ditentukan pada insiden yang ditentukan.

```
aws ssm-incidents create-timeline-event \  
  --event-data "\"example timeline event\"" \  
  --event-time 2022-10-01T20:30:00.000 \  
  --event-type "Custom Event" \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
```

### Output:

```
{  
  "eventId": "c0bcc885-a41d-eb01-b4ab-9d2deEXAMPLE",  
  "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"  
}
```

### Contoh 2: Untuk membuat peristiwa timeline dengan catatan insiden

`create-timeline-event` Contoh berikut membuat peristiwa timeline yang tercantum di panel 'Catatan kejadian'.

```
aws ssm-incidents create-timeline-event \  
  --event-data "\"New Note\"" \  
  --event-type "Note" \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Test/6cc46130-ca6c-3b38-68f1-f6abeEXAMPLE" \  
  --event-time 2023-06-20T12:06:00.000 \  
  --event-references '[{"resource": "arn:aws:ssm-incidents::111122223333:incident-  
record/Test/6cc46130-ca6c-3b38-68f1-f6abeEXAMPLE"}]'
```

### Output:

```
{  
  "eventId": "a41dc885-c0bc-b4ab-eb01-de9d2EXAMPLE",  
  "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
```

```
}
```

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [CreateTimelineEvent](#) di Referensi AWS CLI Perintah.

## **delete-incident-record**

Contoh kode berikut menunjukkan cara menggunakan `delete-incident-record`.

### AWS CLI

Untuk menghapus catatan insiden

`delete-incident-record` Contoh berikut menghapus catatan insiden yang ditentukan.

```
aws ssm-incidents delete-incident-record \  
  --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-  
Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pelacakan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteIncidentRecord](#) di Referensi AWS CLI Perintah.

## **delete-replication-set**

Contoh kode berikut menunjukkan cara menggunakan `delete-replication-set`.

### AWS CLI

Untuk menghapus set replikasi

`delete-replication-set` Contoh berikut menghapus kumpulan replikasi dari akun Amazon Web Services Anda. Menghapus set replikasi juga menghapus semua data Manajer Insiden. Ini tidak bisa dibatalkan.

```
aws ssm-incidents delete-replication-set \  
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-  
bb3f-413c-08df53673b57"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan replikasi Manajer Insiden yang diatur](#) dalam Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteReplicationSet](#) di Referensi AWS CLI Perintah.

## **delete-resource-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-policy`.

### AWS CLI

Untuk menghapus kebijakan sumber daya

`delete-resource-policy` Contoh berikut menghapus kebijakan sumber daya dari rencana respons. Ini akan mencabut akses dari kepala sekolah atau organisasi tempat rencana respons dibagikan.

```
aws ssm-incidents delete-resource-policy \  
  --policy-id "be8b57191f0371f1c6827341aa3f0a03" \  
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan kontak bersama dan rencana respons](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteResourcePolicy](#) di Referensi AWS CLI Perintah.

## **delete-response-plan**

Contoh kode berikut menunjukkan cara menggunakan `delete-response-plan`.

### AWS CLI

Untuk menghapus rencana respons

`delete-response-plan` Contoh berikut menghapus rencana respons yang ditentukan.

```
aws ssm-incidents delete-response-plan \  
  --response-plan-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

```
--arn "arn:aws:ssm-incidents::111122223333:response-plan/example-response"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Persiapan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteResponsePlan](#) di Referensi AWS CLI Perintah.

## **delete-timeline-event**

Contoh kode berikut menunjukkan cara menggunakan `delete-timeline-event`.

### AWS CLI

Untuk menghapus peristiwa timeline

`delete-timeline-event` Contoh berikut menghapus peristiwa timeline kustom dari catatan insiden yang ditentukan.

```
aws ssm-incidents delete-timeline-event \  
  --event-id "c0bcc885-a41d-eb01-b4ab-9d2de193643c" \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteTimelineEvent](#) di Referensi AWS CLI Perintah.

## **get-incident-record**

Contoh kode berikut menunjukkan cara menggunakan `get-incident-record`.

### AWS CLI

Untuk mendapatkan catatan insiden

`get-incident-record` Contoh berikut mendapatkan rincian tentang catatan insiden yang ditentukan.

```
aws ssm-incidents get-incident-record \  
  --event-id "c0bcc885-a41d-eb01-b4ab-9d2de193643c" \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

```
--arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

### Output:

```
{
  "incidentRecord": {
    "arn": "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308",
    "automationExecutions": [],
    "creationTime": "2021-05-21T18:16:57.579000+00:00",
    "dedupeString": "c4bcc812-85e7-938d-2b78-17181176ee1a",
    "impact": 5,
    "incidentRecordSource": {
      "createdBy": "arn:aws:iam::111122223333:user/drალიატპ",
      "invokedBy": "arn:aws:iam::111122223333:user/drალიატპ",
      "source": "aws.ssm-incidents.custom"
    },
    "lastModifiedBy": "arn:aws:iam::111122223333:user/drალიატპ",
    "lastModifiedTime": "2021-05-21T18:16:59.149000+00:00",
    "notificationTargets": [],
    "status": "OPEN",
    "title": "Example-Incident"
  }
}
```

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetIncidentRecord](#) di Referensi AWS CLI Perintah.

## get-replication-set

Contoh kode berikut menunjukkan cara menggunakan `get-replication-set`.

### AWS CLI

Untuk mendapatkan set replikasi

`get-replication-set` Contoh berikut mendapatkan detail kumpulan replikasi yang digunakan Incident Manager untuk mereplikasi dan mengenkripsi data di akun Amazon Web Services Anda.

```
aws ssm-incidents get-replication-set \
```

```
--arn "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-  
bb3f-413c-08df53673b57"
```

Output:

```
{  
  "replicationSet": {  
    "createdBy": "arn:aws:sts::111122223333:assumed-role/Admin/username",  
    "createdTime": "2021-05-14T17:57:22.010000+00:00",  
    "deletionProtected": false,  
    "lastModifiedBy": "arn:aws:sts::111122223333:assumed-role/Admin/username",  
    "lastModifiedTime": "2021-05-14T17:57:22.010000+00:00",  
    "regionMap": {  
      "us-east-1": {  
        "sseKmsKeyId": "DefaultKey",  
        "status": "ACTIVE"  
      },  
      "us-east-2": {  
        "sseKmsKeyId": "DefaultKey",  
        "status": "ACTIVE",  
        "statusMessage": "Tagging inaccessible"  
      }  
    },  
    "status": "ACTIVE"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan replikasi Manajer Insiden yang diatur](#) dalam Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetReplicationSet](#) di Referensi AWS CLI Perintah.

## get-resource-policies

Contoh kode berikut menunjukkan cara menggunakan `get-resource-policies`.

### AWS CLI

Untuk membuat daftar kebijakan sumber daya untuk rencana respons

`command-name` Contoh berikut mencantumkan kebijakan sumber daya yang terkait dengan rencana respons yang ditentukan.



```
aws ssm-incidents get-resource-policies \
--resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

### Output:

```
{
  "resourcePolicies": [
    {
      "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":\"d901b37a-dbb0-458a-8842-75575c464219-external-principals\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::222233334444:root\"},\"Action\":[\"ssm-incidents:GetResponsePlan\",\"ssm-incidents:StartIncident\",\"ssm-incidents:UpdateIncidentRecord\",\"ssm-incidents:GetIncidentRecord\",\"ssm-incidents:CreateTimelineEvent\",\"ssm-incidents:UpdateTimelineEvent\",\"ssm-incidents:GetTimelineEvent\",\"ssm-incidents:ListTimelineEvents\",\"ssm-incidents:UpdateRelatedItems\",\"ssm-incidents:ListRelatedItems\"]},\"Resource\":[\"arn:aws:ssm-incidents*:111122223333:response-plan/Example-Response-Plan\",\"arn:aws:ssm-incidents*:111122223333:incident-record/Example-Response-Plan/*\"]}]}\",
      "policyId": "be8b57191f0371f1c6827341aa3f0a03",
      "ramResourceShareRegion": "us-east-1"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan kontak bersama dan rencana respons](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetResourcePolicies](#) di Referensi AWS CLI Perintah.

## get-response-plan

Contoh kode berikut menunjukkan cara menggunakan `get-response-plan`.

### AWS CLI

Untuk mendapatkan rincian rencana respons

command-name Contoh berikut mendapatkan detail tentang rencana respons tertentu di AWS akun Anda.

```
aws ssm-incidents get-response-plan \  
  --arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

### Output:

```
{  
  "actions": [  
    {  
      "ssmAutomation": {  
        "documentName": "AWSIncidents-CriticalIncidentRunbookTemplate",  
        "documentVersion": "$DEFAULT",  
        "roleArn": "arn:aws:iam::111122223333:role/aws-service-role/ssm-  
incidents.amazonaws.com/AWSServiceRoleForIncidentManager",  
        "targetAccount": "RESPONSE_PLAN_OWNER_ACCOUNT"  
      }  
    }  
  ],  
  "arn": "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-  
Plan",  
  "chatChannel": {  
    "chatbotSns": [  
      "arn:aws:sns:us-east-1:111122223333:Standard_User"  
    ]  
  },  
  "displayName": "Example response plan",  
  "engagements": [  
    "arn:aws:ssm-contacts:us-east-1:111122223333:contact/example"  
  ],  
  "incidentTemplate": {  
    "impact": 5,  
    "title": "Example-Incident"  
  },  
  "name": "Example-Response-Plan"  
}
```

Untuk informasi selengkapnya, lihat [Persiapan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetResponsePlan](#) di Referensi AWS CLI Perintah.

## get-timeline-event

Contoh kode berikut menunjukkan cara menggunakan `get-timeline-event`.

## AWS CLI

Untuk mendapatkan detail acara timeline

`get-timeline-event` Contoh berikut mengembalikan rincian peristiwa timeline tertentu.

```
aws ssm-incidents get-timeline-event \  
  --event-id 20bcc812-8a94-4cd7-520c-0ff742111424 \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Output:

```
{  
  "event": {  
    "eventData": "\"Incident Started\"",  
    "eventId": "20bcc812-8a94-4cd7-520c-0ff742111424",  
    "eventTime": "2021-05-21T18:16:57+00:00",  
    "eventType": "Custom Event",  
    "eventUpdatedTime": "2021-05-21T18:16:59.944000+00:00",  
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetTimelineEvent](#) di Referensi AWS CLI Perintah.

## list-incident-records

Contoh kode berikut menunjukkan cara menggunakan `list-incident-records`.

### AWS CLI

Untuk membuat daftar catatan insiden

`command-name` Contoh berikut mencantumkan catatan insiden di akun Amazon Web Services Anda.

```
aws ssm-incidents list-incident-records
```

**Output:**

```
{
  "incidentRecordSummaries": [
    {
      "arn": "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308",
      "creationTime": "2021-05-21T18:16:57.579000+00:00",
      "impact": 5,
      "incidentRecordSource": {
        "createdBy": "arn:aws:iam::111122223333:user/draliatp",
        "invokedBy": "arn:aws:iam::111122223333:user/draliatp",
        "source": "aws.ssm-incidents.custom"
      },
      "status": "OPEN",
      "title": "Example-Incident"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Daftar insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListIncidentRecords](#) di Referensi AWS CLI Perintah.

**list-related-items**

Contoh kode berikut menunjukkan cara menggunakan `list-related-items`.

**AWS CLI**

Untuk daftar item terkait

`list-related-items` Contoh berikut mencantumkan item terkait dari insiden yang ditentukan.

```
aws ssm-incidents list-related-items \
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

**Output:**

```
{
```

```

    "relatedItems": [
      {
        "identifier": {
          "type": "OTHER",
          "value": {
            "url": "https://console.aws.amazon.com/systems-manager/opsitems/oi-8ef82158e190/workbench?region=us-east-1"
          }
        },
        "title": "Example related item"
      },
      {
        "identifier": {
          "type": "PARENT",
          "value": {
            "arn": "arn:aws:ssm:us-east-1:111122223333:opsitem/oi-8084126392ac"
          }
        },
        "title": "parentItem"
      }
    ]
  }

```

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListRelatedItems](#) di Referensi AWS CLI Perintah.

## list-replication-sets

Contoh kode berikut menunjukkan cara menggunakan `list-replication-sets`.

### AWS CLI

Untuk daftar set replikasi

`list-replication-set` Contoh berikut mencantumkan kumpulan replikasi yang digunakan Incident Manager untuk mereplikasi dan mengenkripsi data di akun Anda. AWS

```
aws ssm-incidents list-replication-sets
```

Output:

```
{
  "replicationSetArns": [
    "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
bb3f-413c-08df53673b57"
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan replikasi Manajer Insiden yang diatur](#) dalam Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListReplicationSets](#) di Referensi AWS CLI Perintah.

## list-response-plans

Contoh kode berikut menunjukkan cara menggunakan `list-response-plans`.

### AWS CLI

Untuk membuat daftar rencana respons yang tersedia

`list-response-plans` Contoh berikut mencantumkan paket respons yang tersedia di akun Amazon Web Services Anda.

```
aws ssm-incidents list-response-plans
```

Output:

```
{
  "responsePlanSummaries": [
    {
      "arn": "arn:aws:ssm-incidents::111122223333:response-plan/Example-
Response-Plan",
      "displayName": "Example response plan",
      "name": "Example-Response-Plan"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Persiapan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListResponsePlans](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag untuk rencana respons

`list-tags-for-resource` Contoh berikut mencantumkan tag yang terkait dengan rencana respons yang ditentukan.

```
aws ssm-incidents list-tags-for-resource \  
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-  
Response-Plan"
```

Output:

```
{  
  "tags": {  
    "group1": "1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menandai](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## list-timeline-events

Contoh kode berikut menunjukkan cara menggunakan `list-timeline-events`.

### AWS CLI

Untuk membuat daftar peristiwa timeline dari suatu insiden

`command-name` Contoh berikut mencantumkan peristiwa timeline dari insiden yang ditentukan.

```
aws ssm-incidents list-timeline-events \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

## Output:

```
{
  "eventSummaries": [
    {
      "eventId": "8cbcc889-35e1-a42d-2429-d6f100799915",
      "eventTime": "2021-05-21T22:36:13.766000+00:00",
      "eventType": "SSM Incident Record Update",
      "eventUpdatedTime": "2021-05-21T22:36:13.766000+00:00",
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    },
    {
      "eventId": "a2bcc825-aab5-1787-c605-f9bb2640d85b",
      "eventTime": "2021-05-21T18:58:46.443000+00:00",
      "eventType": "SSM Incident Record Update",
      "eventUpdatedTime": "2021-05-21T18:58:46.443000+00:00",
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    },
    {
      "eventId": "5abcc812-89c0-b0a8-9437-1c74223d4685",
      "eventTime": "2021-05-21T18:16:59.149000+00:00",
      "eventType": "SSM Incident Record Update",
      "eventUpdatedTime": "2021-05-21T18:16:59.149000+00:00",
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    },
    {
      "eventId": "06bcc812-8820-405e-4065-8d2b14d29b92",
      "eventTime": "2021-05-21T18:16:58+00:00",
      "eventType": "SSM Automation Execution Start Failure for Incident",
      "eventUpdatedTime": "2021-05-21T18:16:58.689000+00:00",
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    },
    {
      "eventId": "20bcc812-8a94-4cd7-520c-0ff742111424",
      "eventTime": "2021-05-21T18:16:57+00:00",
      "eventType": "Custom Event",
      "eventUpdatedTime": "2021-05-21T18:16:59.944000+00:00",
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    },
  ],
}
```



```

    {
      "eventId": "c0bcc885-a41d-eb01-b4ab-9d2de193643c",
      "eventTime": "2020-10-01T20:30:00+00:00",
      "eventType": "Custom Event",
      "eventUpdatedTime": "2021-05-21T22:28:26.299000+00:00",
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListTimelineEvents](#) di Referensi AWS CLI Perintah.

## put-resource-policy

Contoh kode berikut menunjukkan cara menggunakan `put-resource-policy`.

### AWS CLI

Untuk berbagi rencana respons dan insiden

command-name Contoh berikut menambahkan kebijakan sumber daya ke `Example-Response-Plan` yang membagikan rencana respons dan insiden terkait dengan prinsipal yang ditentukan.

```

aws ssm-incidents put-resource-policy \
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-
Response-Plan" \
  --policy "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"ExampleResourcePolciy\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
\"arn:aws:iam::222233334444:root\"},\"Action\":[\"ssm-incidents:GetResponsePlan
\",\"ssm-incidents:StartIncident\",\"ssm-incidents:UpdateIncidentRecord
\",\"ssm-incidents:GetIncidentRecord\",\"ssm-incidents:CreateTimelineEvent
\",\"ssm-incidents:UpdateTimelineEvent\",\"ssm-incidents:GetTimelineEvent
\",\"ssm-incidents:ListTimelineEvents\",\"ssm-incidents:UpdateRelatedItems
\",\"ssm-incidents:ListRelatedItems\"],\"Resource\":[\"arn:aws:ssm-
incidents:*:111122223333:response-plan/Example-Response-Plan\",\"arn:aws:ssm-
incidents:*:111122223333:incident-record/Example-Response-Plan/*\"]}]}"

```

Output:

```
{
```

```
"policyId": "be8b57191f0371f1c6827341aa3f0a03"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan kontak bersama dan rencana respons](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [PutResourcePolicy](#) di Referensi AWS CLI Perintah.

## start-incident

Contoh kode berikut menunjukkan cara menggunakan `start-incident`.

### AWS CLI

Untuk memulai sebuah insiden

`start-incident` Contoh berikut memulai insiden menggunakan rencana respons yang ditentukan.

```
aws ssm-incidents start-incident \  
  --response-plan-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-  
  Response-Plan"
```

Output:

```
{  
  "incidentRecordArn": "arn:aws:ssm-incidents::682428703967:incident-record/  
  Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"  
}
```

Untuk informasi selengkapnya, lihat [Pembuatan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [StartIncident](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai rencana respons

`tag-resource` Contoh berikut menandai rencana respons tertentu dengan pasangan nilai kunci tag yang disediakan.

```
aws ssm-incidents tag-resource \  
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan" \  
  --tags '{"group1":"1"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari rencana respons

`untag-resource` Contoh berikut menghapus tag yang ditentukan dari rencana respons.

```
aws ssm-incidents untag-resource \  
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan" \  
  --tag-keys '["group1"]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-deletion-protection**

Contoh kode berikut menunjukkan cara menggunakan `update-deletion-protection`.

### AWS CLI

Untuk memperbarui replikasi set perlindungan penghapusan

`update-deletion-protection` Contoh berikut memperbarui perlindungan penghapusan di akun Anda untuk melindungi Anda dari menghapus Wilayah terakhir dalam kumpulan replikasi Anda.

```
aws ssm-incidents update-deletion-protection \  
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/  
a2bcc5c9-0f53-8047-7fef-c20749989b40" \  
  --deletion-protected
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan replikasi Manajer Insiden yang diatur](#) dalam Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateDeletionProtection](#) di Referensi AWS CLI Perintah.

## **update-incident-record**

Contoh kode berikut menunjukkan cara menggunakan `update-incident-record`.

### AWS CLI

Untuk memperbarui catatan insiden

`command-name` Contoh berikut menyelesaikan insiden yang ditentukan.

```
aws ssm-incidents update-incident-record \  
  --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-  
Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \  
  --status "RESOLVED"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateIncidentRecord](#) di Referensi AWS CLI Perintah.

## **update-related-items**

Contoh kode berikut menunjukkan cara menggunakan `update-related-items`.

## AWS CLI

Untuk memperbarui item terkait insiden

`update-related-item` Contoh berikut menghapus item terkait dari catatan insiden yang ditentukan.

```
aws ssm-incidents update-related-items \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \  
  --related-items-update '{"itemToRemove": {"type": "OTHER", "value": {"url":  
"https://console.aws.amazon.com/systems-manager/opsitems/oi-8ef82158e190/workbench?  
region=us-east-1"}}}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateRelatedItems](#) di Referensi AWS CLI Perintah.

## update-replication-set

Contoh kode berikut menunjukkan cara menggunakan `update-replication-set`.

### AWS CLI

Untuk memperbarui set replikasi

`command-name` Contoh berikut menghapus Wilayah us-east-2 dari set replikasi.

```
aws ssm-incidents update-replication-set \  
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/  
a2bcc5c9-0f53-8047-7fef-c20749989b40" \  
  --actions '[{"deleteRegionAction": {"regionName": "us-east-2"}}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan replikasi Manajer Insiden yang diatur](#) dalam Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateReplicationSet](#) di Referensi AWS CLI Perintah.

## update-response-plan

Contoh kode berikut menunjukkan cara menggunakan `update-response-plan`.

### AWS CLI

Untuk memperbarui rencana respons

`update-response-plan` Contoh berikut menghapus saluran obrolan dari rencana respons yang ditentukan.

```
aws ssm-incidents update-response-plan \  
  --arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan" \  
  \  
  --chat-channel '{"empty":{}}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Persiapan insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateResponsePlan](#) di Referensi AWS CLI Perintah.

## update-timeline-event

Contoh kode berikut menunjukkan cara menggunakan `update-timeline-event`.

### AWS CLI

Untuk memperbarui acara timeline

`update-timeline-event` Contoh berikut memperbarui waktu peristiwa itu terjadi.

```
aws ssm-incidents update-timeline-event \  
  --event-id 20bcc812-8a94-4cd7-520c-0ff742111424 \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \  
  --event-time "2021-05-21T18:10:57+00:00"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Detail insiden](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateTimelineEvent](#) di Referensi AWS CLI Perintah.

## Contoh Kontak Manajer Insiden menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Kontak Manajer Insiden AWS Command Line Interface dengan.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **accept-page**

Contoh kode berikut menunjukkan cara menggunakan `accept-page`.

AWS CLI

Untuk menerima halaman selama dan keterlibatan

`accept-page` Contoh berikut menggunakan kode terima yang dikirim ke saluran kontak untuk menerima halaman.

```
aws ssm-contacts accept-page \  
  --page-id "arn:aws:ssm-contacts:us-east-2:682428703967:page/  
akuam/94ea0c7b-56d9-46c3-b84a-a37c8b067ad3" \  
  --accept-type READ \  
  --accept-code 425440
```

Perintah ini tidak menghasilkan output

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [AcceptPage](#) di Referensi AWS CLI Perintah.

## activate-contact-channel

Contoh kode berikut menunjukkan cara menggunakan `activate-contact-channel`.

### AWS CLI

Aktifkan saluran kontak kontak

`activate-contact-channel` Contoh berikut mengaktifkan saluran kontak dan membuatnya dapat digunakan sebagai bagian dari insiden.

```
aws ssm-contacts activate-contact-channel \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-\  
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d" \  
  --activation-code "466136"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ActivateContactChannel](#) di Referensi AWS CLI Perintah.

## command-name

Contoh kode berikut menunjukkan cara menggunakan `command-name`.

### AWS CLI

Untuk menghapus kontak

`command-name` Contoh berikut menghapus kontak. Kontak tidak lagi dapat dijangkau dari rencana eskalasi apa pun yang merujuk pada mereka.

```
aws ssm-contacts delete-contact \  
  --contact-id "arn:aws:ssm-contacts:us-east-1:682428703967:contact/alejr"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [CommandName](#) di Referensi AWS CLI Perintah.



## create-contact-channel

Contoh kode berikut menunjukkan cara menggunakan `create-contact-channel`.

### AWS CLI

Untuk membuat saluran kontak

Membuat saluran kontak jenis SMS untuk kontak Akua Mansa. Saluran kontak dapat dibuat dari jenis SMS, EMAIL, atau VOICE.

```
aws ssm-contacts create-contact-channel \  
  --contact-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \  
  --name "akuas sms-test" \  
  --type SMS \  
  --delivery-address '{"SimpleAddress": "+15005550199"}'
```

Output:

```
{  
  "ContactChannelArn": "arn:aws:ssm-contacts:us-east-1:111122223333:contact-  
channel/akuam/02f506b9-ea5d-4764-af89-2daa793ff024"  
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [CreateContactChannel](#) di Referensi AWS CLI Perintah.

## create-contact

Contoh kode berikut menunjukkan cara menggunakan `create-contact`.

### AWS CLI

Untuk membuat kontak

`create-contact` Contoh berikut membuat kontak di lingkungan Anda dengan rencana kosong. Paket dapat diperbarui setelah membuat saluran kontak. Gunakan `create-contact-channel` perintah dengan output ARN dari perintah ini. Setelah Anda membuat saluran kontak untuk kontak ini, gunakan `update-contact` untuk memperbarui paket.

```
aws ssm-contacts create-contact \  

```

```
--alias "akuam" \  
--display-name "Akua Mansa" \  
--type PERSONAL \  
--plan '{"Stages": []}'
```

Output:

```
{  
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"  
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [CreateContact](#) di Referensi AWS CLI Perintah.

## deactivate-contact-channel

Contoh kode berikut menunjukkan cara menggunakan `deactivate-contact-channel`.

### AWS CLI

Untuk menonaktifkan saluran kontak

`deactivate-contact-channel` Contoh berikut menonaktifkan saluran kontak. Menonaktifkan saluran kontak berarti saluran kontak tidak akan lagi dipaging selama insiden terjadi. Anda juga dapat mengaktifkan kembali saluran kontak kapan saja menggunakan `activate-contact-channel` perintah.

```
aws ssm-contacts deactivate-contact-channel \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-  
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeactivateContactChannel](#) di Referensi AWS CLI Perintah.

## delete-contact-channel

Contoh kode berikut menunjukkan cara menggunakan `delete-contact-channel`.

## AWS CLI

Untuk menghapus saluran kontak

`delete-contact-channel` Contoh berikut menghapus saluran kontak. Menghapus saluran kontak memastikan saluran kontak tidak akan diberi halaman selama insiden terjadi.

```
aws ssm-contacts delete-contact-channel \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact-  
channel/akuam/13149bad-52ee-45ea-ae1e-45857f78f9b2"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteContactChannel](#) di Referensi AWS CLI Perintah.

## `delete-contact`

Contoh kode berikut menunjukkan cara menggunakan `delete-contact`.

## AWS CLI

Untuk menghapus kontak

`delete-contact` Contoh berikut menghapus kontak. Kontak tidak lagi dapat dijangkau dari rencana eskalasi apa pun yang merujuk pada mereka.

```
aws ssm-contacts delete-contact \  
  --contact-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact/alejrr"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DeleteContact](#) di Referensi AWS CLI Perintah.

## `describe-engagement`

Contoh kode berikut menunjukkan cara menggunakan `describe-engagement`.

## AWS CLI

Untuk menjelaskan detail pertunangan

`describe-engagement` Contoh berikut mencantumkan rincian keterlibatan ke kontak atau rencana eskalasi. Subjek dan konten dikirim ke saluran kontak.

```
aws ssm-contacts describe-engagement \
  --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
  example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

Output:

```
{
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
  example_escalation",
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
  example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356",
  "Sender": "cli",
  "Subject": "cli-test",
  "Content": "Testing engagements via CLI",
  "PublicSubject": "cli-test",
  "PublicContent": "Testing engagements va CLI",
  "StartTime": "2021-05-18T18:25:41.151000+00:00"
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DescribeEngagement](#) di Referensi AWS CLI Perintah.

## `describe-page`

Contoh kode berikut menunjukkan cara menggunakan `describe-page`.

## AWS CLI

Untuk mencantumkan detail halaman ke saluran kontak

`describe-page` Contoh berikut mencantumkan detail halaman ke saluran kontak. Halaman akan mencakup subjek dan konten yang disediakan.

```
aws ssm-contacts describe-page \
```

```
--page-id "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-e606-498a-861b-25726292eb93"
```

### Output:

```
{
  "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-e606-498a-861b-25726292eb93",
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
  "Sender": "cli",
  "Subject": "cli-test",
  "Content": "Testing engagements via CLI",
  "PublicSubject": "cli-test",
  "PublicContent": "Testing engagements va CLI",
  "SentTime": "2021-05-18T18:43:29.301000+00:00",
  "ReadTime": "2021-05-18T18:43:55.708000+00:00",
  "DeliveryTime": "2021-05-18T18:43:55.265000+00:00"
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [DescribePage](#) di Referensi AWS CLI Perintah.

## get-contact-channel

Contoh kode berikut menunjukkan cara menggunakan `get-contact-channel`.

### AWS CLI

Untuk mencantumkan rincian saluran kontak

`get-contact-channel` Contoh berikut mencantumkan rincian saluran kontak.

```
aws ssm-contacts get-contact-channel \
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d"
```

### Output:

```
{
```

```

    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
    "ContactChannelArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
    "Name": "akuas sms",
    "Type": "SMS",
    "DeliveryAddress": {
      "SimpleAddress": "+15005550199"
    },
    "ActivationStatus": "ACTIVATED"
  }

```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetContactChannel](#) di Referensi AWS CLI Perintah.

## get-contact-policy

Contoh kode berikut menunjukkan cara menggunakan `get-contact-policy`.

### AWS CLI

Untuk membuat daftar kebijakan sumber daya kontak

`get-contact-policy` Contoh berikut mencantumkan kebijakan sumber daya yang terkait dengan kontak yang ditentukan.

```

aws ssm-contacts get-contact-policy \
  --contact-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam"

```

Output:

```

{
  "ContactArn": "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam",
  "Policy": "{\n\"Version\": \"2012-10-17\", \"Statement\": [\n{\n\"Sid\":\n\n\"SharePolicyForDocumentationDrAlia\", \"Effect\": \"Allow\", \"Principal\":\n\n{\n\"AWS\": \"222233334444\"}, \"Action\": [\n\"ssm-contacts:GetContact\", \"ssm-
contacts:StartEngagement\", \"ssm-contacts:DescribeEngagement\", \"ssm-
contacts:ListPagesByEngagement\", \"ssm-contacts:StopEngagement\"], \"Resource
\": [\n\"arn:aws:ssm-contacts:*:111122223333:contact/akuam\", \"arn:aws:ssm-
contacts:*:111122223333:engagement/akuam/*\"]\n}]]}"
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan kontak bersama dan rencana respons](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetContactPolicy](#) di Referensi AWS CLI Perintah.

## get-contact

Contoh kode berikut menunjukkan cara menggunakan `get-contact`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan rencana kontak

`get-contact` Contoh berikut menjelaskan kontak.

```
aws ssm-contacts get-contact \
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Output:

```
{
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
  "Alias": "akuam",
  "DisplayName": "Akua Mansa",
  "Type": "PERSONAL",
  "Plan": {
    "Stages": [
      {
        "DurationInMinutes": 5,
        "Targets": [
          {
            "ChannelTargetInfo": {
              "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/beb25840-5ac8-4644-95cc-7a8de390fa65",
              "RetryIntervalInMinutes": 1
            }
          }
        ]
      },
      {
        "DurationInMinutes": 5,
        "Targets": [
```

```

        {
            "ChannelTargetInfo": {
                "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad",
                "RetryIntervalInMinutes": 1
            }
        }
    ],
    {
        "DurationInMinutes": 5,
        "Targets": [
            {
                "ChannelTargetInfo": {
                    "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/77d4f447-f619-4954-afff-85551e369c2a",
                    "RetryIntervalInMinutes": 1
                }
            }
        ]
    }
]
}
}

```

Contoh 2: Untuk menggambarkan rencana eskalasi

get-contact Contoh berikut menjelaskan rencana eskalasi.

```

aws ssm-contacts get-contact \
--contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation"

```

Output:

```

{
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
    "Alias": "example_escalation",
    "DisplayName": "Example Escalation",
    "Type": "ESCALATION",
    "Plan": {
        "Stages": [

```



```
{
  "DurationInMinutes": 5,
  "Targets": [
    {
      "ContactTargetInfo": {
        "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/akuam",
        "IsEssential": true
      }
    }
  ],
},
{
  "DurationInMinutes": 5,
  "Targets": [
    {
      "ContactTargetInfo": {
        "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/alejr",
        "IsEssential": false
      }
    }
  ],
},
{
  "DurationInMinutes": 0,
  "Targets": [
    {
      "ContactTargetInfo": {
        "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/anasi",
        "IsEssential": false
      }
    }
  ]
}
]
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [GetContact](#) di Referensi AWS CLI Perintah.

## list-contact-channels

Contoh kode berikut menunjukkan cara menggunakan `list-contact-channels`.

### AWS CLI

Untuk membuat daftar saluran kontak kontak kontak

`list-contact-channels` Contoh berikut mencantumkan saluran kontak yang tersedia dari kontak yang ditentukan.

```
aws ssm-contacts list-contact-channels \
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

### Output:

```
{
  [
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Name": "akuas email",
      "Type": "EMAIL",
      "DeliveryAddress": {
        "SimpleAddress": "akuam@example.com"
      },
      "ActivationStatus": "NOT_ACTIVATED"
    },
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Name": "akuas sms",
      "Type": "SMS",
      "DeliveryAddress": {
        "SimpleAddress": "+15005550100"
      },
      "ActivationStatus": "ACTIVATED"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListContactChannels](#) di Referensi AWS CLI Perintah.

## list-contacts

Contoh kode berikut menunjukkan cara menggunakan `list-contacts`.

### AWS CLI

Untuk membuat daftar semua rencana eskalasi dan kontak

`list-contacts` Contoh berikut mencantumkan kontak dan rencana eskalasi di akun Anda.

```
aws ssm-contacts list-contacts
```

Output:

```
{
  "Contacts": [
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Alias": "akuam",
      "DisplayName": "Akua Mansa",
      "Type": "PERSONAL"
    },
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
alejr",
      "Alias": "alejr",
      "DisplayName": "Alejandro Rosalez",
      "Type": "PERSONAL"
    },
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
anasi",
      "Alias": "anasi",
      "DisplayName": "Ana Carolina Silva",
      "Type": "PERSONAL"
    },
    {
```

```

        "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
        "Alias": "example_escalation",
        "DisplayName": "Example Escalation",
        "Type": "ESCALATION"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListContacts](#) di Referensi AWS CLI Perintah.

## list-engagements

Contoh kode berikut menunjukkan cara menggunakan `list-engagements`.

### AWS CLI

Untuk membuat daftar semua keterlibatan

`list-engagements` Contoh berikut mencantumkan keterlibatan untuk rencana eskalasi dan kontak. Anda juga dapat membuat daftar keterlibatan untuk satu insiden.

```
aws ssm-contacts list-engagements
```

Output:

```

{
  "Engagements": [
    {
      "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/91792571-0b53-4821-9f73-d25d13d9e529",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Sender": "cli",
      "StartTime": "2021-05-18T20:37:50.300000+00:00"
    },
    {
      "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",

```

```

    "Sender": "cli",
    "StartTime": "2021-05-18T18:40:26.666000+00:00"
  },
  {
    "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356",
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
    "Sender": "cli",
    "StartTime": "2021-05-18T18:25:41.151000+00:00"
  },
  {
    "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/607ced0e-e8fa-4ea7-8958-a237b8803f8f",
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
    "Sender": "cli",
    "StartTime": "2021-05-18T18:20:58.093000+00:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListEngagements](#) di Referensi AWS CLI Perintah.

## list-page-receipts

Contoh kode berikut menunjukkan cara menggunakan `list-page-receipts`.

### AWS CLI

Untuk membuat daftar tanda terima halaman

`command-name` Contoh berikut mencantumkan apakah halaman diterima atau tidak oleh kontak.

```

aws ssm-contacts list-page-receipts \
  --page-id "arn:aws:ssm-contacts:us-east-2:111122223333:page/
akuam/94ea0c7b-56d9-46c3-b84a-a37c8b067ad3"

```

Output:

```
{
  "Receipts": [
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ReceiptType": "DELIVERED",
      "ReceiptInfo": "425440",
      "ReceiptTime": "2021-05-18T20:42:57.485000+00:00"
    },
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ReceiptType": "READ",
      "ReceiptInfo": "425440",
      "ReceiptTime": "2021-05-18T20:42:57.907000+00:00"
    },
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ReceiptType": "SENT",
      "ReceiptInfo": "SM6656c19132f1465f9c9c1123a5dde7c9",
      "ReceiptTime": "2021-05-18T20:40:52.962000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListPageReceipts](#) di Referensi AWS CLI Perintah.

## list-pages-by-contact

Contoh kode berikut menunjukkan cara menggunakan `list-pages-by-contact`.

### AWS CLI

Untuk daftar halaman dengan kontak

`list-pages-by-contact` Contoh berikut mencantumkan semua halaman ke kontak yang ditentukan.

```
aws ssm-contacts list-pages-by-contact \
```

```
--contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Output:

```
{
  "Pages": [
    {
      "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/
ad0052bd-e606-498a-861b-25726292eb93",
      "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Sender": "cli",
      "SentTime": "2021-05-18T18:43:29.301000+00:00",
      "DeliveryTime": "2021-05-18T18:43:55.265000+00:00",
      "ReadTime": "2021-05-18T18:43:55.708000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListPagesByContact](#) di Referensi AWS CLI Perintah.

## list-pages-by-engagement

Contoh kode berikut menunjukkan cara menggunakan `list-pages-by-engagement`.

AWS CLI

Untuk daftar halaman untuk menghubungi saluran dimulai dari keterlibatan.

`list-pages-by-engagement` Contoh berikut mencantumkan halaman yang terjadi saat melibatkan rencana keterlibatan yang ditentukan.

```
aws ssm-contacts list-pages-by-engagement \
  --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
akuam/78a29753-3674-4ac5-9f83-0468563567f0"
```

Output:

```
{
  "Pages": [
    {
      "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-e606-498a-861b-25726292eb93",
      "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
      "Sender": "cli",
      "SentTime": "2021-05-18T18:40:27.245000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListPagesByEngagement](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag untuk kontak

`list-tags-for-resource` Contoh berikut mencantumkan tag kontak yang ditentukan.

```
aws ssm-contacts list-tags-for-resource \
  --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam"
```

Output:

```
{
  "Tags": [
    {
      "Key": "group1",
      "Value": "1"
    }
  ]
}
```



Untuk informasi selengkapnya, lihat [Menandai](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-contact-policy

Contoh kode berikut menunjukkan cara menggunakan `put-contact-policy`.

### AWS CLI

Untuk berbagi kontak dan keterlibatan

`put-contact-policy` Contoh berikut menambahkan kebijakan sumber daya ke kontak Akua yang berbagi kontak dan keterlibatan terkait dengan kepala sekolah.

```
aws ssm-contacts put-contact-policy \
  --contact-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \
  --policy "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
  \\\"ExampleResourcePolicy\\\",\\\"Action\":[\\\"ssm-contacts:GetContact\\\",\\\"ssm-
  contacts:StartEngagement\\\",\\\"ssm-contacts:DescribeEngagement\\\",\\\"ssm-
  contacts:ListPagesByEngagement\\\",\\\"ssm-contacts:StopEngagement\\\"],
  \\\"Principal\":[\\\"AWS\\\":\\\"222233334444\\\"],\\\"Effect\\\":\\\"Allow\\\",\\\"Resource
  \":[\\\"arn:aws:ssm-contacts:*:111122223333:contact/akuam\\\",\\\"arn:aws:ssm-
  contacts:*:111122223333:engagement/akuam/*\\\"]}]}"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan kontak bersama dan rencana respons](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [PutContactPolicy](#) di Referensi AWS CLI Perintah.

## send-activation-code

Contoh kode berikut menunjukkan cara menggunakan `send-activation-code`.

### AWS CLI

Untuk mengirim kode aktivasi

`send-activation-code` Contoh berikut mengirimkan kode aktivasi dan pesan ke saluran kontak yang ditentukan.

```
aws ssm-contacts send-activation-code \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact-  
channel/akuam/8ddae2d1-12c8-4e45-b852-c8587266c400"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [SendActivationCode](#) di Referensi AWS CLI Perintah.

## start-engagement

Contoh kode berikut menunjukkan cara menggunakan `start-engagement`.

### AWS CLI

Contoh 1: Untuk halaman saluran kontak kontak

`start-engagement` Halaman-halaman berikut menghubungi saluran kontak. Pengirim, subjek, subjek publik, dan konten publik semuanya bebas dari bidang. Manajer Insiden mengirimkan subjek dan konten ke saluran kontak VOICE atau EMAIL yang disediakan. Manajer Insiden mengirimkan subjek publik dan konten publik ke saluran kontak SMS yang disediakan. Pengirim digunakan untuk melacak siapa yang memulai keterlibatan.

```
aws ssm-contacts start-engagement \  
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam" \  
  --sender "cli" \  
  --subject "cli-test" \  
  --content "Testing engagements via CLI" \  
  --public-subject "cli-test" \  
  --public-content "Testing engagements va CLI"
```

Output:

```
{  
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/  
akuam/607ced0e-e8fa-4ea7-8958-a237b8803f8f"  
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

Contoh 2: Untuk halaman kontak dalam rencana eskalasi yang disediakan.

Berikut ini `start-engagement` melibatkan kontak melalui rencana eskalasi. Setiap kontak diberi halaman sesuai dengan rencana pertunangan mereka.

```
aws ssm-contacts start-engagement \  
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/  
example_escalation" \  
  --sender "cli" \  
  --subject "cli-test" \  
  --content "Testing engagements via CLI" \  
  --public-subject "cli-test" \  
  --public-content "Testing engagements va CLI"
```

Output:

```
{  
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/  
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"  
}
```

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [StartEngagement](#) di Referensi AWS CLI Perintah.

## stop-engagement

Contoh kode berikut menunjukkan cara menggunakan `stop-engagement`.

### AWS CLI

Untuk menghentikan pertunangan

`stop-engagement` Contoh berikut menghentikan keterlibatan dari paging kontak lebih lanjut dan saluran kontak.

```
aws ssm-contacts stop-engagement \  
  --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/  
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [StopEngagement](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai kontak

`tag-resource` Contoh berikut menandai kontak tertentu dengan pasangan nilai kunci tag yang disediakan.

```
aws ssm-contacts tag-resource \  
  --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \  
  --tags '[{"Key":"group1","Value":"1"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari kontak

`untag-resource` Contoh berikut menghapus tag `group1` dari kontak yang ditentukan.

```
aws ssm-contacts untag-resource \  
  --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \  
  --tag-keys "group1"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-contact-channel

Contoh kode berikut menunjukkan cara menggunakan `update-contact-channel`.

### AWS CLI

Untuk memperbarui saluran kontak

`update-contact-channel` Contoh berikut memperbarui nama dan alamat pengiriman saluran kontak.

```
aws ssm-contacts update-contact-channel \
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad" \
  --name "akuas voice channel" \
  --delivery-address '{"SimpleAddress": "+15005550198"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateContactChannel](#) di Referensi AWS CLI Perintah.

## update-contact

Contoh kode berikut menunjukkan cara menggunakan `update-contact`.

### AWS CLI

Untuk memperbarui rencana keterlibatan kontak

`update-contact` Contoh berikut memperbarui rencana keterlibatan kontak Akua untuk menyertakan tiga jenis saluran kontak. Ini dilakukan setelah membuat saluran kontak untuk Akua.

```
aws ssm-contacts update-contact \
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam" \
  --plan '{"Stages": [{"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo":
{"ContactChannelId": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/beb25840-5ac8-4644-95cc-7a8de390fa65", "RetryIntervalInMinutes":
1 }]}], {"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo":
{"ContactChannelId": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/
akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad", "RetryIntervalInMinutes": 1}]}],
{"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo": {"ContactChannelId":
```

```
"arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/77d4f447-f619-4954-afff-85551e369c2a", "RetryIntervalInMinutes": 1 }]]]]}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kontak](#) di Panduan Pengguna Manajer Insiden.

- Untuk detail API, lihat [UpdateContact](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Inspector menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon Inspector.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-attributes-to-findings**

Contoh kode berikut menunjukkan cara menggunakan `add-attributes-to-findings`.

AWS CLI

Untuk menambahkan atribut ke temuan

```
add-attribute-to-findingPerintah berikut memberikan atribut dengan kunci Example dan nilai example untuk temuan dengan arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU ARN dari:
```

```
aws inspector add-attributes-to-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU --attributes key=Example,value=example
```

Output:

```
{
  "failedItems": {}
}
```

Untuk informasi selengkapnya, lihat Temuan Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [AddAttributesToFindings](#) di Referensi AWS CLI Perintah.

## create-assessment-target

Contoh kode berikut menunjukkan cara menggunakan `create-assessment-target`.

AWS CLI

Untuk membuat target penilaian

`create-assessment-target` Perintah berikut membuat target penilaian bernama `ExampleAssessmentTarget` menggunakan grup sumber daya dengan ARN dari `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-AB6DMKnv`

```
aws inspector create-assessment-target --assessment-target-name
ExampleAssessmentTarget --resource-group-arn arn:aws:inspector:us-
west-2:123456789012:resourcegroup/0-AB6DMKnv
```

Output:

```
{
  "assessmentTargetArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX"
}
```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [CreateAssessmentTarget](#) di Referensi AWS CLI Perintah.

## create-assessment-template

Contoh kode berikut menunjukkan cara menggunakan `create-assessment-template`.

### AWS CLI

Untuk membuat template penilaian

`create-assessment-template` Perintah berikut membuat template penilaian yang dipanggil `ExampleAssessmentTemplate` untuk target penilaian dengan ARN: `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX`

```
aws inspector create-assessment-template --assessment-target-arn
arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX --assessment-template-
name ExampleAssessmentTemplate --duration-in-seconds 180 --rules-package-arns
arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p --user-attributes-
for-findings key=ExampleTag,value=examplevalue
```

Output:

```
{
  "assessmentTemplateArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-it5r2S4T"
}
```

Untuk informasi selengkapnya, lihat [Amazon Inspector Assessment Templates dan Assessment Runs](#) di panduan Amazon Inspector.

- Untuk detail API, lihat [CreateAssessmentTemplate](#) di Referensi AWS CLI Perintah.

## create-resource-group

Contoh kode berikut menunjukkan cara menggunakan `create-resource-group`.

### AWS CLI

Untuk membuat grup sumber daya

`create-resource-group` Perintah berikut membuat grup sumber daya menggunakan kunci tag `Name` dan nilai `example`:



```
aws inspector create-resource-group --resource-group-tags key=Name,value=example
```

Output:

```
{
  "resourceGroupArn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-AB6DMKnv"
}
```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [CreateResourceGroup](#) di Referensi AWS CLI Perintah.

## **delete-assessment-run**

Contoh kode berikut menunjukkan cara menggunakan `delete-assessment-run`.

### AWS CLI

Untuk menghapus proses penilaian

`delete-assessment-run` Perintah berikut menghapus penilaian yang dijalankan dengan `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-11LMTAVe` ARN dari:

```
aws inspector delete-assessment-run --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-11LMTAVe
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [DeleteAssessmentRun](#) di Referensi AWS CLI Perintah.

## **delete-assessment-target**

Contoh kode berikut menunjukkan cara menggunakan `delete-assessment-target`.

### AWS CLI

Untuk menghapus target penilaian

`delete-assessment-target` Perintah berikut menghapus target penilaian dengan `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq` ARN dari:

```
aws inspector delete-assessment-target --assessment-target-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [DeleteAssessmentTarget](#) di Referensi AWS CLI Perintah.

## **delete-assessment-template**

Contoh kode berikut menunjukkan cara menggunakan `delete-assessment-template`.

### AWS CLI

Untuk menghapus template penilaian

`delete-assessment-template` Perintah berikut menghapus template penilaian dengan `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T` ARN dari:

```
aws inspector delete-assessment-template --assessment-template-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [DeleteAssessmentTemplate](#) di Referensi AWS CLI Perintah.

## **describe-assessment-runs**

Contoh kode berikut menunjukkan cara menggunakan `describe-assessment-runs`.

### AWS CLI

Untuk menggambarkan penilaian berjalan

`describe-assessment-run` Perintah berikut menjelaskan penilaian yang dijalankan dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE`

```
aws inspector describe-assessment-runs --assessment-run-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

## Output:

```
{
  "assessmentRuns": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
      "assessmentTemplateArn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw",
      "completedAt": 1458680301.4,
      "createdAt": 1458680170.035,
      "dataCollected": true,
      "durationInSeconds": 3600,
      "name": "Run 1 for ExampleAssessmentTemplate",
      "notifications": [],
      "rulesPackageArns": [
        "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP"
      ],
      "startedAt": 1458680170.161,
      "state": "COMPLETED",
      "stateChangedAt": 1458680301.4,
      "stateChanges": [
        {
          "state": "CREATED",
          "stateChangedAt": 1458680170.035
        },
        {
          "state": "START_DATA_COLLECTION_PENDING",
          "stateChangedAt": 1458680170.065
        },
        {
          "state": "START_DATA_COLLECTION_IN_PROGRESS",
          "stateChangedAt": 1458680170.096
        },
        {
          "state": "COLLECTING_DATA",
          "stateChangedAt": 1458680170.161
        },
        {
          "state": "STOP_DATA_COLLECTION_PENDING",

```

```

        "stateChangedAt": 1458680239.883
      },
      {
        "state": "DATA_COLLECTED",
        "stateChangedAt": 1458680299.847
      },
      {
        "state": "EVALUATING_RULES",
        "stateChangedAt": 1458680300.099
      },
      {
        "state": "COMPLETED",
        "stateChangedAt": 1458680301.4
      }
    ],
    "userAttributesForFindings": []
  }
],
"failedItems": {}
}

```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeAssessmentRuns](#) di Referensi AWS CLI Perintah.

## describe-assessment-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-assessment-targets`.

### AWS CLI

Untuk menggambarkan target penilaian

`describe-assessment-targets` Perintah berikut menjelaskan target penilaian dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`

```
aws inspector describe-assessment-targets --assessment-target-arns
arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Output:

```
{
```

```

    "assessmentTargets": [
      {
        "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq",
        "createdAt": 1458074191.459,
        "name": "ExampleAssessmentTarget",
        "resourceGroupArn": "arn:aws:inspector:us-
west-2:123456789012:resourcegroup/0-PyGXopAI",
        "updatedAt": 1458074191.459
      }
    ],
    "failedItems": {}
  }

```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeAssessmentTargets](#) di Referensi AWS CLI Perintah.

## describe-assessment-templates

Contoh kode berikut menunjukkan cara menggunakan `describe-assessment-templates`.

### AWS CLI

Untuk menggambarkan templat penilaian

`describe-assessment-templates` Perintah berikut menjelaskan template penilaian dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw`

```

aws inspector describe-assessment-templates --assessment-template-arns
arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw

```

Output:

```

{
  "assessmentTemplates": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw",
      "assessmentTargetArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq",

```

```
        "createdAt": 1458074191.844,  
        "durationInSeconds": 3600,  
        "name": "ExampleAssessmentTemplate",  
        "rulesPackageArns": [  
            "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP"  
        ],  
        "userAttributesForFindings": []  
    }  
],  
"failedItems": {}  
}
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeAssessmentTemplates](#) di Referensi AWS CLI Perintah.

## describe-cross-account-access-role

Contoh kode berikut menunjukkan cara menggunakan `describe-cross-account-access-role`.

### AWS CLI

Untuk mendeskripsikan peran akses lintas akun

`describe-cross-account-access-role` Perintah berikut menjelaskan peran IAM yang memungkinkan Amazon Inspector mengakses AWS akun Anda:

```
aws inspector describe-cross-account-access-role
```

Output:

```
{  
    "registeredAt": 1458069182.826,  
    "roleArn": "arn:aws:iam::123456789012:role/inspector",  
    "valid": true  
}
```

Untuk informasi selengkapnya, lihat Menyiapkan Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeCrossAccountAccessRole](#) di Referensi AWS CLI Perintah.

## describe-findings

Contoh kode berikut menunjukkan cara menggunakan `describe-findings`.

### AWS CLI

Untuk menggambarkan temuan

`describe-findings` Perintah berikut menjelaskan temuan dengan ARN dari:  
`arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4`

```
aws inspector describe-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4
```

Output:

```
{
  "failedItems": {},
  "findings": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4",
      "assetAttributes": {
        "ipv4Addresses": [],
        "schemaVersion": 1
      },
      "assetType": "ec2-instance",
      "attributes": [],
      "confidence": 10,
      "createdAt": 1458680301.37,
      "description": "Amazon Inspector did not find any potential security issues during this assessment.",
      "indicatorOfCompromise": false,
      "numericSeverity": 0,
      "recommendation": "No remediation needed.",
      "schemaVersion": 1,
      "service": "Inspector",
      "serviceAttributes": {
        "assessmentRunArn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",

```

```

        "rulesPackageArn": "arn:aws:inspector:us-
west-2:758058086616:rulespackage/0-X1KXtawP",
        "schemaVersion": 1
    },
    "severity": "Informational",
    "title": "No potential security issues found",
    "updatedAt": 1458680301.37,
    "userAttributes": []
}
]
}

```

Untuk informasi selengkapnya, lihat Temuan Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeFindings](#) di Referensi AWS CLI Perintah.

## describe-resource-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-resource-groups`.

### AWS CLI

Untuk menggambarkan kelompok sumber daya

`describe-resource-groups` Perintah berikut menjelaskan grup sumber daya dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-PyGXopAI`

```
aws inspector describe-resource-groups --resource-group-arns arn:aws:inspector:us-
west-2:123456789012:resourcegroup/0-PyGXopAI
```

Output:

```

{
  "failedItems": {},
  "resourceGroups": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-
PyGXopAI",
      "createdAt": 1458074191.098,
      "tags": [
        {
          "key": "Name",
          "value": "example"
        }
      ]
    }
  ]
}

```



```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeResourceGroups](#) di Referensi AWS CLI Perintah.

## describe-rules-packages

Contoh kode berikut menunjukkan cara menggunakan `describe-rules-packages`.

### AWS CLI

Untuk menggambarkan paket aturan

`describe-rules-packages` Perintah berikut menjelaskan paket aturan dengan ARN dari: `arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p`

```
aws inspector describe-rules-packages --rules-package-arns arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p
```

### Output:

```
{
  "failedItems": {},
  "rulesPackages": [
    {
      "arn": "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p",
      "description": "The rules in this package help verify whether the EC2 instances in your application are exposed to Common Vulnerabilities and Exposures (CVEs). Attacks can exploit unpatched vulnerabilities to compromise the confidentiality, integrity, or availability of your service or data. The CVE system provides a reference for publicly known information security vulnerabilities and exposures. For more information, see [https://cve.mitre.org/](https://cve.mitre.org/). If a particular CVE appears in one of the produced Findings at the end of a completed Inspector assessment, you can search [https://cve.mitre.org/](https://cve.mitre.org/) using the CVE's ID (for example, \"CVE-2009-0021\") to
```

```

        find detailed information about this CVE, its severity, and how to
        mitigate it. ",
        "name": "Common Vulnerabilities and Exposures",
        "provider": "Amazon Web Services, Inc.",
        "version": "1.1"
    }
]
}

```

Untuk informasi selengkapnya, lihat Paket dan Aturan Amazon Inspector dalam panduan Amazon Inspector.

- Untuk detail API, lihat [DescribeRulesPackages](#) di Referensi AWS CLI Perintah.

## get-telemetry-metadata

Contoh kode berikut menunjukkan cara menggunakan `get-telemetry-metadata`.

### AWS CLI

Untuk mendapatkan metadata telemetri

`get-telemetry-metadata` Perintah berikut menghasilkan informasi tentang data yang dikumpulkan untuk penilaian yang dijalankan dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE`

```
aws inspector get-telemetry-metadata --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

Output:

```

{
  "telemetryMetadata": [
    {
      "count": 2,
      "dataSize": 345,
      "messageType": "InspectorDuplicateProcess"
    },
    {
      "count": 3,
      "dataSize": 255,
      "messageType": "InspectorTimeEventMsg"
    }
  ],
}

```

```
{
  "count": 4,
  "dataSize": 1082,
  "messageType": "InspectorNetworkInterface"
},
{
  "count": 2,
  "dataSize": 349,
  "messageType": "InspectorDnsEntry"
},
{
  "count": 11,
  "dataSize": 2514,
  "messageType": "InspectorDirectoryInfoMsg"
},
{
  "count": 1,
  "dataSize": 179,
  "messageType": "InspectorTcpV6ListeningPort"
},
{
  "count": 101,
  "dataSize": 10949,
  "messageType": "InspectorTerminal"
},
{
  "count": 26,
  "dataSize": 5916,
  "messageType": "InspectorUser"
},
{
  "count": 282,
  "dataSize": 32148,
  "messageType": "InspectorDynamicallyLoadedCodeModule"
},
{
  "count": 18,
  "dataSize": 10172,
  "messageType": "InspectorCreateProcess"
},
{
  "count": 3,
  "dataSize": 8001,
  "messageType": "InspectorProcessPerformance"
}
```

```
  },
  {
    "count": 1,
    "dataSize": 360,
    "messageType": "InspectorOperatingSystem"
  },
  {
    "count": 6,
    "dataSize": 546,
    "messageType": "InspectorStopProcess"
  },
  {
    "count": 1,
    "dataSize": 1553,
    "messageType": "InspectorInstanceMetaData"
  },
  {
    "count": 2,
    "dataSize": 434,
    "messageType": "InspectorTcpV4Connection"
  },
  {
    "count": 474,
    "dataSize": 2960322,
    "messageType": "InspectorPackageInfo"
  },
  {
    "count": 3,
    "dataSize": 2235,
    "messageType": "InspectorSystemPerformance"
  },
  {
    "count": 105,
    "dataSize": 46048,
    "messageType": "InspectorCodeModule"
  },
  {
    "count": 1,
    "dataSize": 182,
    "messageType": "InspectorUdpV6ListeningPort"
  },
  {
    "count": 2,
    "dataSize": 371,
```

```
    "messageType": "InspectorUdpV4ListeningPort"
  },
  {
    "count": 18,
    "dataSize": 8362,
    "messageType": "InspectorKernelModule"
  },
  {
    "count": 29,
    "dataSize": 48788,
    "messageType": "InspectorConfigurationInfo"
  },
  {
    "count": 1,
    "dataSize": 79,
    "messageType": "InspectorMonitoringStart"
  },
  {
    "count": 5,
    "dataSize": 0,
    "messageType": "InspectorSplitMsgBegin"
  },
  {
    "count": 51,
    "dataSize": 4593,
    "messageType": "InspectorGroup"
  },
  {
    "count": 1,
    "dataSize": 184,
    "messageType": "InspectorTcpV4ListeningPort"
  },
  {
    "count": 1159,
    "dataSize": 3146579,
    "messageType": "Total"
  },
  {
    "count": 5,
    "dataSize": 0,
    "messageType": "InspectorSplitMsgEnd"
  },
  {
    "count": 1,
```

```

        "dataSize": 612,
        "messageType": "InspectorLoadImageInProgress"
      }
    ]
  }

```

- Untuk detail API, lihat [GetTelemetryMetadata](#) di Referensi AWS CLI Perintah.

## list-assessment-run-agents

Contoh kode berikut menunjukkan cara menggunakan `list-assessment-run-agents`.

### AWS CLI

Untuk membuat daftar agen yang dijalankan penilaian

`list-assessment-run-agents` Perintah berikut mencantumkan agen penilaian yang dijalankan dengan ARN yang ditentukan.

```

aws inspector list-assessment-run-agents \
  --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
  template/0-4r1V2mAw/run/0-MKkpXXPE

```

### Output:

```

{
  "assessmentRunAgents": [
    {
      "agentHealth": "HEALTHY",
      "agentHealthCode": "HEALTHY",
      "agentId": "i-49113b93",
      "assessmentRunArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
      "telemetryMetadata": [
        {
          "count": 2,
          "dataSize": 345,
          "messageType": "InspectorDuplicateProcess"
        },
        {
          "count": 3,
          "dataSize": 255,

```

```
    "messageType": "InspectorTimeEventMsg"
  },
  {
    "count": 4,
    "dataSize": 1082,
    "messageType": "InspectorNetworkInterface"
  },
  {
    "count": 2,
    "dataSize": 349,
    "messageType": "InspectorDnsEntry"
  },
  {
    "count": 11,
    "dataSize": 2514,
    "messageType": "InspectorDirectoryInfoMsg"
  },
  {
    "count": 1,
    "dataSize": 179,
    "messageType": "InspectorTcpV6ListeningPort"
  },
  {
    "count": 101,
    "dataSize": 10949,
    "messageType": "InspectorTerminal"
  },
  {
    "count": 26,
    "dataSize": 5916,
    "messageType": "InspectorUser"
  },
  {
    "count": 282,
    "dataSize": 32148,
    "messageType": "InspectorDynamicallyLoadedCodeModule"
  },
  {
    "count": 18,
    "dataSize": 10172,
    "messageType": "InspectorCreateProcess"
  },
  {
    "count": 3,
```

```
        "dataSize": 8001,  
        "messageType": "InspectorProcessPerformance"  
    },  
    {  
        "count": 1,  
        "dataSize": 360,  
        "messageType": "InspectorOperatingSystem"  
    },  
    {  
        "count": 6,  
        "dataSize": 546,  
        "messageType": "InspectorStopProcess"  
    },  
    {  
        "count": 1,  
        "dataSize": 1553,  
        "messageType": "InspectorInstanceMetaData"  
    },  
    {  
        "count": 2,  
        "dataSize": 434,  
        "messageType": "InspectorTcpV4Connection"  
    },  
    {  
        "count": 474,  
        "dataSize": 2960322,  
        "messageType": "InspectorPackageInfo"  
    },  
    {  
        "count": 3,  
        "dataSize": 2235,  
        "messageType": "InspectorSystemPerformance"  
    },  
    {  
        "count": 105,  
        "dataSize": 46048,  
        "messageType": "InspectorCodeModule"  
    },  
    {  
        "count": 1,  
        "dataSize": 182,  
        "messageType": "InspectorUdpV6ListeningPort"  
    },  
    {
```



```
        "count": 2,  
        "dataSize": 371,  
        "messageType": "InspectorUdpV4ListeningPort"  
    },  
    {  
        "count": 18,  
        "dataSize": 8362,  
        "messageType": "InspectorKernelModule"  
    },  
    {  
        "count": 29,  
        "dataSize": 48788,  
        "messageType": "InspectorConfigurationInfo"  
    },  
    {  
        "count": 1,  
        "dataSize": 79,  
        "messageType": "InspectorMonitoringStart"  
    },  
    {  
        "count": 5,  
        "dataSize": 0,  
        "messageType": "InspectorSplitMsgBegin"  
    },  
    {  
        "count": 51,  
        "dataSize": 4593,  
        "messageType": "InspectorGroup"  
    },  
    {  
        "count": 1,  
        "dataSize": 184,  
        "messageType": "InspectorTcpV4ListeningPort"  
    },  
    {  
        "count": 1159,  
        "dataSize": 3146579,  
        "messageType": "Total"  
    },  
    {  
        "count": 5,  
        "dataSize": 0,  
        "messageType": "InspectorSplitMsgEnd"  
    },  
    },
```

```
    {
      "count": 1,
      "dataSize": 612,
      "messageType": "InspectorLoadImageInProgress"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [AWS Agen](#) di Panduan Pengguna Amazon Inspector.

- Untuk detail API, lihat [ListAssessmentRunAgents](#) di Referensi AWS CLI Perintah.

## list-assessment-runs

Contoh kode berikut menunjukkan cara menggunakan `list-assessment-runs`.

### AWS CLI

Untuk membuat daftar penilaian berjalan

`list-assessment-runs` Perintah berikut mencantumkan semua proses penilaian yang ada.

```
aws inspector list-assessment-runs
```

Output:

```
{
  "assessmentRunArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE",
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-v5D6fI3v"
  ]
}
```

Untuk informasi selengkapnya, lihat [Amazon Inspector Assessment Templates dan Assessment Runs](#) di Panduan Pengguna Amazon Inspector.

- Untuk detail API, lihat [ListAssessmentRuns](#) di Referensi AWS CLI Perintah.

## list-assessment-targets

Contoh kode berikut menunjukkan cara menggunakan `list-assessment-targets`.

### AWS CLI

Untuk membuat daftar target penilaian

`list-assessment-targets` Perintah berikut mencantumkan semua target penilaian yang ada:

```
aws inspector list-assessment-targets
```

Output:

```
{
  "assessmentTargetArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq"
  ]
}
```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [ListAssessmentTargets](#) di Referensi AWS CLI Perintah.

## list-assessment-templates

Contoh kode berikut menunjukkan cara menggunakan `list-assessment-templates`.

### AWS CLI

Untuk membuat daftar templat penilaian

`list-assessment-templates` Perintah berikut mencantumkan semua templat penilaian yang ada:

```
aws inspector list-assessment-templates
```

Output:

```
{
```

```
    "assessmentTemplateArns": [  
      "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/  
template/0-4r1V2mAw",  
      "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-  
Uza6ihLh"  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [ListAssessmentTemplates](#) di Referensi AWS CLI Perintah.

## list-event-subscriptions

Contoh kode berikut menunjukkan cara menggunakan `list-event-subscriptions`.

### AWS CLI

Untuk daftar langganan acara

`list-event-subscriptions` Perintah berikut mencantumkan semua langganan acara untuk templat penilaian dengan `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0` ARN dari:

```
aws inspector list-event-subscriptions --resource-arn arn:aws:inspector:us-  
west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0
```

Output:

```
{  
  "subscriptions": [  
    {  
      "eventSubscriptions": [  
        {  
          "event": "ASSESSMENT_RUN_COMPLETED",  
          "subscribedAt": 1459455440.867  
        }  
      ],  
      "resourceArn": "arn:aws:inspector:us-west-2:123456789012:target/0-  
nvgVhaxX/template/0-7sbz2Kz0",  
    }  
  ]  
}
```

```
        "topicArn": "arn:aws:sns:us-west-2:123456789012:exampletopic"
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [ListEventSubscriptions](#) di Referensi AWS CLI Perintah.

## list-findings

Contoh kode berikut menunjukkan cara menggunakan `list-findings`.

### AWS CLI

Untuk membuat daftar temuan

`list-findings` Perintah berikut mencantumkan semua temuan yang dihasilkan:

```
aws inspector list-findings
```

Output:

```
{
  "findingArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4",
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-v5D6fI3v/finding/0-tyvmqBLy"
  ]
}
```

Untuk informasi selengkapnya, lihat Temuan Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [ListFindings](#) di Referensi AWS CLI Perintah.

## list-rules-packages

Contoh kode berikut menunjukkan cara menggunakan `list-rules-packages`.

## AWS CLI

Untuk daftar paket aturan

`list-rules-packages` Perintah berikut mencantumkan semua paket aturan Inspector yang tersedia:

```
aws inspector list-rules-packages
```

Output:

```
{
  "rulesPackageArns": [
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-H5hpSawc",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-JJ0tZiqQ",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-vg5GGHSD"
  ]
}
```

Untuk informasi selengkapnya, lihat Paket dan Aturan Amazon Inspector dalam panduan Amazon Inspector.

- Untuk detail API, lihat [ListRulesPackages](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk sumber daya

`list-tags-for-resource` Perintah berikut mencantumkan semua tag yang terkait dengan template penilaian dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-gcwFliYu`

```
aws inspector list-tags-for-resource --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-gcwFliYu
```

Output:

```
{
  "tags": [
    {
      "key": "Name",
      "value": "Example"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## preview-agents

Contoh kode berikut menunjukkan cara menggunakan `preview-agents`.

### AWS CLI

Untuk pratinjau agen

`preview-agents` Perintah berikut mempratinjau agen yang diinstal pada instans EC2 yang merupakan bagian dari target penilaian dengan ARN: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`

```
aws inspector preview-agents --preview-agents-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Output:

```
{
  "agentPreviews": [
    {
      "agentId": "i-49113b93"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [PreviewAgents](#) di Referensi AWS CLI Perintah.

## **register-cross-account-access-role**

Contoh kode berikut menunjukkan cara menggunakan `register-cross-account-access-role`.

### AWS CLI

Untuk mendaftarkan peran akses lintas akun

`register-cross-account-access-role` Perintah berikut mendaftarkan peran IAM dengan ARN yang digunakan Amazon `arn:aws:iam::123456789012:role/inspector` Inspector untuk mencantumkan instans EC2 Anda di awal proses penilaian saat Anda memanggil perintah agen pratinjau:

```
aws inspector register-cross-account-access-role --role-arn
arn:aws:iam::123456789012:role/inspector
```

Untuk informasi selengkapnya, lihat [Menyiapkan Amazon Inspector](#) di panduan Amazon Inspector.

- Untuk detail API, lihat [RegisterCrossAccountAccessRole](#) di Referensi AWS CLI Perintah.

## **remove-attributes-from-findings**

Contoh kode berikut menunjukkan cara menggunakan `remove-attributes-from-findings`.

### AWS CLI

Untuk menghapus atribut dari temuan

`remove-attributes-from-finding` Perintah berikut menghapus atribut dengan kunci Example dan nilai example dari temuan dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU`

```
aws inspector remove-attributes-from-findings --finding-arns arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-
T8yM9mEU --attribute-keys key=Example,value=example
```

Output:



```
{
  "failedItems": {}
}
```

Untuk informasi selengkapnya, lihat Temuan Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [RemoveAttributesFromFindings](#) di Referensi AWS CLI Perintah.

## set-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `set-tags-for-resource`.

### AWS CLI

Untuk mengatur tag untuk sumber daya

`set-tags-for-resource` Perintah berikut menetapkan tag dengan kunci `Example` dan nilai `example` untuk template penilaian dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0`

```
aws inspector set-tags-for-resource --resource-arn arn:aws:inspector:us-
west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 --tags
key=Example,value=example
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [SetTagsForResource](#) di Referensi AWS CLI Perintah.

## start-assessment-run

Contoh kode berikut menunjukkan cara menggunakan `start-assessment-run`.

### AWS CLI

Untuk memulai penilaian

`start-assessment-run` Perintah berikut memulai proses penilaian bernama `examplerrun` menggunakan template penilaian dengan ARN dari: `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T`

```
aws inspector start-assessment-run --assessment-run-name examplerrun --assessment-template-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T
```

Output:

```
{
  "assessmentRunArn": "arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-j0oroxyY"
}
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [StartAssessmentRun](#) di Referensi AWS CLI Perintah.

## stop-assessment-run

Contoh kode berikut menunjukkan cara menggunakan `stop-assessment-run`.

### AWS CLI

Untuk menghentikan penilaian berjalan

`stop-assessment-run` Perintah berikut menghentikan penilaian yang dijalankan dengan ARN dari `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-j0oroxyY`

```
aws inspector stop-assessment-run --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-j0oroxyY
```

Untuk informasi selengkapnya, lihat Amazon Inspector Assessment Templates dan Assessment Runs di panduan Amazon Inspector.

- Untuk detail API, lihat [StopAssessmentRun](#) di Referensi AWS CLI Perintah.

## subscribe-to-event

Contoh kode berikut menunjukkan cara menggunakan `subscribe-to-event`.

## AWS CLI

Untuk berlangganan acara

Contoh berikut memungkinkan proses pengiriman pemberitahuan Amazon SNS tentang ASSESSMENT\_RUN\_COMPLETED acara ke topik dengan ARN dari `arn:aws:sns:us-west-2:123456789012:exampletopic`

```
aws inspector subscribe-to-event \  
  --event ASSESSMENT_RUN_COMPLETED \  
  --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/  
template/0-7sbz2Kz0 \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:exampletopic
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Amazon Inspector Assessment Templates dan Assessment Runs](#) di panduan Amazon Inspector.

- Untuk detail API, lihat [SubscribeToEvent](#) di Referensi AWS CLI Perintah.

## unsubscribe-from-event

Contoh kode berikut menunjukkan cara menggunakan `unsubscribe-from-event`.

## AWS CLI

Untuk berhenti berlangganan dari suatu acara

`unsubscribe-from-event` Perintah berikut menonaktifkan proses pengiriman pemberitahuan Amazon SNS tentang acara ASSESSMENT\_RUN\_COMPLETED ke topik dengan ARN dari: `arn:aws:sns:us-west-2:123456789012:exampletopic`

```
aws inspector unsubscribe-from-event --event ASSESSMENT_RUN_COMPLETED --resource-arn  
arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 --  
topic arn:aws:sns:us-west-2:123456789012:exampletopic
```

Untuk informasi selengkapnya, lihat [Amazon Inspector Assessment Templates dan Assessment Runs](#) di panduan Amazon Inspector.

- Untuk detail API, lihat [UnsubscribeFromEvent](#) di Referensi AWS CLI Perintah.

## update-assessment-target

Contoh kode berikut menunjukkan cara menggunakan `update-assessment-target`.

### AWS CLI

Untuk memperbarui target penilaian

`update-assessment-target` Perintah berikut memperbarui target penilaian dengan ARN dan nama `Example`, `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX` dan kelompok sumber daya dengan ARN dari `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-yNbgL5Pt`

```
aws inspector update-assessment-target --assessment-target-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX --assessment-target-name Example --resource-group-arn arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-yNbgL5Pt
```

Untuk informasi selengkapnya, lihat Target Penilaian Amazon Inspector di panduan Amazon Inspector.

- Untuk detail API, lihat [UpdateAssessmentTarget](#) di Referensi AWS CLI Perintah.

## AWS IoT contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **accept-certificate-transfer**

Contoh kode berikut menunjukkan cara menggunakan `accept-certificate-transfer`.

#### AWS CLI

Untuk menerima sertifikat perangkat yang ditransfer dari AWS akun lain

`accept-certificate-transfer` Contoh berikut menerima sertifikat perangkat yang ditransfer dari AWS akun lain. Sertifikat diidentifikasi oleh ID-nya.

```
aws iot accept-certificate-transfer \  
  --certificate-id  
  488b6a7f2acdeb00a77384e63c4e40b18bEXAMPLEe57b7272ba44c45e3448142
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Mentransfer sertifikat ke akun lain](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [AcceptCertificateTransfer](#) di Referensi AWS CLI Perintah.

### **add-thing-to-billing-group**

Contoh kode berikut menunjukkan cara menggunakan `add-thing-to-billing-group`.

#### AWS CLI

Contoh 1: Untuk menambahkan sesuatu dengan nama ke grup penagihan

`add-thing-to-billing-group` Contoh berikut menambahkan hal bernama `MyLightBulb` ke grup penagihan bernama `GroupOne`.

```
aws iot add-thing-to-billing-group \  
  --billing-group-name GroupOne \  
  --thing-name MyLightBulb
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menambahkan sesuatu oleh ARN ke grup penagihan

`add-thing-to-billing-group` Contoh berikut menambahkan sesuatu dengan ARN tertentu ke grup penagihan dengan ARN yang ditentukan. Menentukan ARN sangat membantu jika Anda bekerja dengan AWS beberapa Wilayah atau akun. Ini dapat membantu memastikan bahwa Anda menambahkan ke Wilayah dan akun yang tepat.

```
aws iot add-thing-to-thing-group \  
  --billing-group-arn "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne" \  
  --thing-arn "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [AddThingToBillingGroup](#) di Referensi AWS CLI Perintah.

## **add-thing-to-thing-group**

Contoh kode berikut menunjukkan cara menggunakan `add-thing-to-thing-group`.

### AWS CLI

Untuk menambahkan sesuatu ke grup

`add-thing-to-thing-group` Contoh berikut menambahkan hal yang ditentukan ke kelompok hal yang ditentukan.

```
aws iot add-thing-to-thing-group \  
  --thing-name MyLightBulb \  
  --thing-group-name LightBulbs
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [AddThingToThingGroup](#) di Referensi AWS CLI Perintah.

## **associate-targets-with-job**

Contoh kode berikut menunjukkan cara menggunakan `associate-targets-with-job`.

## AWS CLI

Untuk mengasosiasikan kelompok sesuatu dengan pekerjaan berkelanjutan

`associate-targets-with-job` Contoh berikut mengaitkan kelompok hal yang ditentukan dengan pekerjaan berkelanjutan yang ditentukan.

```
aws iot associate-targets-with-job \  
  --targets "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \  
  --job-id "example-job-04"
```

Output:

```
{  
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-04",  
  "jobId": "example-job-04",  
  "description": "example continuous job"  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [AssociateTargetsWithJob](#) di Referensi AWS CLI Perintah.

## **attach-policy**

Contoh kode berikut menunjukkan cara menggunakan `attach-policy`.

### AWS CLI

Contoh 1: Untuk melampirkan kebijakan ke grup benda

`attach-policy` Contoh berikut melampirkan kebijakan yang ditentukan ke grup benda yang diidentifikasi oleh ARN-nya.

```
aws iot attach-policy \  
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \  
  --policy-name "UpdateDeviceCertPolicy"
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

Contoh 2: Untuk melampirkan kebijakan ke sertifikat

`attach-policy` Contoh berikut melampirkan kebijakan `UpdateDeviceCertPolicy` ke kepala sekolah yang ditentukan oleh sertifikat.

```
aws iot attach-policy \  
  --policy-name UpdateDeviceCertPolicy \  
  --target "arn:aws:iot:us-  
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e"
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Melampirkan Kebijakan AWS IoT ke Sertifikat Perangkat di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [AttachPolicy](#) di Referensi AWS CLI Perintah.

## **attach-security-profile**

Contoh kode berikut menunjukkan cara menggunakan `attach-security-profile`.

### AWS CLI

Untuk mengaitkan profil keamanan dengan semua perangkat yang tidak terdaftar

`attach-security-profile` Contoh berikut mengaitkan profil keamanan AWS IoT Device Defender yang `Testprofile` dinamai dengan semua perangkat yang tidak terdaftar di `us-west-2` wilayah untuk akun ini. AWS

```
aws iot attach-security-profile \  
  --security-profile-name Testprofile \  
  --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/  
unregistered-things"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [AttachSecurityProfile](#) di Referensi AWS CLI Perintah.



## **attach-thing-principal**

Contoh kode berikut menunjukkan cara menggunakan `attach-thing-principal`.

### AWS CLI

Untuk melampirkan sertifikat ke barang Anda

`attach-thing-principal` Contoh berikut melampirkan sertifikat untuk `MyTemperatureSensor` benda itu. Sertifikat diidentifikasi oleh ARN. Anda dapat menemukan ARN untuk sertifikat di konsol IoT AWS .

```
aws iot attach-thing-principal \  
  --thing-name MyTemperatureSensor \  
  --principal arn:aws:iot:us-  
west-2:123456789012:cert/2e1eb273792174ec2b9bf4e9b37e6c6c692345499506002a35159767055278e8
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Mengelola Sesuatu dengan Registri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [AttachThingPrincipal](#) di Referensi AWS CLI Perintah.

## **cancel-audit-mitigation-actions-task**

Contoh kode berikut menunjukkan cara menggunakan `cancel-audit-mitigation-actions-task`.

### AWS CLI

Untuk membatalkan tugas tindakan mitigasi audit

`cancel-audit-mitigations-action-task` Contoh berikut membatalkan penerapan tindakan mitigasi untuk tugas yang ditentukan. Anda tidak dapat membatalkan tugas yang sudah selesai.

```
aws iot cancel-audit-mitigation-actions-task  
  --task-id "myActionsTaskId"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [CancelAuditMitigationActionsTask \(Perintah Tindakan Mitigasi\) di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CancelAuditMitigationActionsTask](#) di Referensi AWS CLI Perintah.

## cancel-audit-task

Contoh kode berikut menunjukkan cara menggunakan `cancel-audit-task`.

### AWS CLI

Untuk membatalkan tugas audit

`cancel-audit-task` Contoh berikut membatalkan tugas audit dengan ID tugas yang ditentukan. Anda tidak dapat membatalkan tugas yang sudah selesai.

```
aws iot cancel-audit-task \  
  --task-id a3aea009955e501a31b764abe1bebd3d
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CancelAuditTask](#) di Referensi AWS CLI Perintah.

## cancel-certificate-transfer

Contoh kode berikut menunjukkan cara menggunakan `cancel-certificate-transfer`.

### AWS CLI

Untuk membatalkan transfer sertifikat ke AWS akun yang berbeda

`cancel-certificate-transfer` Contoh berikut membatalkan transfer transfer sertifikat yang ditentukan. Sertifikat diidentifikasi oleh ID sertifikat. Anda dapat menemukan ID untuk sertifikat di konsol AWS IoT.

```
aws iot cancel-certificate-transfer \  
  --certificate-id  
  f0f33678c7c9a046e5cc87b2b1a58dfa0beec26db78add5e605d630e05c7fc8
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mentransfer sertifikat ke akun lain](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [CancelCertificateTransfer](#) di Referensi AWS CLI Perintah.

## cancel-job-execution

Contoh kode berikut menunjukkan cara menggunakan `cancel-job-execution`.

### AWS CLI

Untuk membatalkan eksekusi pekerjaan di perangkat

`cancel-job-execution` Contoh berikut membatalkan eksekusi pekerjaan yang ditentukan pada perangkat. Jika pekerjaan tidak dalam QUEUED keadaan, Anda harus menambahkan `--force` parameter.

```
aws iot cancel-job-execution \  
  --job-id "example-job-03" \  
  --thing-name "MyRPi"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CancelJobExecution](#) di Referensi AWS CLI Perintah.

## cancel-job

Contoh kode berikut menunjukkan cara menggunakan `cancel-job`.

### AWS CLI

Untuk membatalkan pekerjaan

`cancel-job` Contoh berikut membatalkan pekerjaan yang ditentukan.

```
aws iot cancel-job \  
  --job-id "example-job-03"
```

Output:

```
{
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-03",
  "jobId": "example-job-03",
  "description": "example job test"
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CancelJob](#) di Referensi AWS CLI Perintah.

## **clear-default-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `clear-default-authorizer`.

### AWS CLI

Untuk menghapus otorisasi default

`clear-default-authorizer` Contoh berikut menghapus otorisasi kustom default yang saat ini dikonfigurasi. Setelah Anda menjalankan perintah ini, tidak ada otorisasi default. Saat Anda menggunakan otorisasi khusus, Anda harus menentukannya dengan nama di header permintaan HTTP.

```
aws iot clear-default-authorizer
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ClearDefaultAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ClearDefaultAuthorizer](#) di Referensi AWS CLI Perintah.

## **confirm-topic-rule-destination**

Contoh kode berikut menunjukkan cara menggunakan `confirm-topic-rule-destination`.

### AWS CLI

Untuk mengonfirmasi tujuan aturan topik

`confirm-topic-rule-destination` Contoh berikut mengonfirmasi tujuan aturan topik dengan token konfirmasi yang diterima di titik akhir HTTP.

```
aws iot confirm-topic-rule-destination \
  --confirmation-token "AYADeIcmtq-
ZkxfpiWIQqHWM5ucAXwABABVhd3MtY3J5cHRvLXB1YmXpYy1rZXkAREFyY1E0Um1GeDg0V21BZWZ1VjZtZWFRVUJJUKt
aywpPqg8YEsa1lD4B40aJ2s1wEHKMybiF1Ro0ZzYisI0IvslzQY5UmCkq3tV-3f7-
nKfosgIAAAAADAAAEEAAAAAAAAAAAAAAAAAAAAAAi9RMgy-
V19V9m6Iw2xfbw_____wAAAAEAAAAAAAAAAAAAAAAAAEAAAB1hw4SokgUcxiJ3gT06n50NLJVpzyQR1UmPIj5sShqXEQGcC
iufgrzTePl8RZY0Wr006Aj9DiVzJZx-1iD6Pu-
G6PUw1ka07Knzs2B4AD0qfrHUF4pYRTvyUgBnMGUCMQC8ZRmhKqntd_c6Kgrow3bMUDbvNqo2qZr8Z8Jm2rzgseR01An
PIetJ803Z4I1I1F8xX1cdPGP-PV1d0XFemyL8g"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfirmasi tujuan aturan topik](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ConfirmTopicRuleDestination](#) di Referensi AWS CLI Perintah.

## create-audit-suppression

Contoh kode berikut menunjukkan cara menggunakan `create-audit-suppression`.

### AWS CLI

Untuk membuat penindasan temuan audit

`create-audit-suppression` Contoh berikut menciptakan penindasan temuan audit untuk kebijakan bernama "virtualMachinePolicy" yang telah ditandai karena terlalu permisif.

```
aws iot create-audit-suppression \
  --check-name IOT_POLICY_OVERLY_PERMISSIVE_CHECK \
  --resource-identifier
policyVersionIdentifier={"policyName"="virtualMachinePolicy","policyVersionId"="1"}
\
  --no-suppress-indefinitely \
  --expiration-date 2020-10-20
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Audit menemukan penekanan di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreateAuditSuppression](#) di Referensi AWS CLI Perintah.

## create-authorizer

Contoh kode berikut menunjukkan cara menggunakan `create-authorizer`.

### AWS CLI

Untuk membuat otorisasi kustom

`create-authorizer` Contoh berikut membuat otorisasi kustom yang menggunakan fungsi Lambda yang ditentukan sebagai bagian dari layanan otentikasi kustom.

```
aws iot create-authorizer \
  --authorizer-name "CustomAuthorizer" \
  --authorizer-function-arn "arn:aws:lambda:us-
west-2:123456789012:function:CustomAuthorizerFunction" \
  --token-key-name "MyAuthToken" \
  --status ACTIVE \
  --token-signing-public-keys FIRST_KEY="-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIBCgKCAQEA1uJOB4lQPgG/lM6ZfIwo
Z+7ENxAio9q6QD4FFqjGZsvjtYwjoe1RKK0U8Eq9xb503kRSmyIwTzwzm/f4Gf0Y
ZUloJ+t3PUUwHrmbYTAgrCUgRFyggfVwGCPs5ZAX4Eyqt5cr+AIHIiUDbxSa7p
zw0BKPeic0asNJpqT8PkBbRaKylEJh5oo81NDHmVtbBm5A5YiJjqYXLaVAowKzZ
+GqsNvAQ9Jy1wI2VrEa10fL8f1DB/BJLm7zjpfPOHDJQgID0XnZwAlNnZc0hCwIx
50g2LW20y9R/dmqtDmJiVP97Z4GykxPvw1YHrUXY0iW1R3AR/Ac1NhCTGZMwVDB1
lQIDAQAB
-----END PUBLIC KEY-----"
```

Output:

```
{
  "authorizerName": "CustomAuthorizer",
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer2"
}
```

Untuk informasi selengkapnya, lihat [CreateAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [CreateAuthorizer](#) di Referensi AWS CLI Perintah.

## create-billing-group

Contoh kode berikut menunjukkan cara menggunakan `create-billing-group`.

## AWS CLI

Untuk membuat grup penagihan

`create-billing-group` Contoh berikut membuat grup penagihan sederhana bernama `GroupOne`.

```
aws iot create-billing-group \  
  --billing-group-name GroupOne
```

Output:

```
{  
  "billingGroupName": "GroupOne",  
  "billingGroupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne",  
  "billingGroupId": "103de383-114b-4f51-8266-18f209ef5562"  
}
```

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateBillingGroup](#) di Referensi AWS CLI Perintah.

## `create-certificate-from-csr`

Contoh kode berikut menunjukkan cara menggunakan `create-certificate-from-csr`.

## AWS CLI

Untuk membuat sertifikat perangkat dari permintaan penandatanganan sertifikat (CSR)

`create-certificate-from-csr` Contoh berikut membuat sertifikat perangkat dari CSR. Anda dapat menggunakan `openssl` perintah untuk membuat CSR.

```
aws iot create-certificate-from-csr \  
  --certificate-signing-request=file://certificate.csr
```

Output:

```
{  
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/  
c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbbee1428d216d54d53ac9",
```

```

    "certificateId":
      "c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbee1428d216d54d53ac9",
    "certificatePem": "<certificate-text>"
  }

```

Untuk informasi selengkapnya, lihat [CreateCertificateFromCSR](#) di Referensi AWS API IoT.

- Untuk detail API, lihat [CreateCertificateFromCsrdi](#) Referensi AWS CLI Perintah.

## create-custom-metric

Contoh kode berikut menunjukkan cara menggunakan `create-custom-metric`.

### AWS CLI

Untuk membuat metrik kustom yang diterbitkan oleh perangkat Anda ke Device Defender

`create-custom-metric` Contoh berikut membuat metrik khusus yang mengukur persentase baterai.

```

aws iot create-custom-metric \
  --metric-name "batteryPercentage" \
  --metric-type "number" \
  --display-name "Remaining battery percentage." \
  --region us-east-1 \
  --client-request-token "02ccb92b-33e8-4dfa-a0c1-35b181ed26b0"

```

Output:

```

{
  "metricName": "batteryPercentage",
  "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/
batteryPercentage"
}

```

Untuk informasi selengkapnya, lihat [Metrik khusus](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [CreateCustomMetric](#) di Referensi AWS CLI Perintah.

## create-dimension

Contoh kode berikut menunjukkan cara menggunakan `create-dimension`.



## AWS CLI

Untuk membuat dimensi

Berikut ini `create-dimension` menciptakan dimensi dengan filter topik tunggal yang disebut `TopicFilterForAuthMessages`.

```
aws iot create-dimension \  
  --name TopicFilterForAuthMessages \  
  --type TOPIC_FILTER \  
  --string-values device/+/auth
```

Output:

```
{  
  "name": "TopicFilterForAuthMessages",  
  "arn": "arn:aws:iot:eu-west-2:123456789012:dimension/TopicFilterForAuthMessages"  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateDimension](#) di Referensi AWS CLI Perintah.

## **create-domain-configuration**

Contoh kode berikut menunjukkan cara menggunakan `create-domain-configuration`.

## AWS CLI

Untuk membuat konfigurasi domain

`create-domain-configuration` Contoh berikut membuat konfigurasi domain AWS-managed dengan jenis DATA layanan.

```
aws iot create-domain-configuration \  
  --domain-configuration-name "additionalDataDomain" \  
  --service-type "DATA"
```

Output:

```
{
```

```
"domainConfigurationName": "additionalDataDomain",
"domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh"
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir yang Dapat Dikonfigurasi](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateDomainConfiguration](#) di Referensi AWS CLI Perintah.

## create-dynamic-thing-group

Contoh kode berikut menunjukkan cara menggunakan `create-dynamic-thing-group`.

### AWS CLI

Untuk membuat grup hal yang dinamis

`create-dynamic-thing-group` Contoh berikut membuat grup benda dinamis yang berisi benda apa pun dengan atribut suhu yang lebih besar dari 60 derajat. Anda harus mengaktifkan pengindeksan armada AWS IoT sebelum dapat menggunakan grup benda dinamis.

```
aws iot create-dynamic-thing-group \
  --thing-group-name "RoomTooWarm" \
  --query-string "attributes.temperature>60"
```

Output:

```
{
  "thingGroupName": "RoomTooWarm",
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RoomTooWarm",
  "thingGroupId": "9d52492a-fc87-43f4-b6e2-e571d2ffcad1",
  "indexName": "AWS_Things",
  "queryString": "attributes.temperature>60",
  "queryVersion": "2017-09-30"
}
```

Untuk informasi selengkapnya, lihat [Dynamic Thing Groups](#) di AWS Panduan Pengembang IoT.

- Untuk detail API, lihat [CreateDynamicThingGroup](#) di Referensi AWS CLI Perintah.

## create-job

Contoh kode berikut menunjukkan cara menggunakan `create-job`.

### AWS CLI

Contoh 1: Untuk membuat pekerjaan

`create-job` Contoh berikut membuat pekerjaan AWS IoT sederhana yang mengirimkan dokumen JSON ke perangkat. `MyRaspberryPi`

```
aws iot create-job \  
  --job-id "example-job-01" \  
  --targets "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi" \  
  --document file://example-job.json \  
  --description "example job test" \  
  --target-selection SNAPSHOT
```

Output:

```
{  
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",  
  "jobId": "example-job-01",  
  "description": "example job test"  
}
```

Contoh 2: Untuk membuat pekerjaan berkelanjutan

`create-job` Contoh berikut menciptakan pekerjaan yang terus berjalan setelah hal-hal yang ditentukan sebagai target telah menyelesaikan pekerjaan. Dalam contoh ini, targetnya adalah grup benda, jadi ketika perangkat baru ditambahkan ke grup, pekerjaan berkelanjutan berjalan pada hal-hal baru tersebut.

```
aws iot create-job --job-id "example-job-04" --targets "arn:aws:iot:us-west-2:123456789012:thinggroup/" --document file://example-job.json --description "contoh pekerjaan berkelanjutan" --target-selection CONTINUOUS DeadBulbs
```

Output:

```
{  
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-04",
```

```
"jobId": "example-job-04",
"description": "example continuous job"
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CreateJob](#) di Referensi AWS CLI Perintah.

## create-keys-and-certificate

Contoh kode berikut menunjukkan cara menggunakan `create-keys-and-certificate`.

### AWS CLI

Untuk membuat key pair RSA dan mengeluarkan sertifikat X.509

Berikut ini `create-keys-and-certificate` membuat key pair RSA 2048-bit dan mengeluarkan sertifikat X.509 menggunakan kunci publik yang diterbitkan. Karena ini adalah satu-satunya waktu AWS IoT menyediakan kunci pribadi untuk sertifikat ini, pastikan untuk menyimpannya di lokasi yang aman.

```
aws iot create-keys-and-certificate \
  --certificate-pem-outfile "myTest.cert.pem" \
  --public-key-outfile "myTest.public.key" \
  --private-key-outfile "myTest.private.key"
```

Output:

```
{
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/9894ba17925e663f1d29c23af4582b8e3b7619c31f3fbd93adcb51ae54b83dc2",
  "certificateId":
  "9894ba17925e663f1d29c23af4582b8e3b7619c31f3fbd93adcb51ae54b83dc2",
  "certificatePem": "
-----BEGIN CERTIFICATE-----
MIICiTCCEXAMPLE6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMC
VVMxCzAJBgNVBAGYEXAMPLEAwDgYDVoQHEwdTZWF0dGx1MQ8wDQYDVoQKEwZBbWF6
b24xFDASBgNVBASTC01BTSEXAMPLE2x1MRIwEAYDVoQDEwLUZXN0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGFTYEXAMPLEb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCXAMPLEJBgNVBAGTA1dBMRAdDgYD
VoQHEwdTZWF0dGx1MQ8wDQYDVoQKEwZBbWF6b24xFAQEXAMPLEsTC01BTSBDb25z
```

```

b2x1MRIwEAYDVQQDEw1UZXXN0Q21sYWMxHzAdBgkqhkiG9w0BCQEXAMPLE251QGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+aEXAMPLE
EXAMPLEfEvySwTc2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZEXAMPLEELG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZncvQAEXAMPLEEWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9qEXAMPLEEyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEEXAMPLEEBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQ0CAQ8AMIIBCgKCAQEAEXAMPLE1nnyJwKSMHw4h\nMMEXAMPLEuuN/
dMAS3fyce8DW/4+EXAMPLEYjmoF/YVF/gHr99VEEXAMPLE5VF13\n59VK7cEXAMPLE67GK+y+jikqX0gHh/
xJTtwo
+sGpWEXAMPLEDz18x0d2ka4tCzuWEXAMPLEeahJbYkCPUBSU8opVkr7qkEXAMPLE1DR6sx2Hocli00Ltu6Fkw91swQWEX
\GB3ZPrNh0PzQYvjUSTzecyNCx2EXAMPLEvp9mQ0UXP6p1fgxwKRX2fEXAMPLEDa
\nhJLXkX3rHU2xbxJSq7D+XEXAMPLEcw+LyFhI5mgFR188eGdsAEXAMPLE1nI9EesG\nFQIDAQAB\n-----
END PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----\nkey omitted for security
reasons\n-----END RSA PRIVATE KEY-----\n"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat dan Mendaftarkan Sertifikat Perangkat AWS IoT di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CreateKeysAndCertificate](#) di Referensi AWS CLI Perintah.

## create-mitigation-action

Contoh kode berikut menunjukkan cara menggunakan `create-mitigation-action`.

### AWS CLI

Untuk membuat tindakan mitigasi

`create-mitigation-action` Contoh berikut mendefinisikan tindakan mitigasi bernama `AddThingsToQuarantineGroup1Action` bahwa, ketika diterapkan, memindahkan hal-hal ke dalam kelompok hal bernama `QuarantineGroup1`. Tindakan ini mengesampingkan grup benda dinamis.

```
aws iot create-mitigation-action --cli-input-json file::params.json
```

Isi dari `params.json`:

```
{
  "actionName": "AddThingsToQuarantineGroup1Action",
  "actionParams": {
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "QuarantineGroup1"
      ],
      "overrideDynamicGroups": true
    }
  },
  "roleArn": "arn:aws:iam::123456789012:role/service-role/MoveThingsToQuarantineGroupRole"
}
```

Output:

```
{
  "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/AddThingsToQuarantineGroup1Action",
  "actionId": "992e9a63-a899-439a-aa50-4e20c52367e1"
}
```

Untuk informasi selengkapnya, lihat [CreateMitigationAction \(Perintah Tindakan Mitigasi\)](#) di [Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreateMitigationAction](#) di Referensi AWS CLI Perintah.

## **create-ota-update**

Contoh kode berikut menunjukkan cara menggunakan `create-ota-update`.

### AWS CLI

Untuk membuat pembaruan OTA untuk digunakan dengan Amazon FreeRTOS

`create-ota-update` Contoh berikut membuat AWS IoT OTAupdate pada kelompok target hal atau grup. Ini adalah bagian dari pembaruan Amazon over-the-air FreeRTOS yang memungkinkan Anda untuk menyebarkan gambar firmware baru ke satu perangkat atau sekelompok perangkat.

```
aws iot create-ota-update \  
  --cli-input-json file://create-ota-update.json
```

Isi dari create-ota-update.json:

```
{  
  "otaUpdateId": "ota12345",  
  "description": "A critical update needed right away.",  
  "targets": [  
    "device1",  
    "device2",  
    "device3",  
    "device4"  
  ],  
  "targetSelection": "SNAPSHOT",  
  "awsJobExecutionsRolloutConfig": {  
    "maximumPerMinute": 10  
  },  
  "files": [  
    {  
      "fileName": "firmware.bin",  
      "fileLocation": {  
        "stream": {  
          "streamId": "004",  
          "fileId": 123  
        }  
      },  
      "codeSigning": {  
        "awsSignerJobId": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"  
      }  
    }  
  ]  
  "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_role"  
}
```

Output:

```
{  
  "otaUpdateId": "ota12345",  
  "awsIotJobId": "job54321",  
  "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",  
  "awsIotJobArn": "arn:aws:iot:us-west-2:123456789012:job/itsajob",
```

```
"otaUpdateStatus": "CREATE_IN_PROGRESS"
}
```

Untuk informasi selengkapnya, lihat [createOtaUpdate di Referensi API IoT.AWS](#)

- Untuk detail API, lihat [CreateOtaUpdate](#) di Referensi AWS CLI Perintah.

## create-policy-version

Contoh kode berikut menunjukkan cara menggunakan `create-policy-version`.

### AWS CLI

Untuk memperbarui kebijakan dengan versi baru

`create-policy-version` Contoh berikut memperbarui definisi kebijakan, membuat versi kebijakan baru. Contoh ini juga menjadikan versi baru sebagai default.

```
aws iot create-policy-version \
  --policy-name UpdateDeviceCertPolicy \
  --policy-document file://policy.json \
  --set-as-default
```

Isi dari `policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:UpdateCertificate",
      "Resource": "*"
    }
  ]
}
```

Output:

```
{
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
  "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\", \"Action\": \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
```



```
"policyVersionId": "2",
"isDefaultVersion": true
}
```

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreatePolicyVersion](#) di Referensi AWS CLI Perintah.

## create-policy

Contoh kode berikut menunjukkan cara menggunakan `create-policy`.

### AWS CLI

Untuk membuat kebijakan AWS IoT

`create-policy` Contoh berikut membuat kebijakan AWS IoT bernama.

`TemperatureSensorPolicy` `policy.json` berisi pernyataan yang memungkinkan tindakan kebijakan AWS IoT.

```
aws iot create-policy \
  --policy-name TemperatureSensorPolicy \
  --policy-document file://policy.json
```

Isi dari `policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:Publish",
        "iot:Receive"
      ],
      "Resource": [
        "arn:aws:iot:us-west-2:123456789012:topic/topic_1",
        "arn:aws:iot:us-west-2:123456789012:topic/topic_2"
      ]
    },
    {
      "Effect": "Allow",
```

```

    "Action": [
      "iot:Subscribe"
    ],
    "Resource": [
      "arn:aws:iot:us-west-2:123456789012:topicfilter/topic_1",
      "arn:aws:iot:us-west-2:123456789012:topicfilter/topic_2"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "iot:Connect"
    ],
    "Resource": [
      "arn:aws:iot:us-west-2:123456789012:client/basicPubSub"
    ]
  }
]
}

```

### Output:

```

{
  "policyName": "TemperatureSensorPolicy",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy",
  "policyDocument": "{
    \"Version\": \"2012-10-17\",
    \"Statement\": [
      {
        \"Effect\": \"Allow\",
        \"Action\": [
          \"iot:Publish\",
          \"iot:Receive\"
        ],
        \"Resource\": [
          \"arn:aws:iot:us-west-2:123456789012:topic/topic_1\",
          \"arn:aws:iot:us-west-2:123456789012:topic/topic_2\"
        ]
      },
      {
        \"Effect\": \"Allow\",
        \"Action\": [

```

```

        \"iot:Subscribe\"
    ],
    \"Resource\": [
        \"arn:aws:iot:us-west-2:123456789012:topicfilter/topic_1\",
        \"arn:aws:iot:us-west-2:123456789012:topicfilter/topic_2\"
    ]
},
{
    \"Effect\": \"Allow\",
    \"Action\": [
        \"iot:Connect\"
    ],
    \"Resource\": [
        \"arn:aws:iot:us-west-2:123456789012:client/basicPubSub\"
    ]
}
]
}],
    \"policyVersionId\": \"1\"
}

```

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreatePolicy](#) di Referensi AWS CLI Perintah.

## create-provisioning-claim

Contoh kode berikut menunjukkan cara menggunakan `create-provisioning-claim`.

### AWS CLI

Untuk membuat klaim penyediaan

`create-provisioning-claim` Contoh berikut membuat klaim penyediaan dari template penyediaan.

```
aws iot create-provisioning-claim \
  --template-name MyTestProvisioningTemplate
```

Output:

```
{
```

```

    "certificateId":
      "78de02184b2ce80cf8fb709bda59e62b19fb83513590483eb0434589476ab09f",
      "certificatePem": "-----BEGIN CERTIFICATE-----\nMIIDdzCCA1
+gAwIBAgIUXSZHEBLztMLZ2fHG
14gV0NymYY0wDQYJKoZIhvcNAQEL
\nBQAwfjELMAKGA1UEBhMCMVVMxEzARBgNVBAgMC1dhc2hpbmd0b24xEDA0Bg
VBACM\nB1N1YXR0bGUxGDAWBgNVBAoMD0FtYXpvbi5jb20gSW5jLjEgMB4GA1UECwwXQW1h
\nem9uIElVVCBQcm9
2aXNpb25pbmcxDDAKBgNVBAUTAzEuMDAeFw0yMDA3Mjg1NjUxMDZaFw0yMDA3Mjg1NjUxMDZaMEsxBHBHbGVB
AMMQDFhNDEyM2VkNmIxYjU3MzE3\nZTgzMTJmY2MzN2FiNTdhY2MzYTZkZGVjOGQ5OGY3NzUwMWR1Mjc0YjhmYTQ
xN2Iw\nnggEiMA0GCSqGSIb3EXAMPLEAA4IBDwAwggEKAoIBAQBDBhKI94ktKLqTwnj+ay0q1\nTAJt/
N6s6IJDZv1
rYjkC0E7wzaeY3TprWk03S29vUzVuE0XHXQXZbihgpg2m6fza\nkwm9/
wpjzE9ny5+xkPGVH4Wnwz7yK5m8S0agL
T96cRBSWnWmon0WdY0GKVzni0CA\n+iyGudgrFKm7Eae/
v18oXrf82Kt0AG04xG0KE2WKYHsT1fx3c9xZh1XP/eX
Lhv00\n+1Gp0WwV9PbhKfrxliKJ5q6sL5nVUaUHq6h1QPYwsATe0vAp3u0ak5zgTyL0fg7Y
\nPyKk6VYwLW62r+V
YBSForEM0Ahkq3LsP/rjxpEKmi2W41PVS6oFZRKcD+H1Kyil5\nAgMBAAGjIDAeMAwGA1UdEwEB/
wQMAAwDgYDV
R0PAQH/BAQDAgeAMA0GCSqGSIb3\nDQEBcWUAA4IBAQAQgix2k6nVqbZFKq97/fZBzLGS0dyz5rT/
E41cDIRX+1j
EPW41\nnw0D+2sXheCZLZZnSkvIiP74IToNeXDrjdcaodeGFVHIElRjhMIq+4ZebPbRLtidF
\nRc2hfcTAlqq9Z6v
5Vk6BeM1tu0RqH1wPoVUccLPya8EjNCbnJZUmGd0frN/Y9pho\n5ikV+HPeZhG/k6dhE2GsQJyKfVHL/
uBgKSily
1bRyWU1r6qcpWBNBHjUoD7Hg0wD
\nnzMh4XRb2FQDsqFalkCSYmeL8IVC49sgPD90typ5uteGMTy62usAAUQdq/f
ZvrWg\n0kFpwMVnGKVKT7Kg0kK0LzKW0BB2Jm4/gmrJ\n-----END CERTIFICATE-----\n",
      "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCg
KCAQEAWYSiPeJLSi6k8J4/msjq
\nUwCbfzer0iCQ2b5a2I5AtB08M2nmN06a1pNN0tvb1M1bhDlx10F2W4oYKYN
pun8\n2pFpvf8KY8xPZ8ufsZDx1R+Fp8M+8iuZvEtGoC0/enEQUl1pqJz1nWNBilc54tA
\nngPoshrnYKxSpuxGn
v79fKF63/NirTgBjuMRtChNlimEXAMPLE3PcWYZVz/3ly4b9\nNPPRqdf1cPT24Sn68ZYiieaurC
+Z1VGLB6uoZU
D2MLAE3jrwKd7tGp0c4E8i9H40\n2D8ip0lWMC1utq/
LWAUhaKxDDgIZKty7D/648aRCpotluJT1UuqBWUSNa/h9
Ssop\nEQIDAQAB\n-----END PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIBAAKCAQEAWYSiPeJLSi6k8J4/

```

```

msjqtUwCbzfer0iCQ2b5a2I5AtB08M2n
\nmN06a1pNN0tvb1M1bhDlx10F2W4oYKYNpun82pFpvf8KY8xPZ8ufsz
Dx1R+Fp8M+\n8iuZvEtGoC0/enEQUl1pqJz1nWNBilc54tAgPoshrnYKxSpuxGnv79fKF63/Nir
\nTgBjuMRtCh
NlimB7E9X8d3PcWYZVz/31y4b9NPPRqdFlcPT24Sn68ZYiieaurC+Z
\n1VGLB6uoZUD2MLAE3jrwKd7tGp0c4E8i
9H402D8ip01WMC1utq/1WAUhaKxDDgIZ\nKty7D/648aRCpotluJT1UuqBWUSnA/
h9SsopeQIDAQABAOIBAEAybN
QUtx9T2/nK\ntZT2pA4iugecxI4dz+DmT0XVXs5VJmrx/
nBSq6ejXExEpSIM04RY7LE3ZdJcnd56\nF7tQkkY7yR
VzfxHeXFU1kr0IPuxWebN0rRoPZr+1Rser+ww2aBC525+88pVuR6tM
\nm3pgkrR2ycCj9Fd0UoQxdjHBHaM5PDMj
9aSxCKdg3nReepeGwsR2TQA+m2vVxWk7\nnou0+91eTOP+/QfP7P8Zj0Ik02Xiv1RcVDyN/
E4QXPKuIkM/8vS8VK+
E9pATQ0MtB\n21w8R/YU5AJd6jlEXAMPLEGU2UzRzInNWiLtkPPPqgqXXhx0f+mxByjcMa1VJk0L
\nh0G2R0UCgY
EA+R0cHNHy/XbsP7Fih0hEh+6Q2QxQ2ncBUPYbBazrR8Hn+7SCICQK
\nVyYfd8Ajfq3e7RsKVL5S1MBp7S1idxak
bIn28fKfPn62DaemGCIoyDgLf+eUxBx
\nngzbCiBZga8brfurza43UZjKZLpg3hq721+FeAiXi1Nma4Yr9YWEHEN
8CgYEAxuwT\npzdwWmsiFzfsAw0sy9ySDA/xr5WRWzJyAqUsjsks6rxNzWebpufnYHcmtW7pLdqM
\nkboHwN2pXa
kmZvrk2nKkEMq5brBYGDxuxDe+V369Bianx8aZfyIsckA70wXW1w1h
\nngRC5rQ4X0gp3+Jmw7eA08LRYDjaN846+
Qbt02KcCgYAWS0UL51bijQR0ZwI0dz27\nnFQVuCAYsp748aurcRTACCj8jbnK/
QbqTNlxWsaH7ssBjZKo2D5sAqY
BRtASW0Dab\naHXsDhVm2Jye+ESLohMaClOyCkT3118yqXIcEDStM07f01Ryag164EiJvSIrMfny\nnNL/
fXVjCSH
/udCxdzPt+7QKBgQC+LAD7rxd4J9538hTqpc4XK9vxRbrMXEH55XH
\nHbMa2x0NZXpmeTgEQBukyohCVceyRhK9
i0e6irZTjVXgh0eoTpC8VXkzcnzouTiQ
\nEFQQSGfnp7Ioe6UIz23715pKdudzSNkMSKrG924ktv7CyDBF1gBQI5g
aDoHnddJBJ\nnPRtIZQKBgA8MASxtTxQntRwXXzR92U0vAighiuRkB/mx9jQpUcK1qiqHbkAMqgNF
\nPFCBYIUbFT
iYKKKeJNbyJQvjfsJcKAnaFJ+RnTxk0Q6Wjm20peJ/ii4QiDdnigoE\nnvd1c5cFQewWb4/
zqAtPdinkPLN94ileI
79XQdc7R1J0jpgTimL+V\n-----END RSA PRIVATE KEY-----\n"
    },
    "expiration": 1595955066.0
}

```

Untuk informasi selengkapnya, lihat [Penyediaan oleh pengguna tepercaya di Panduan Pengembang Inti AWS IoT](#).

- Untuk detail API, lihat [CreateProvisioningClaim](#) di Referensi AWS CLI Perintah.

## create-provisioning-template-version

Contoh kode berikut menunjukkan cara menggunakan `create-provisioning-template-version`.

### AWS CLI

Untuk membuat versi template penyediaan

Contoh berikut membuat versi untuk template penyediaan yang ditentukan. Tubuh versi baru disediakan dalam file `template.json`.

```
aws iot create-provisioning-template-version \  
  --template-name widget-template \  
  --template-body file://template.json
```

Isi dari `template.json`:

```
{  
  "Parameters" : {  
    "DeviceLocation": {  
      "Type": "String"  
    }  
  },  
  "Mappings": {  
    "LocationTable": {  
      "Seattle": {  
        "LocationUrl": "https://example.aws"  
      }  
    }  
  },  
  "Resources" : {  
    "thing" : {  
      "Type" : "AWS::IoT::Thing",  
      "Properties" : {  
        "AttributePayload" : {  
          "version" : "v1",  
          "serialNumber" : "serialNumber"  
        }  
      }  
    }  
  }  
}
```

```

        "ThingName" : {"Fn::Join":["",["ThingPrefix_",
{"Ref":"SerialNumber"}]]},
        "ThingTypeName" : {"Fn::Join":["",["ThingTypePrefix_",
{"Ref":"SerialNumber"}]]},
        "ThingGroups" : ["widgets", "WA"],
        "BillingGroup": "BillingGroup"
    },
    "OverrideSettings" : {
        "AttributePayload" : "MERGE",
        "ThingTypeName" : "REPLACE",
        "ThingGroups" : "DO_NOTHING"
    }
},
"certificate" : {
    "Type" : "AWS::IoT::Certificate",
    "Properties" : {
        "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
        "Status" : "Active"
    }
},
"policy" : {
    "Type" : "AWS::IoT::Policy",
    "Properties" : {
        "PolicyDocument" : {
            "Version": "2012-10-17",
            "Statement": [{
                "Effect": "Allow",
                "Action":["iot:Publish"],
                "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/foo/
bar"]
            }]
        }
    }
},
"DeviceConfiguration": {
    "FallbackUrl": "https://www.example.com/test-site",
    "LocationUrl": {
        "Fn::FindInMap": ["LocationTable",{"Ref": "DeviceLocation"},
"LocationUrl"]}
    }
}
}

```

## Output:

```
{
  "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/widget-
template",
  "templateName": "widget-template",
  "versionId": 2,
  "isDefaultVersion": false
}
```

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [CreateProvisioningTemplateVersion](#) di Referensi AWS CLI Perintah.

## create-provisioning-template

Contoh kode berikut menunjukkan cara menggunakan `create-provisioning-template`.

### AWS CLI

Untuk membuat template penyediaan

`create-provisioning-template` Contoh berikut membuat template penyediaan seperti yang didefinisikan oleh file `template.json`

```
aws iot create-provisioning-template \
  --template-name widget-template \
  --description "A provisioning template for widgets" \
  --provisioning-role-arn arn:aws:iam::123456789012:role/Provision_role \
  --template-body file://template.json
```

Isi dari `template.json`:

```
{
  "Parameters" : {
    "DeviceLocation": {
      "Type": "String"
    }
  },
  "Mappings": {
    "LocationTable": {
```



```
        "Seattle": {
            "LocationUrl": "https://example.aws"
        }
    },
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : {
                    "version" : "v1",
                    "serialNumber" : "serialNumber"
                },
                "ThingName" : {"Fn::Join":["",["ThingPrefix_",
{"Ref":"SerialNumber"}]]},
                "ThingTypeName" : {"Fn::Join":["",["ThingTypePrefix_",
{"Ref":"SerialNumber"}]]},
                "ThingGroups" : ["widgets", "WA"],
                "BillingGroup": "BillingGroup"
            },
            "OverrideSettings" : {
                "AttributePayload" : "MERGE",
                "ThingTypeName" : "REPLACE",
                "ThingGroups" : "DO_NOTHING"
            }
        },
        "certificate" : {
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
                "Status" : "Active"
            }
        },
        "policy" : {
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : {
                    "Version": "2012-10-17",
                    "Statement": [{
                        "Effect": "Allow",
                        "Action":["iot:Publish"],
                        "Resource": ["arn:aws:iot:us-east-1:504350838278:topic/foo/
bar"]
                    }]
                }
            }
        }
    }
}
```

```

    }
  }
},
"DeviceConfiguration": {
  "FallbackUrl": "https://www.example.com/test-site",
  "LocationUrl": {
    "Fn::FindInMap": ["LocationTable", {"Ref": "DeviceLocation"}],
    "LocationUrl": ""
  }
}
}
}
}

```

Output:

```

{
  "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/widget-
template",
  "templateName": "widget-template",
  "defaultVersionId": 1
}

```

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [CreateProvisioningTemplate](#) di Referensi AWS CLI Perintah.

## create-role-alias

Contoh kode berikut menunjukkan cara menggunakan `create-role-alias`.

### AWS CLI

Untuk membuat alias peran

`create-role-alias` Contoh berikut menciptakan alias peran yang dipanggil `LightBulbRole` untuk peran tertentu.

```

aws iot create-role-alias \
  --role-alias LightBulbRole \
  --role-arn arn:aws:iam::123456789012:role/lightbulbrole-001

```

**Output:**

```
{
  "roleAlias": "LightBulbRole",
  "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/LightBulbRole"
}
```

Untuk informasi selengkapnya, lihat [CreateRoleAlias](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [CreateRoleAlias](#) di Referensi AWS CLI Perintah.

**create-scheduled-audit**

Contoh kode berikut menunjukkan cara menggunakan `create-scheduled-audit`.

**AWS CLI**

Untuk membuat audit terjadwal

`create-scheduled-audit` Contoh berikut membuat audit terjadwal yang berjalan setiap minggu, pada hari Rabu, untuk memeriksa apakah sertifikat CA atau sertifikat perangkat kedaluwarsa.

```
aws iot create-scheduled-audit \
  --scheduled-audit-name WednesdayCertCheck \
  --frequency WEEKLY \
  --day-of-week WED \
  --target-check-names CA_CERTIFICATE_EXPIRING_CHECK
  DEVICE_CERTIFICATE_EXPIRING_CHECK
```

**Output:**

```
{
  "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
  WednesdayCertCheck"
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateScheduledAudit](#) di Referensi AWS CLI Perintah.

## create-security-profile

Contoh kode berikut menunjukkan cara menggunakan `create-security-profile`.

### AWS CLI

Untuk membuat profil keamanan

`create-security-profile` Contoh berikut membuat profil keamanan yang memeriksa apakah bandwidth seluler melebihi ambang batas atau jika lebih dari 10 kegagalan otorisasi terjadi dalam periode lima menit.

```
aws iot create-security-profile \
  --security-profile-name PossibleIssue \
  --security-profile-description "Check to see if authorization fails 10 times in
  5 minutes or if cellular bandwidth exceeds 128" \
  --behaviors "[{"name":"CellularBandwidth","metric":"aws:message-byte-size",
  "criteria":{"comparisonOperator":"greater-than","value":{"count":128},
  "consecutiveDatapointsToAlarm":1,"consecutiveDatapointsToClear":1}},{"name
  ":"Authorization","metric":"aws:num-authorization-failures","criteria":
  {"comparisonOperator":"less-than","value":{"count":10},"durationSeconds
  ":300,"consecutiveDatapointsToAlarm":1,"consecutiveDatapointsToClear":1}]]"
```

Output:

```
{
  "securityProfileName": "PossibleIssue",
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
  PossibleIssue"
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateSecurityProfile](#) di Referensi AWS CLI Perintah.

## create-stream

Contoh kode berikut menunjukkan cara menggunakan `create-stream`.

### AWS CLI

Untuk membuat aliran untuk mengirimkan satu atau lebih file besar dalam potongan melalui MQTT

`create-stream` Contoh berikut membuat aliran untuk mengirimkan satu atau lebih file besar dalam potongan melalui MQTT. Aliran mengangkut byte data dalam potongan atau blok yang dikemas sebagai pesan MQTT dari sumber seperti S3. Anda dapat memiliki satu atau lebih file yang terkait dengan aliran.

```
aws iot create-stream \  
  --cli-input-json file://create-stream.json
```

Isi dari `create-stream.json`:

```
{  
  "streamId": "stream12345",  
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",  
  "files": [  
    {  
      "fileId": 123,  
      "s3Location": {  
        "bucket": "codesign-ota-bucket",  
        "key": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"  
      }  
    }  
  ],  
  "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_stream_role"  
}
```

Output:

```
{  
  "streamId": "stream12345",  
  "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",  
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",  
  "streamVersion": "1"  
}
```

Untuk informasi selengkapnya, lihat [CreateStream](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [CreateStream](#) di Referensi AWS CLI Perintah.

## **create-thing-group**

Contoh kode berikut menunjukkan cara menggunakan `create-thing-group`.

## AWS CLI

Contoh 1: Untuk membuat grup benda

`create-thing-group` Contoh berikut membuat grup hal bernama `LightBulbs` dengan deskripsi dan dua atribut.

```
aws iot create-thing-group \  
  --thing-group-name LightBulbs \  
  --thing-group-properties "thingGroupDescription=\"Generic bulb group\  
attributePayload={attributes={Manufacturer=AnyCompany,wattage=60}}"
```

Output:

```
{  
  "thingGroupName": "LightBulbs",  
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs",  
  "thingGroupId": "9198bf9f-1e76-4a88-8e8c-e7140142c331"  
}
```

Contoh 2: Untuk membuat grup benda yang merupakan bagian dari grup induk

Berikut ini `create-thing-group` membuat grup hal bernama `HalogenBulbs` yang memiliki kelompok hal induk bernama `LightBulbs`.

```
aws iot create-thing-group \  
  --thing-group-name HalogenBulbs \  
  --parent-group-name LightBulbs
```

Output:

```
{  
  "thingGroupName": "HalogenBulbs",  
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs",  
  "thingGroupId": "f4ec6b84-b42b-499d-9ce1-4dbd4d4f6f6e"  
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateThingGroup](#) di Referensi AWS CLI Perintah.

## create-thing-type

Contoh kode berikut menunjukkan cara menggunakan `create-thing-type`.

### AWS CLI

Untuk mendefinisikan tipe benda

`create-thing-type` Contoh berikut mendefinisikan tipe benda dan atribut terkait.

```
aws iot create-thing-type \  
  --thing-type-name "LightBulb" \  
  --thing-type-properties "thingTypeDescription=light bulb type,  
searchableAttributes=wattage,model"
```

Output:

```
{  
  "thingTypeName": "LightBulb",  
  "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",  
  "thingTypeId": "ce3573b0-0a3c-45a7-ac93-4e0ce14cd190"  
}
```

Untuk informasi selengkapnya, lihat [Tipe Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateThingType](#) di Referensi AWS CLI Perintah.

## create-thing

Contoh kode berikut menunjukkan cara menggunakan `create-thing`.

### AWS CLI

Contoh 1: Untuk membuat catatan sesuatu di registri

`create-thing` Contoh berikut membuat entri untuk perangkat di registri hal AWS IoT.

```
aws iot create-thing \  
  --thing-name SampleIoTThing
```

Output:

```
{
  "thingName": "SampleIoTThing",
  "thingArn": "arn:aws:iot:us-west-2: 123456789012:thing/SampleIoTThing",
  "thingId": " EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE "
}
```

Contoh 2: Untuk mendefinisikan sesuatu yang terkait dengan tipe benda

`create-thing` Contoh berikut membuat sesuatu yang memiliki jenis hal yang ditentukan dan atributnya.

```
aws iot create-thing \
  --thing-name "MyLightBulb" \
  --thing-type-name "LightBulb" \
  --attribute-payload '{"attributes": {"wattage": "75", "model": "123"}}'
```

Output:

```
{
  "thingName": "MyLightBulb",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
  "thingId": "40da2e73-c6af-406e-b415-15acae538797"
}
```

Untuk informasi selengkapnya, lihat [Cara Mengelola Sesuatu dengan Registri](#) dan [Jenis Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateThing](#) di Referensi AWS CLI Perintah.

## **create-topic-rule-destination**

Contoh kode berikut menunjukkan cara menggunakan `create-topic-rule-destination`.

### AWS CLI

Untuk membuat tujuan aturan topik

`create-topic-rule-destination` Contoh berikut membuat tujuan aturan topik untuk titik akhir HTTP.

```
aws iot create-topic-rule-destination \
```



```
--destination-configuration httpUrlConfiguration={confirmationUrl=https://
example.com}
```

### Output:

```
{
  "topicRuleDestination": {
    "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "status": "IN_PROGRESS",
    "statusReason": "Awaiting confirmation. Confirmation message sent on
2020-07-09T22:47:54.154Z; no response received from the endpoint.",
    "httpUrlProperties": {
      "confirmationUrl": "https://example.com"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat tujuan aturan topik](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [CreateTopicRuleDestination](#) di Referensi AWS CLI Perintah.

## create-topic-rule

Contoh kode berikut menunjukkan cara menggunakan `create-topic-rule`.

### AWS CLI

Untuk membuat aturan yang mengirimkan peringatan Amazon SNS

`create-topic-rule` Contoh berikut membuat aturan yang mengirimkan pesan Amazon SNS saat pembacaan tingkat kelembaban tanah, seperti yang ditemukan dalam bayangan perangkat, rendah.

```
aws iot create-topic-rule \
  --rule-name "LowMoistureRule" \
  --topic-rule-payload file://plant-rule.json
```

Contoh ini memerlukan kode JSON berikut untuk disimpan ke file bernama `plant-rule.json`:

```
{
```

```
"sql": "SELECT * FROM '$aws/things/MyRPi/shadow/update/accepted' WHERE
state.reported.moisture = 'low'\n",
"description": "Sends an alert whenever soil moisture level readings are too
low.",
"ruleDisabled": false,
"awsIotSqlVersion": "2016-03-23",
"actions": [{
  "sns": {
    "targetArn": "arn:aws:sns:us-
west-2:123456789012:MyRPiLowMoistureTopic",
    "roleArn": "arn:aws:iam::123456789012:role/service-role/
MyRPiLowMoistureTopicRole",
    "messageFormat": "RAW"
  }
}]
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Aturan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreateTopicRule](#) di Referensi AWS CLI Perintah.

## **delete-account-audit-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-account-audit-configuration`.

### AWS CLI

Untuk menonaktifkan semua pemeriksaan audit untuk AWS akun Anda

`delete-account-audit-configuration` Contoh berikut mengembalikan pengaturan default untuk AWS IoT Device Defender untuk akun ini, menonaktifkan semua pemeriksaan audit dan menghapus data konfigurasi. Ini juga menghapus audit terjadwal untuk akun ini. Gunakan perintah ini dengan hati-hati.

```
aws iot delete-account-audit-configuration \
--delete-scheduled-audits
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteAccountAuditConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-audit-suppression**

Contoh kode berikut menunjukkan cara menggunakan `delete-audit-suppression`.

### AWS CLI

Untuk menghapus penindasan temuan audit

`delete-audit-suppression` Contoh berikut menghapus penekanan pencarian audit untuk `DEVICE_CERTIFICATE_EXPIRING_CHECK`.

```
aws iot delete-audit-suppression \  
  --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \  
  --resource-identifier deviceCertificateId="c7691e<shortened>"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Audit menemukan penekanan di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteAuditSuppression](#) di Referensi AWS CLI Perintah.

## **delete-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `delete-authorizer`.

### AWS CLI

Untuk menghapus otorisasi kustom

`delete-authorizer` Contoh berikut menghapus authorizer bernama `CustomAuthorizer`. `Authorizer` khusus harus dalam `INACTIVE` status sebelum Anda dapat menghapusnya.

```
aws iot delete-authorizer \  
  --authorizer-name CustomAuthorizer
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteAuthorizer](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteAuthorizer](#) di Referensi AWS CLI Perintah.

## **delete-billing-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-billing-group`.

### AWS CLI

Untuk menghapus grup penagihan

`delete-billing-group` Contoh berikut menghapus grup penagihan yang ditentukan. Anda dapat menghapus grup penagihan meskipun berisi satu atau beberapa hal.

```
aws iot delete-billing-group \  
  --billing-group-name BillingGroupTwo
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteBillingGroup](#) di Referensi AWS CLI Perintah.

## **delete-ca-certificate**

Contoh kode berikut menunjukkan cara menggunakan `delete-ca-certificate`.

### AWS CLI

Untuk menghapus sertifikat CA

`delete-ca-certificate` Contoh berikut menghapus sertifikat CA dengan ID sertifikat yang ditentukan.

```
aws iot delete-ca-certificate \  
  --certificate-id  
  f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteCertificate](#) di Referensi API IoT.AWS

- Untuk detail API, lihat [DeleteCertificate](#) di Referensi AWS CLI Perintah.

## **delete-certificate**

Contoh kode berikut menunjukkan cara menggunakan `delete-certificate`.

### AWS CLI

Untuk menghapus sertifikat perangkat

`delete-certificate` Contoh berikut menghapus sertifikat perangkat dengan ID yang ditentukan.

```
aws iot delete-certificate \  
  --certificate-id  
  c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddb1428d216d54d53ac9
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteCertificate](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DeleteCertificate](#) di Referensi AWS CLI Perintah.

## **delete-custom-metric**

Contoh kode berikut menunjukkan cara menggunakan `delete-custom-metric`.

### AWS CLI

Untuk menghapus metrik kustom

`delete-custom-metric` Contoh berikut menghapus metrik kustom.

```
aws iot delete-custom-metric \  
  --metric-name batteryPercentage \  
  --region us-east-1
```

Output:

```
HTTP 200
```

Untuk informasi selengkapnya, lihat [Metrik khusus](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [DeleteCustomMetric](#) di Referensi AWS CLI Perintah.

## **delete-dimension**

Contoh kode berikut menunjukkan cara menggunakan `delete-dimension`.

### AWS CLI

Untuk menghapus dimensi

`delete-dimension` Contoh berikut menghapus dimensi yang disebut `TopicFilterForAuthMessages`.

```
aws iot delete-dimension \  
  --name TopicFilterForAuthMessages
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteDimension](#) di Referensi AWS CLI Perintah.

## **delete-domain-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-domain-configuration`.

### AWS CLI

Untuk menghapus konfigurasi domain

`delete-domain-configuration` Contoh berikut menghapus konfigurasi domain bernama `additionalDataDomain` dari AWS akun Anda.

```
aws iot delete-domain-configuration \  
  --domain-configuration-name "additionalDataDomain" \  
  --domain-configuration-status "OK"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Titik Akhir yang Dapat Dikonfigurasi](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteDomainConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-dynamic-thing-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-dynamic-thing-group`.

### AWS CLI

Untuk menghapus grup hal dinamis

`delete-dynamic-thing-group` Contoh berikut menghapus kelompok hal dinamis yang ditentukan.

```
aws iot delete-dynamic-thing-group \  
  --thing-group-name "RoomTooWarm"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Dynamic Thing Groups](#) di AWS Panduan Pengembang IoT.

- Untuk detail API, lihat [DeleteDynamicThingGroup](#) di Referensi AWS CLI Perintah.

## **delete-job-execution**

Contoh kode berikut menunjukkan cara menggunakan `delete-job-execution`.

### AWS CLI

Untuk menghapus eksekusi pekerjaan

`delete-job-execution` Contoh berikut menghapus eksekusi pekerjaan dari pekerjaan yang ditentukan pada perangkat. Gunakan `describe-job-execution` untuk mendapatkan nomor eksekusi.

```
aws iot delete-job-execution \  
  --job-id "example-job-02"
```

```
--thing-name "MyRaspberryPi"  
--execution-number 1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteJobExecution](#) di Referensi AWS CLI Perintah.

## **delete-job**

Contoh kode berikut menunjukkan cara menggunakan `delete-job`.

### AWS CLI

Untuk menghapus pekerjaan

`delete-job` Contoh berikut menghapus pekerjaan yang ditentukan. Dengan menentukan `--force` opsi, pekerjaan dihapus bahkan jika statusnya `IN_PROGRESS`.

```
aws iot delete-job \  
  --job-id "example-job-04" \  
  --force
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteJob](#) di Referensi AWS CLI Perintah.

## **delete-mitigation-action**

Contoh kode berikut menunjukkan cara menggunakan `delete-mitigation-action`.

### AWS CLI

Untuk menghapus tindakan mitigasi

`delete-mitigation-action` Contoh berikut menghapus tindakan mitigasi yang ditentukan.



```
aws iot delete-mitigation-action \  
  --action-name AddThingsToQuarantineGroup1Action
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteMitigationAction \(Perintah Tindakan Mitigasi\) di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteMitigationAction](#) di Referensi AWS CLI Perintah.

## **delete-ota-update**

Contoh kode berikut menunjukkan cara menggunakan `delete-ota-update`.

### AWS CLI

Untuk menghapus pembaruan OTA

`delete-ota-update` Contoh berikut menghapus pembaruan OTA yang ditentukan.

```
aws iot delete-ota-update \  
  --ota-update-id ota12345 \  
  --delete-stream \  
  --force-delete-aws-job
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteOtaUpdate di Referensi API IoT.AWS](#)

- Untuk detail API, lihat [DeleteOtaUpdate](#) di Referensi AWS CLI Perintah.

## **delete-policy-version**

Contoh kode berikut menunjukkan cara menggunakan `delete-policy-version`.

### AWS CLI

Untuk menghapus versi kebijakan

`delete-policy-version` Contoh berikut menghapus versi 2 dari kebijakan yang ditentukan dari AWS akun Anda.

```
aws iot delete-policy-version \  
  --policy-name UpdateDeviceCertPolicy \  
  --policy-version-id 2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DeletePolicyVersion](#) di Referensi AWS CLI Perintah.

## delete-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-policy`.

### AWS CLI

Untuk menghapus kebijakan

`delete-policy` Contoh berikut menghapus kebijakan yang ditentukan dari AWS akun Anda.

```
aws iot delete-policy --policy-name UpdateDeviceCertPolicy
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## delete-provisioning-template-version

Contoh kode berikut menunjukkan cara menggunakan `delete-provisioning-template-version`.

### AWS CLI

Untuk menghapus versi template penyediaan

`delete-provisioning-template-version` Contoh berikut menghapus versi 2 dari template penyediaan yang ditentukan.

```
aws iot delete-provisioning-template-version \  
  --template-name UpdateDeviceCertPolicy \  
  --template-version-id 2
```

```
--version-id 2 \  
--template-name "widget-template"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [DeleteProvisioningTemplateVersion](#) di Referensi AWS CLI Perintah.

## **delete-provisioning-template**

Contoh kode berikut menunjukkan cara menggunakan `delete-provisioning-template`.

### AWS CLI

Untuk menghapus template penyediaan

`delete-provisioning-template` Contoh berikut menghapus template penyediaan yang ditentukan.

```
aws iot delete-provisioning-template \  
--template-name widget-template
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [DeleteProvisioningTemplate](#) di Referensi AWS CLI Perintah.

## **delete-registration-code**

Contoh kode berikut menunjukkan cara menggunakan `delete-registration-code`.

### AWS CLI

Untuk menghapus cod pendaftaran Anda

`delete-registration-code` Contoh berikut menghapus kode pendaftaran khusus akun AWS IoT.

```
aws iot delete-registration-code
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan Sertifikat Anda Sendiri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteRegistrationCode](#) di Referensi AWS CLI Perintah.

## **delete-role-alias**

Contoh kode berikut menunjukkan cara menggunakan `delete-role-alias`.

### AWS CLI

Untuk menghapus AWS alias peran IoT

`delete-role-alias` Contoh berikut menghapus alias peran AWS IoT bernama `LightBulbRole`

```
aws iot delete-role-alias \  
  --role-alias LightBulbRole
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengotorisasi Panggilan Langsung ke AWS Layanan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteRoleAlias](#) di Referensi AWS CLI Perintah.

## **delete-scheduled-audit**

Contoh kode berikut menunjukkan cara menggunakan `delete-scheduled-audit`.

### AWS CLI

Untuk menghapus audit terjadwal

`delete-scheduled-audit` Contoh berikut menghapus audit terjadwal AWS IoT Device Defender yang diberi nama `AWSIoTDeviceDefenderDailyAudit`

```
aws iot delete-scheduled-audit \  
  --scheduled-audit-name AWSIoTDeviceDefenderDailyAudit
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteScheduledAudit](#) di Referensi AWS CLI Perintah.

## **delete-security-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-security-profile`.

### AWS CLI

Untuk menghapus profil keamanan

`delete-security-profile` Contoh berikut menghapus profil keamanan bernama `PossibleIssue`.

```
aws iot delete-security-profile \  
  --security-profile-name PossibleIssue
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteSecurityProfile](#) di Referensi AWS CLI Perintah.

## **delete-stream**

Contoh kode berikut menunjukkan cara menggunakan `delete-stream`.

### AWS CLI

Untuk menghapus aliran

`delete-stream` Contoh berikut menghapus aliran yang ditentukan.

```
aws iot delete-stream \  
  --stream-id stream12345
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteStream](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DeleteStream](#) di Referensi AWS CLI Perintah.

## **delete-thing-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-thing-group`.

### AWS CLI

Untuk menghapus grup benda

`delete-thing-group` Contoh berikut menghapus kelompok hal yang ditentukan. Anda tidak dapat menghapus grup benda jika berisi grup hal anak.

```
aws iot delete-thing-group \  
  --thing-group-name DefectiveBulbs
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteThingGroup](#) di Referensi AWS CLI Perintah.

## **delete-thing-type**

Contoh kode berikut menunjukkan cara menggunakan `delete-thing-type`.

### AWS CLI

Contoh 1: Untuk menghapus tipe benda

`delete-thing-type` Contoh berikut menghapus jenis hal usang.

```
aws iot delete-thing-type \  
  --thing-type-name "obsoleteThingType"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Tipe Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteThingType](#) di Referensi AWS CLI Perintah.

## **delete-thing**

Contoh kode berikut menunjukkan cara menggunakan `delete-thing`.

### AWS CLI

Untuk menampilkan informasi rinci tentang suatu hal

`delete-thing` Contoh berikut menghapus sesuatu dari registri AWS IoT untuk AWS akun Anda.

```
aws iot delete-thing --thing-name "" FourthBulb
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Mengelola Sesuatu dengan Registri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteThing](#) di Referensi AWS CLI Perintah.

## **delete-topic-rule-destination**

Contoh kode berikut menunjukkan cara menggunakan `delete-topic-rule-destination`.

### AWS CLI

Untuk menghapus tujuan aturan topik

`delete-topic-rule-destination` Contoh berikut menghapus tujuan aturan topik yang ditentukan.

```
aws iot delete-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus tujuan aturan topik](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteTopicRuleDestination](#) di Referensi AWS CLI Perintah.

## **delete-topic-rule**

Contoh kode berikut menunjukkan cara menggunakan `delete-topic-rule`.

### AWS CLI

Untuk menghapus aturan

`delete-topic-rule` Contoh berikut menghapus aturan yang ditentukan.

```
aws iot delete-topic-rule \  
  --rule-name "LowMoistureRule"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Aturan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteTopicRule](#) di Referensi AWS CLI Perintah.

## **delete-v2-logging-level**

Contoh kode berikut menunjukkan cara menggunakan `delete-v2-logging-level`.

### AWS CLI

Untuk menghapus level logging untuk grup sesuatu

`delete-v2-logging-level` Contoh berikut menghapus tingkat logging untuk kelompok hal yang ditentukan.

```
aws iot delete-v2-logging-level \  
  --target-type THING_GROUP \  
  --target-name LightBulbs
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteV2 LoggingLevel](#) di Referensi AWS CLI Perintah.

## **deprecate-thing-type**

Contoh kode berikut menunjukkan cara menggunakan `deprecate-thing-type`.



## AWS CLI

Contoh 1: Untuk menghentikan tipe benda

`deprecate-thing-type` Contoh berikut mencela tipe benda sehingga pengguna tidak dapat mengaitkan hal-hal baru dengannya.

```
aws iot deprecate-thing-type \  
  --thing-type-name "obsoleteThingType"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk membalikkan penghentian tipe benda

`deprecate-thing-type` Contoh berikut membalikkan penghentian tipe benda, yang memungkinkan pengguna untuk mengaitkan hal-hal baru dengannya lagi.

```
aws iot deprecate-thing-type \  
  --thing-type-name "obsoleteThingType" \  
  --undo-deprecate
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Tipe Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeprecateThingType](#) di Referensi AWS CLI Perintah.

## **describe-account-audit-configuration**

Contoh kode berikut menunjukkan cara menggunakan `describe-account-audit-configuration`.

## AWS CLI

Untuk melihat setelan konfigurasi audit saat ini

`describe-account-audit-configuration` Contoh berikut mencantumkan pengaturan saat ini untuk konfigurasi audit AWS IoT Device Defender Anda.

```
aws iot describe-account-audit-configuration
```

## Output:

```
{
  "roleArn": "arn:aws:iam::123456789012:role/service-role/
AWSIoTDeviceDefenderAudit_1551201085996",
  "auditNotificationTargetConfigurations": {
    "SNS": {
      "targetArn": "arn:aws:sns:us-west-2:123456789012:ddaudits",
      "roleArn": "arn:aws:iam::123456789012:role/service-role/
AWSIoTDeviceDefenderAudit",
      "enabled": true
    }
  },
  "auditCheckConfigurations": {
    "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
      "enabled": true
    },
    "CA_CERTIFICATE_EXPIRING_CHECK": {
      "enabled": true
    },
    "CONFLICTING_CLIENT_IDS_CHECK": {
      "enabled": true
    },
    "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
      "enabled": true
    },
    "DEVICE_CERTIFICATE_SHARED_CHECK": {
      "enabled": true
    },
    "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
      "enabled": true
    },
    "LOGGING_DISABLED_CHECK": {
      "enabled": true
    },
    "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {
      "enabled": true
    },
    "REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK": {
      "enabled": true
    },
    "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
      "enabled": true
    }
  }
}
```

```
}
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeAccountAuditConfiguration](#) di Referensi AWS CLI Perintah.

## describe-audit-finding

Contoh kode berikut menunjukkan cara menggunakan `describe-audit-finding`.

### AWS CLI

Untuk membuat daftar detail untuk temuan audit

`describe-audit-finding` Contoh berikut mencantumkan detail untuk temuan audit AWS IoT Device Defender yang ditentukan. Audit dapat menghasilkan banyak temuan. Gunakan `list-audit-findings` perintah untuk mendapatkan daftar temuan dari audit untuk mendapatkan `findingId`.

```
aws iot describe-audit-finding \
  --finding-id "ef4826b8-e55a-44b9-b460-5c485355371b"
```

Output:

```
{
  "finding": {
    "findingId": "ef4826b8-e55a-44b9-b460-5c485355371b",
    "taskId": "873ed69c74a9ec8fa9b8e88e9abc4661",
    "checkName": "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",
    "taskStartTime": 1576012045.745,
    "findingTime": 1576012046.168,
    "severity": "CRITICAL",
    "nonCompliantResource": {
      "resourceType": "IOT_POLICY",
      "resourceIdentifier": {
        "policyVersionIdentifier": {
          "policyName": "smp-ggrass-group_Core-policy",
          "policyVersionId": "1"
        }
      }
    }
  },
}
```

```

    "reasonForNonCompliance": "Policy allows broad access to IoT data plane
actions: [iot:Subscribe, iot:Connect, iot:GetThingShadow, iot>DeleteThingShadow,
iot:UpdateThingShadow, iot:Publish].",
    "reasonForNonComplianceCode":
"ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS"
  }
}

```

Untuk informasi selengkapnya, lihat [Periksa Hasil Audit \(Perintah Audit\)](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeAuditFinding](#) di Referensi AWS CLI Perintah.

## describe-audit-mitigation-actions-task

Contoh kode berikut menunjukkan cara menggunakan `describe-audit-mitigation-actions-task`.

### AWS CLI

Untuk menunjukkan rincian tugas tindakan mitigasi audit

`describe-audit-mitigation-actions-task` Contoh berikut menunjukkan rincian untuk tugas yang ditentukan, di `ResetPolicyVersionAction` mana diterapkan untuk temuan. Hasilnya meliputi kapan tugas dimulai dan berakhir, berapa banyak temuan yang ditargetkan (dan hasilnya), dan definisi tindakan yang diterapkan sebagai bagian dari tugas ini.

```

aws iot describe-audit-mitigation-actions-task \
  --task-id ResetPolicyTask01

```

### Output:

```

{
  "taskStatus": "COMPLETED",
  "startTime": "2019-12-10T15:13:19.457000-08:00",
  "endTime": "2019-12-10T15:13:19.947000-08:00",
  "taskStatistics": {
    "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
      "totalFindingsCount": 1,
      "failedFindingsCount": 0,
      "succeededFindingsCount": 1,
      "skippedFindingsCount": 0,

```

```

        "canceledFindingsCount": 0
      }
    },
    "target": {
      "findingIds": [
        "ef4826b8-e55a-44b9-b460-5c485355371b"
      ]
    },
    "auditCheckToActionsMapping": {
      "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": [
        "ResetPolicyVersionAction"
      ]
    },
    "actionsDefinition": [
      {
        "name": "ResetPolicyVersionAction",
        "id": "1ea0b415-bef1-4a01-bd13-72fb63c59afb",
        "roleArn": "arn:aws:iam::123456789012:role/service-role/
ReplacePolicyVersionRole",
        "actionParams": {
          "replaceDefaultPolicyVersionParams": {
            "templateName": "BLANK_POLICY"
          }
        }
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [DescribeAuditMitigationActionsTask \(Perintah Tindakan Mitigasi\) di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeAuditMitigationActionsTask](#) di Referensi AWS CLI Perintah.

## describe-audit-suppression

Contoh kode berikut menunjukkan cara menggunakan `describe-audit-suppression`.

### AWS CLI

Untuk mendapatkan detail tentang penindasan temuan audit

`describe-audit-suppression` Contoh berikut mencantumkan rincian tentang penindasan temuan audit.

```
aws iot describe-audit-task \  
  --task-id "787ed873b69cb4d6cdbae6ddd06996c5"
```

Output:

```
{  
  "taskStatus": "COMPLETED",  
  "taskType": "SCHEDULED_AUDIT_TASK",  
  "taskStartTime": 1596168096.157,  
  "taskStatistics": {  
    "totalChecks": 1,  
    "inProgressChecks": 0,  
    "waitingForDataCollectionChecks": 0,  
    "compliantChecks": 0,  
    "nonCompliantChecks": 1,  
    "failedChecks": 0,  
    "canceledChecks": 0  
  },  
  "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",  
  "auditDetails": {  
    "DEVICE_CERTIFICATE_EXPIRING_CHECK": {  
      "checkRunStatus": "COMPLETED_NON_COMPLIANT",  
      "checkCompliant": false,  
      "totalResourcesCount": 195,  
      "nonCompliantResourcesCount": 2  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Audit menemukan penekanan di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeAuditSuppression](#) di Referensi AWS CLI Perintah.

## describe-audit-task

Contoh kode berikut menunjukkan cara menggunakan `describe-audit-task`.

### AWS CLI

Untuk mendapatkan informasi tentang instance audit

`describe-audit-task` Contoh berikut mendapatkan informasi tentang instance audit AWS IoT Device Defender. Jika audit selesai, statistik ringkasan untuk menjalankan disertakan dalam hasil.

```
aws iot describe-audit-task \  
  --task-id a3aea009955e501a31b764abe1bebd3d
```

Output:

```
{  
  "taskStatus": "COMPLETED",  
  "taskType": "ON_DEMAND_AUDIT_TASK",  
  "taskStartTime": 1560356923.434,  
  "taskStatistics": {  
    "totalChecks": 3,  
    "InProgressChecks": 0,  
    "waitingForDataCollectionChecks": 0,  
    "compliantChecks": 3,  
    "nonCompliantChecks": 0,  
    "failedChecks": 0,  
    "canceledChecks": 0  
  },  
  "auditDetails": {  
    "CA_CERTIFICATE_EXPIRING_CHECK": {  
      "checkRunStatus": "COMPLETED_COMPLIANT",  
      "checkCompliant": true,  
      "totalResourcesCount": 0,  
      "nonCompliantResourcesCount": 0  
    },  
    "DEVICE_CERTIFICATE_EXPIRING_CHECK": {  
      "checkRunStatus": "COMPLETED_COMPLIANT",  
      "checkCompliant": true,  
      "totalResourcesCount": 6,  
      "nonCompliantResourcesCount": 0  
    },  
    "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {  
      "checkRunStatus": "COMPLETED_COMPLIANT",  
      "checkCompliant": true,  
      "totalResourcesCount": 0,  
      "nonCompliantResourcesCount": 0  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeAuditTask](#) di Referensi AWS CLI Perintah.

## describe-authorizer

Contoh kode berikut menunjukkan cara menggunakan `describe-authorizer`.

### AWS CLI

Untuk mendapatkan informasi tentang otorisasi khusus

`describe-authorizer` Contoh berikut menampilkan rincian untuk authorizer kustom yang ditentukan.

```
aws iot describe-authorizer \  
  --authorizer-name CustomAuthorizer
```

Output:

```
{  
  "authorizerDescription": {  
    "authorizerName": "CustomAuthorizer",  
    "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/  
CustomAuthorizer",  
    "authorizerFunctionArn": "arn:aws:lambda:us-  
west-2:123456789012:function:CustomAuthorizerFunction",  
    "tokenKeyName": "MyAuthToken",  
    "tokenSigningPublicKeys": {  
      "FIRST_KEY": "-----BEGIN PUBLIC KEY-----  
\nMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEA1uJOB4lQPgG/1M6ZfIwo  
\nZ+7ENxAio9q6QD4FFqjGZsvjtYwjoe1RKK0U8Eq9xb503kRSmyIwTzwm/f4Gf0Y  
\nZUloJ+t3PUUwHrmbYTAgrCUgRFyggjfgVwGCPs5ZAX4Eyqt5cr+AIHIiUDbxSa7p  
\nzw0BKPeic0asNJpqT8PkBbRaKYLEJh5oo81NDHmVtbBm5A5YiJjqYXLaVAowKzZ\n+GqsNvAQ9Jy1wI2VrEa10fL8f1DB/BJLm7zjpfPOHDJQgID0XnZwAlNnZc0hCwIx\n50g2LW20y9R/  
dmqtDmJiVP97Z4GykxPvwLYHrUXY0iW1R3AR/Ac1NhCTGZMwVDB1\nlQIDAQAB\n-----END PUBLIC  
KEY-----"  
    },  
    "status": "ACTIVE",  
    "creationDate": 1571245658.069,  
    "lastModifiedDate": 1571245658.069  
  }  
}
```



Untuk informasi selengkapnya, lihat [DescribeAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DescribeAuthorizer](#) di Referensi AWS CLI Perintah.

## describe-billing-group

Contoh kode berikut menunjukkan cara menggunakan `describe-billing-group`.

### AWS CLI

Untuk mendapatkan informasi tentang grup penagihan

`describe-billing-group` Contoh berikut mendapatkan informasi untuk grup penagihan yang ditentukan.

```
aws iot describe-billing-group --billing-group-name GroupOne
```

Output:

```
{
  "billingGroupName": "GroupOne",
  "billingGroupId": "103de383-114b-4f51-8266-18f209ef5562",
  "billingGroupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne",
  "version": 1,
  "billingGroupProperties": {},
  "billingGroupMetadata": {
    "creationDate": 1560199355.378
  }
}
```

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeBillingGroup](#) di Referensi AWS CLI Perintah.

## describe-ca-certificate

Contoh kode berikut menunjukkan cara menggunakan `describe-ca-certificate`.

### AWS CLI

Untuk mendapatkan detail tentang sertifikat CA

`describe-ca-certificate` Contoh berikut menampilkan rincian untuk sertifikat CA tertentu.

```
aws iot describe-ca-certificate \
  --certificate-id
  f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467
```

Output:

```
{
  "certificateDescription": {
    "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
    "certificateId":
    "f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
    "status": "INACTIVE",
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIICzzCCAbegEXAMPLEJANVEPWX18taPMA0GCSqGSIb3DQEBBQUAMB4xCzAJBgNV
\nBAYTA1VMTQ8wDQYDVQQKDAZBbWF6b24wHhcNMk0TI0MjEzMTU1WhcNMjkwOTIx
\nMjEzMTU1WjAeMQswCQYDVQQGEwJVUzEPMA0GA1UECgwGQW1hem9uMIIBIjANBgkq
\nhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAZd3R3ioalCS0MhFWfBrVGR036EK07Uaf
\nVdz9EXAMPLE1VczICbADnATK522kEIB51/18Vz1FtAhQL5V5eybXKnB7QebNer5m
\n4Yibx7shR5oqNzFsrXWxuugN5+w5gEfqNMaw0jhF4Lscu1KG49yuqjcdU19/13ua
\n3B2gxs1Pe7TiWWvUskzxb01F2WCshbEJvqY8fIWtGYCjTeJAgQ9hvZx/69XhKen
\nwV9LJw0QxrsUS0Ty8IHwbB8fRy72VM3u7fJoaU+n04jd5cqaoEPtzoefUEXAMPLE
\nyVAJpqHwgbYbcUfn7V+AB6yh1+0Fa1rEQGuZDPGyJs1xwr5vh8nRewIDAQABoxAw
\nDjAMBgNVHRMEBTADAQH/MA0GCSqGSIb3DQEBBQUAA4IBAQA+3a5CV3IJg0nd0AgI
\nBgVMtmYzTvqAngx26aG9/spvCjXckh2SBF+EcBlCFwH1yakwjJL1dR4yarnrfxgI
\nEqP4A0YVimAVoQ5FBwnloHe16+3qtDib1U9DeXBUCtS55EcfrEXAMPLEYtXdqU5C
\nU9ia4KAjV0dxW1+EFYMwX5eGeb0gDTNHBylV6B/f0SZiQAwDYp4x3B+gAP+a/bWB
\nu1um0qtBdWe6L6/83L+JhaTByqV25iVJ4c/UZUnG8926wU1DM9zQvEXuEVvzZ7+m\n4PSNqst/
nV0vnLpoG4e0WgcJgANuB33CSWtjWSuYsbhmQQRknGhREXAMPLEZT4fm\nfo0e\n-----END
CERTIFICATE-----\n",
    "ownedBy": "123456789012",
    "creationDate": 1569365372.053,
    "autoRegistrationStatus": "DISABLE",
    "lastModifiedDate": 1569365372.053,
    "customerVersion": 1,
    "generationId": "c5c2eb95-140b-4f49-9393-6aaac85b2a90",
    "validity": {
      "notBefore": 1569360675.0,
      "notAfter": 1884720675.0
    }
  }
}
```

```
}

```

Untuk informasi selengkapnya, lihat [DescribeCertificate](#) di Referensi API IoT.AWS

- Untuk detail API, lihat [DescribeCertificate](#) di Referensi AWS CLI Perintah.

## describe-certificate

Contoh kode berikut menunjukkan cara menggunakan `describe-certificate`.

### AWS CLI

Untuk mendapatkan informasi tentang sertifikat

`describe-certificate` Contoh berikut menampilkan rincian untuk sertifikat yang ditentukan.

```
aws iot describe-certificate \
  --certificate-id
  "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e"
```

Output:

```
{
  "certificateDescription": {
    "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
    "certificateId":
    "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
    "status": "ACTIVE",
    "certificatePem": "-----BEGIN CERTIFICATE-----
MIICiTEXAMPLEQD6m7oRw0uX0jANBgqhkiG9w0BAQUFADCBiDELMakGA1UEBhMC
VVMxCzAJBgNVBEXAMPLEMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC01BTSBDEXAMPLE1MRIwEAYDVQQDEwLUZXN0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5EXAMPLEcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMakGA1UEBhMCVVMxCzAJBgNEXAMPLEdBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC01BEXAMPLEz
b2xEXAMPLEYDVQQDEwLUZXN0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8EXAMPLEZIHvcNAQEbbQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvYswTc2XADZ4nB+BLYEXAMPLEpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7EXAMPLEGBzZswY6786m86gpE
Ibb30hjZnzcvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFEXAMPLEAtCu4
nUhVVxYUnEXAMPLE8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GEXAMPLE10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb

```

```

NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
  "ownedBy": "123456789012",
  "creationDate": 1541022751.983,
  "lastModifiedDate": 1541022751.983,
  "customerVersion": 1,
  "transferData": {},
  "generationId": "6974fbcd-2e61-4114-bc5e-4204cc79b045",
  "validity": {
    "notBefore": 1541022631.0,
    "notAfter": 2524607999.0
  }
}
}
}

```

Untuk informasi selengkapnya, lihat [DescribeCertificate](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DescribeCertificate](#) di Referensi AWS CLI Perintah.

## describe-custom-metric

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-metric`.

### AWS CLI

Untuk mendapatkan informasi tentang metrik kustom Device Defender

`describe-custom-metric` Contoh berikut mendapatkan informasi tentang metrik kustom bernama `myCustomMetric`.

```

aws iot describe-custom-metric \
  --metric-name myCustomMetric

```

Output:

```

{
  "metricName": "myCustomMetric",
  "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/myCustomMetric",
  "metricType": "number",
  "displayName": "My custom metric",
  "creationDate": 2020-11-17T23:02:12.879000-09:00,
  "lastModifiedDate": 2020-11-17T23:02:12.879000-09:00
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Metrik khusus](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [DescribeCustomMetric](#) di Referensi AWS CLI Perintah.

## **describe-default-authorizer**

Contoh kode berikut menunjukkan cara menggunakan `describe-default-authorizer`.

### AWS CLI

Untuk mendapatkan informasi tentang otorisasi kustom default

`describe-default-authorizer` Contoh berikut menampilkan rincian untuk authorizer kustom default.

```
aws iot describe-default-authorizer
```

Output:

```
{
  "authorizerName": "CustomAuthorizer",
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer"
}
```

Untuk informasi selengkapnya, lihat [DescribeDefaultAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DescribeDefaultAuthorizer](#) di Referensi AWS CLI Perintah.

## **describe-dimension**

Contoh kode berikut menunjukkan cara menggunakan `describe-dimension`.

### AWS CLI

Untuk mendapatkan informasi tentang dimensi

`describe-dimension` Contoh berikut mendapatkan informasi tentang dimensi bernama `TopicFilterForAuthMessages`.

```
aws iot describe-dimension \  
  --name TopicFilterForAuthMessages
```

Output:

```
{  
  "name": "TopicFilterForAuthMessages",  
  "arn": "arn:aws:iot:eu-west-2:123456789012:dimension/  
TopicFilterForAuthMessages",  
  "type": "TOPIC_FILTER",  
  "stringValues": [  
    "device/+/auth"  
  ],  
  "creationDate": 1578620223.255,  
  "lastModifiedDate": 1578620223.255  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeDimension](#) di Referensi AWS CLI Perintah.

## describe-domain-configuration

Contoh kode berikut menunjukkan cara menggunakan `describe-domain-configuration`.

AWS CLI

Untuk menggambarkan konfigurasi domain

`describe-domain-configuration` Contoh berikut menampilkan rincian tentang konfigurasi domain yang ditentukan.

```
aws iot describe-domain-configuration \  
  --domain-configuration-name "additionalDataDomain"
```

Output:

```
{  
  "domainConfigurationName": "additionalDataDomain",  
  "domainConfigurationArn": "arn:aws:iot:us-  
east-1:758EXAMPLE143:domainconfiguration/additionalDataDomain/norpw",
```

```
"domainName": "d055exampleed74y71zfd-ats.beta.us-east-1.iot.amazonaws.com",
"serverCertificates": [],
"domainConfigurationStatus": "ENABLED",
"serviceType": "DATA",
"domainType": "AWS_MANAGED",
"lastStatusChangeDate": 1601923783.774
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir yang Dapat Dikonfigurasi](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeDomainConfiguration](#) di Referensi AWS CLI Perintah.

## describe-endpoint

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoint`.

### AWS CLI

Contoh 1: Untuk mendapatkan titik AWS akhir Anda saat ini

`describe-endpoint` Contoh berikut mengambil AWS endpoint default yang semua perintah diterapkan.

```
aws iot describe-endpoint
```

Output:

```
{
  "endpointAddress": "abc123defghijk.iot.us-west-2.amazonaws.com"
}
```

Untuk informasi selengkapnya, lihat [DescribeEndpoint](#) di Panduan Pengembang AWS IoT.

Contoh 2: Untuk mendapatkan titik akhir ATS Anda

`describe-endpoint` Contoh berikut mengambil titik akhir Amazon Trust Services (ATS).

```
aws iot describe-endpoint \
  --endpoint-type iot:Data-ATS
```

**Output:**

```
{
  "endpointAddress": "abc123defghijk-ats.iot.us-west-2.amazonaws.com"
}
```

Untuk informasi selengkapnya, lihat [Sertifikat X.509 dan IoT di Panduan Pengembang AWSAWS IoT](#).

- Untuk detail API, lihat [DescribeEndpoint](#) di Referensi AWS CLI Perintah.

**describe-event-configurations**

Contoh kode berikut menunjukkan cara menggunakan `describe-event-configurations`.

**AWS CLI**

Untuk menunjukkan jenis acara mana yang diterbitkan

`describe-event-configurations` Contoh berikut mencantumkan konfigurasi yang mengontrol peristiwa mana yang dihasilkan ketika sesuatu ditambahkan, diperbarui, atau dihapus.

```
aws iot describe-event-configurations
```

**Output:**

```
{
  "eventConfigurations": {
    "CA_CERTIFICATE": {
      "Enabled": false
    },
    "CERTIFICATE": {
      "Enabled": false
    },
    "JOB": {
      "Enabled": false
    },
    "JOB_EXECUTION": {
      "Enabled": false
    },
    "POLICY": {
      "Enabled": false
    }
  }
}
```



```
    },
    "THING": {
      "Enabled": false
    },
    "THING_GROUP": {
      "Enabled": false
    },
    "THING_GROUP_HIERARCHY": {
      "Enabled": false
    },
    "THING_GROUP_MEMBERSHIP": {
      "Enabled": false
    },
    "THING_TYPE": {
      "Enabled": false
    },
    "THING_TYPE_ASSOCIATION": {
      "Enabled": false
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Pesan Acara](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeEventConfigurations](#) di Referensi AWS CLI Perintah.

## describe-index

Contoh kode berikut menunjukkan cara menggunakan `describe-index`.

### AWS CLI

Untuk mengambil status saat ini dari indeks benda

`describe-index` Contoh berikut mengambil status saat ini dari indeks hal.

```
aws iot describe-index \
  --index-name "AWS_Things"
```

Output:

```
{
  "indexName": "AWS_Things",
```

```
"indexStatus": "ACTIVE",  
"schema": "REGISTRY_AND_SHADOW_AND_CONNECTIVITY_STATUS"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengindeksan Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeIndex](#) di Referensi AWS CLI Perintah.

## describe-job-execution

Contoh kode berikut menunjukkan cara menggunakan `describe-job-execution`.

### AWS CLI

Untuk mendapatkan detail eksekusi untuk pekerjaan di perangkat

`describe-job-execution` Contoh berikut mendapatkan rincian eksekusi untuk pekerjaan yang ditentukan.

```
aws iot describe-job-execution \  
  --job-id "example-job-01" \  
  --thing-name "MyRaspberryPi"
```

Output:

```
{  
  "execution": {  
    "jobId": "example-job-01",  
    "status": "QUEUED",  
    "statusDetails": {},  
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi",  
    "queuedAt": 1560787023.636,  
    "lastUpdatedAt": 1560787023.636,  
    "executionNumber": 1,  
    "versionNumber": 1  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DescribeJobExecution](#) di Referensi AWS CLI Perintah.

## describe-job

Contoh kode berikut menunjukkan cara menggunakan `describe-job`.

### AWS CLI

Untuk mendapatkan status terperinci untuk suatu pekerjaan

`describe-job` Contoh berikut mendapatkan status rinci untuk pekerjaan yang ID-nya `example-job-01`.

```
aws iot describe-job \  
  --job-id "example-job-01"
```

### Output:

```
{  
  "job": {  
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",  
    "jobId": "example-job-01",  
    "targetSelection": "SNAPSHOT",  
    "status": "IN_PROGRESS",  
    "targets": [  
      "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi"  
    ],  
    "description": "example job test",  
    "presignedUrlConfig": {},  
    "jobExecutionsRolloutConfig": {},  
    "createdAt": 1560787022.733,  
    "lastUpdatedAt": 1560787026.294,  
    "jobProcessDetails": {  
      "numberOfCanceledThings": 0,  
      "numberOfSucceededThings": 0,  
      "numberOfFailedThings": 0,  
      "numberOfRejectedThings": 0,  
      "numberOfQueuedThings": 1,  
      "numberOfInProgressThings": 0,  
      "numberOfRemovedThings": 0,  
      "numberOfTimedOutThings": 0  
    },  
  },  
}
```

```
    "timeoutConfig": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DescribeJob](#) di Referensi AWS CLI Perintah.

## **describe-mitigation-action**

Contoh kode berikut menunjukkan cara menggunakan `describe-mitigation-action`.

### AWS CLI

Untuk melihat detail untuk tindakan mitigasi yang ditentukan

`describe-mitigation-action` Contoh berikut menampilkan rincian untuk tindakan mitigasi tertentu.

```
aws iot describe-mitigation-action \
  --action-name AddThingsToQuarantineGroupAction
```

### Output:

```
{
  "actionName": "AddThingsToQuarantineGroupAction",
  "actionType": "ADD_THINGS_TO_THING_GROUP",
  "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
  "actionId": "2fd2726d-98e1-4abf-b10f-09465ccd6bfa",
  "roleArn": "arn:aws:iam::123456789012:role/service-role/
MoveThingsToQuarantineGroupRole",
  "actionParams": {
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "QuarantineGroup1"
      ],
      "overrideDynamicGroups": true
    }
  },
  "creationDate": "2019-12-10T11:09:35.999000-08:00",
```

```
"lastModifiedDate": "2019-12-10T11:09:35.999000-08:00"
}
```

Untuk informasi selengkapnya, lihat [DescribeMitigationAction \(Perintah Tindakan Mitigasi\)](#) di [Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeMitigationAction](#) di Referensi AWS CLI Perintah.

## describe-provisioning-template-version

Contoh kode berikut menunjukkan cara menggunakan `describe-provisioning-template-version`.

### AWS CLI

Untuk mendeskripsikan versi template penyediaan

`describe-provisioning-template-version` Contoh berikut menjelaskan versi template penyediaan.

```
aws iot describe-provisioning-template-version \
  --template-name MyTestProvisioningTemplate \
  --version-id 1
```

Output:

```
{
  "versionId": 1,
  "creationDate": 1589308310.574,
  "templateBody": "{
    \"Parameters\":{
      \"SerialNumber\":{
        \"Type\":\"String\"
      },
      \"AWS::IoT::Certificate::Id\":{
        \"Type\":\"String\"
      }
    },
    \"Resources\":{
      \"certificate\":{
        \"Properties\":{
```

```

        \\"CertificateId\\":{
            \\"Ref\\":\\"AWS::IoT::Certificate::Id\\"
        },
        \\"Status\\":\\"Active\\"
    },
    \\"Type\\":\\"AWS::IoT::Certificate\\"
},
\\"policy\\":{
    \\"Properties\\":{
        \\"PolicyName\\":\\"MyIotPolicy\\"
    },
    \\"Type\\":\\"AWS::IoT::Policy\\"
},
\\"thing\\":{
    \\"OverrideSettings\\":{
        \\"AttributePayload\\":\\"MERGE\\",
        \\"ThingGroups\\":\\"DO_NOTHING\\",
        \\"ThingTypeName\\":\\"REPLACE\\"
    },
    \\"Properties\\":{
        \\"AttributePayload\\":{},
        \\"ThingGroups\\":[],
        \\"ThingName\\":{
            \\"Fn::Join\\":[
                \\"\\",
                [
                    \\"DemoGroup_\\",
                    {\\"Ref\\":\\"SerialNumber\\"}
                ]
            ]
        },
        \\"ThingTypeName\\":\\"VirtualThings\\"
    },
    \\"Type\\":\\"AWS::IoT::Thing\\"
}
}
},
"setDefaultVersion": true
}

```

Untuk informasi selengkapnya, lihat [Menyediakan perangkat yang tidak memiliki sertifikat perangkat menggunakan penyediaan armada di Panduan Pengembang Inti IoT AWS](#) .

- Untuk detail API, lihat [DescribeProvisioningTemplateVersion](#) di Referensi AWS CLI Perintah.

## describe-provisioning-template

Contoh kode berikut menunjukkan cara menggunakan `describe-provisioning-template`.

### AWS CLI

Untuk mendeskripsikan template penyediaan

`describe-provisioning-template` Contoh berikut menjelaskan template penyediaan.

```
aws iot describe-provisioning-template \  
  --template-name MyTestProvisioningTemplate
```

### Output:

```
{  
  "templateArn": "arn:aws:iot:us-west-2:57EXAMPLE833:provisioningtemplate/  
MyTestProvisioningTemplate",  
  "templateName": "MyTestProvisioningTemplate",  
  "creationDate": 1589308310.574,  
  "lastModifiedDate": 1589308345.539,  
  "defaultVersionId": 1,  
  "templateBody": "{  
    \"Parameters\":{  
      \"SerialNumber\":{  
        \"Type\":\"String\"  
      },  
      \"AWS::IoT::Certificate::Id\":{  
        \"Type\":\"String\"  
      }  
    },  
    \"Resources\":{  
      \"certificate\":{  
        \"Properties\":{  
          \"CertificateId\":{  
            \"Ref\":\"AWS::IoT::Certificate::Id\"  
          },  
          \"Status\":\"Active\"  
        },  
        \"Type\":\"AWS::IoT::Certificate\"  
      },  
      \"policy\":{  
        \"Properties\":{  
          \"PolicyName\":\"MyIotPolicy\"  
        }  
      }  
    }  
  }"
```

```

    },
    \"Type\": \"AWS::IoT::Policy\"
  },
  \"thing\": {
    \"OverrideSettings\": {
      \"AttributePayload\": \"MERGE\",
      \"ThingGroups\": \"DO_NOTHING\",
      \"ThingTypeName\": \"REPLACE\"
    },
    \"Properties\": {
      \"AttributePayload\": {},
      \"ThingGroups\": [],
      \"ThingName\": {
        \"Fn::Join\": [
          \"\",
          [
            \"DemoGroup_\",
            {\"Ref\": \"SerialNumber\"}
          ]
        ]
      },
      \"ThingTypeName\": \"VirtualThings\"
    },
    \"Type\": \"AWS::IoT::Thing\"
  }
}
},
\"enabled\": true,
\"provisioningRoleArn\": \"arn:aws:iam::571032923833:role/service-role/IoT_access\"
}

```

Untuk informasi selengkapnya, lihat [Menyediakan perangkat yang tidak memiliki sertifikat perangkat menggunakan penyediaan armada di](#) Panduan Pengembang Inti IoT AWS .

- Untuk detail API, lihat [DescribeProvisioningTemplate](#) di Referensi AWS CLI Perintah.

## describe-role-alias

Contoh kode berikut menunjukkan cara menggunakan `describe-role-alias`.

### AWS CLI

Untuk mendapatkan informasi tentang AWS alias peran IoT



`describe-role-alias` Contoh berikut menampilkan rincian untuk alias peran tertentu.

```
aws iot describe-role-alias \  
  --role-alias LightBulbRole
```

Output:

```
{  
  "roleAliasDescription": {  
    "roleAlias": "LightBulbRole",  
    "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/  
LightBulbRole",  
    "roleArn": "arn:aws:iam::123456789012:role/light_bulb_role_001",  
    "owner": "123456789012",  
    "credentialDurationSeconds": 3600,  
    "creationDate": 1570558643.221,  
    "lastModifiedDate": 1570558643.221  
  }  
}
```

Untuk informasi selengkapnya, lihat [DescribeRoleAlias](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DescribeRoleAlias](#) di Referensi AWS CLI Perintah.

## **describe-scheduled-audit**

Contoh kode berikut menunjukkan cara menggunakan `describe-scheduled-audit`.

### AWS CLI

Untuk mendapatkan informasi tentang audit terjadwal

`describe-scheduled-audit` Contoh berikut mendapatkan informasi rinci tentang audit terjadwal AWS IoT Device Defender bernama `AWSIoTDeviceDefenderDailyAudit`

```
aws iot describe-scheduled-audit \  
  --scheduled-audit-name AWSIoTDeviceDefenderDailyAudit
```

Output:

```
{  
  "frequency": "DAILY",
```

```
"targetCheckNames": [
  "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK",
  "CONFLICTING_CLIENT_IDS_CHECK",
  "DEVICE_CERTIFICATE_SHARED_CHECK",
  "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",
  "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK",
  "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK"
],
"scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
"scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
AWSIoTDeviceDefenderDailyAudit"
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeScheduledAudit](#) di Referensi AWS CLI Perintah.

## describe-security-profile

Contoh kode berikut menunjukkan cara menggunakan `describe-security-profile`.

### AWS CLI

Untuk mendapatkan informasi tentang profil keamanan

`describe-security-profile` Contoh berikut mendapatkan informasi tentang profil keamanan AWS IoT Device Defender bernama `PossibleIssue`.

```
aws iot describe-security-profile \
  --security-profile-name PossibleIssue
```

Output:

```
{
  "securityProfileName": "PossibleIssue",
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
PossibleIssue",
  "securityProfileDescription": "check to see if authorization fails 10 times in 5
minutes or if cellular bandwidth exceeds 128",
  "behaviors": [
    {
      "name": "CellularBandwidth",
      "metric": "aws:message-byte-size",
```

```

        "criteria": {
            "comparisonOperator": "greater-than",
            "value": {
                "count": 128
            },
            "consecutiveDatapointsToAlarm": 1,
            "consecutiveDatapointsToClear": 1
        }
    },
    {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
            "comparisonOperator": "greater-than",
            "value": {
                "count": 10
            },
            "durationSeconds": 300,
            "consecutiveDatapointsToAlarm": 1,
            "consecutiveDatapointsToClear": 1
        }
    }
],
"version": 1,
"creationDate": 1560278102.528,
"lastModifiedDate": 1560278102.528
}

```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeSecurityProfile](#) di Referensi AWS CLI Perintah.

## describe-stream

Contoh kode berikut menunjukkan cara menggunakan `describe-stream`.

### AWS CLI

Untuk mendapatkan informasi tentang aliran

`describe-stream` Contoh berikut menampilkan rincian tentang aliran yang ditentukan.

```
aws iot describe-stream \
```

```
--stream-id stream12345
```

### Output:

```
{
  "streamInfo": {
    "streamId": "stream12345",
    "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
    "streamVersion": 1,
    "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
    "files": [
      {
        "fileId": "123",
        "s3Location": {
          "bucket": "codesign-ota-bucket",
          "key": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
        }
      }
    ],
    "createdAt": 1557863215.995,
    "lastUpdatedAt": 1557863215.995,
    "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_stream_role"
  }
}
```

Untuk informasi selengkapnya, lihat [DescribeStream](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DescribeStream](#) di Referensi AWS CLI Perintah.

## describe-thing-group

Contoh kode berikut menunjukkan cara menggunakan `describe-thing-group`.

### AWS CLI

Untuk mendapatkan informasi tentang grup sesuatu

`describe-thing-group` Contoh berikut mendapat informasi tentang kelompok hal bernama `HalogenBulbs`.

```
aws iot describe-thing-group \
  --thing-group-name HalogenBulbs
```

**Output:**

```
{
  "thingGroupName": "HalogenBulbs",
  "thingGroupId": "f4ec6b84-b42b-499d-9ce1-4dbd4d4f6f6e",
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs",
  "version": 1,
  "thingGroupProperties": {},
  "thingGroupMetadata": {
    "parentGroupName": "LightBulbs",
    "rootToParentThingGroups": [
      {
        "groupName": "LightBulbs",
        "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
LightBulbs"
      }
    ],
    "creationDate": 1559927609.897
  }
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeThingGroup](#) di Referensi AWS CLI Perintah.

**describe-thing-type**

Contoh kode berikut menunjukkan cara menggunakan `describe-thing-type`.

**AWS CLI**

Untuk mendapatkan informasi tentang tipe benda

`describe-thing-type` Contoh berikut menampilkan informasi tentang jenis hal yang ditentukan ditentukan dalam AWS akun Anda.

```
aws iot describe-thing-type \
  --thing-type-name "LightBulb"
```

**Output:**

```
{
```

```

"thingTypeName": "LightBulb",
"thingTypeId": "ce3573b0-0a3c-45a7-ac93-4e0ce14cd190",
"thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",
"thingTypeProperties": {
  "thingTypeDescription": "light bulb type",
  "searchableAttributes": [
    "model",
    "wattage"
  ]
},
"thingTypeMetadata": {
  "deprecated": false,
  "creationDate": 1559772562.498
}
}

```

Untuk informasi selengkapnya, lihat [Tipe Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeThingType](#) di Referensi AWS CLI Perintah.

## describe-thing

Contoh kode berikut menunjukkan cara menggunakan `describe-thing`.

### AWS CLI

Untuk menampilkan informasi rinci tentang suatu hal

`describe-thing` Contoh berikut menampilkan informasi tentang sesuatu (perangkat) yang didefinisikan dalam registri AWS IoT untuk akun Anda AWS .

```
aws iot mendeskripsikan-hal --thing-name "" MyLightBulb
```

Output:

```

{
  "defaultClientId": "MyLightBulb",
  "thingName": "MyLightBulb",
  "thingId": "40da2e73-c6af-406e-b415-15acae538797",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
  "thingTypeName": "LightBulb",
  "attributes": {
    "model": "123",

```

```
    "wattage": "75"  
  },  
  "version": 1  
}
```

Untuk informasi selengkapnya, lihat [Cara Mengelola Sesuatu dengan Registri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeThing](#) di Referensi AWS CLI Perintah.

## detach-policy

Contoh kode berikut menunjukkan cara menggunakan `detach-policy`.

### AWS CLI

Contoh 1: Untuk melepaskan kebijakan AWS IoT dari grup benda

`detach-policy` Contoh berikut melepaskan kebijakan yang ditentukan dari grup benda dan, dengan ekstensi, dari semua hal dalam grup itu dan salah satu grup anak grup.

```
aws iot detach-policy \  
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \  
  --policy-name "MyFirstGroup_Core-policy"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

Contoh 2: Untuk melepaskan kebijakan AWS IoT dari sertifikat perangkat

`detach-policy` Contoh berikut melepaskan `TemperatureSensorPolicy` kebijakan dari sertifikat perangkat yang diidentifikasi oleh ARN.

```
aws iot detach-policy \  
  --policy-name TemperatureSensorPolicy \  
  --target arn:aws:iot:us-  
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DetachPolicy](#) di Referensi AWS CLI Perintah.

## detach-security-profile

Contoh kode berikut menunjukkan cara menggunakan `detach-security-profile`.

### AWS CLI

Untuk memisahkan profil keamanan dari target

`detach-security-profile` Contoh berikut menghapus hubungan antara `Testprofile` profil keamanan AWS IoT Device Defender yang bernama dan target semua hal yang terdaftar.

```
aws iot detach-security-profile \  
  --security-profile-name Testprofile \  
  --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/  
registered-things"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DetachSecurityProfile](#) di Referensi AWS CLI Perintah.

## detach-thing-principal

Contoh kode berikut menunjukkan cara menggunakan `detach-thing-principal`.

### AWS CLI

Untuk melepaskan sertifikat/prinsipal dari suatu hal

`detach-thing-principal` Contoh berikut menghapus sertifikat yang mewakili prinsipal dari hal yang ditentukan.

```
aws iot detach-thing-principal \  
  --thing-name "MyLightBulb" \  
  --principal "arn:aws:iot:us-  
west-2:123456789012:cert/604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Mengelola Sesuatu dengan Registri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DetachThingPrincipal](#) di Referensi AWS CLI Perintah.



## **disable-topic-rule**

Contoh kode berikut menunjukkan cara menggunakan `disable-topic-rule`.

### AWS CLI

Untuk menonaktifkan aturan topik

`disable-topic-rule` Contoh berikut menonaktifkan aturan topik yang ditentukan.

```
aws iot disable-topic-rule \  
  --rule-name "MyPlantPiMoistureAlertRule"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Melihat Aturan Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DisableTopicRule](#) di Referensi AWS CLI Perintah.

## **enable-topic-rule**

Contoh kode berikut menunjukkan cara menggunakan `enable-topic-rule`.

### AWS CLI

Untuk mengaktifkan aturan topik

`enable-topic-rule` Contoh berikut memungkinkan (atau mengaktifkan kembali) aturan topik yang ditentukan.

```
aws iot enable-topic-rule \  
  --rule-name "MyPlantPiMoistureAlertRule"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Melihat Aturan Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [EnableTopicRule](#) di Referensi AWS CLI Perintah.

## **get-behavior-model-training-summaries**

Contoh kode berikut menunjukkan cara menggunakan `get-behavior-model-training-summaries`.

## AWS CLI

Untuk mencantumkan status model pelatihan Profil Keamanan Deteksi Defender dari Device Defender

`get-behavior-model-training-summaries` Contoh berikut mencantumkan status pelatihan model untuk perilaku yang dikonfigurasi dalam Profil Keamanan yang dipilih. Untuk setiap perilaku, nama, status model, dan persentase titik data yang dikumpulkan dicantumkan.

```
aws iot get-behavior-model-training-summaries \  
  --security-profile-name MySecuirtyProfileName
```

Output:

```
{  
  "summaries": [  
    {  
      "securityProfileName": "MySecuirtyProfileName",  
      "behaviorName": "Messages_sent_ML_behavior",  
      "modelStatus": "PENDING_BUILD",  
      "datapointsCollectionPercentage": 0.0  
    },  
    {  
      "securityProfileName": "MySecuirtyProfileName",  
      "behaviorName": "Messages_received_ML_behavior",  
      "modelStatus": "PENDING_BUILD",  
      "datapointsCollectionPercentage": 0.0  
    },  
    {  
      "securityProfileName": "MySecuirtyProfileName",  
      "behaviorName": "Authorization_failures_ML_behavior",  
      "modelStatus": "PENDING_BUILD",  
      "datapointsCollectionPercentage": 0.0  
    },  
    {  
      "securityProfileName": "MySecuirtyProfileName",  
      "behaviorName": "Message_size_ML_behavior",  
      "modelStatus": "PENDING_BUILD",  
      "datapointsCollectionPercentage": 0.0  
    },  
    {  
      "securityProfileName": "MySecuirtyProfileName",  
      "behaviorName": "Connection_attempts_ML_behavior",  
      "modelStatus": "PENDING_BUILD",  
      "datapointsCollectionPercentage": 0.0  
    }  
  ]  
}
```

```

        "modelStatus": "PENDING_BUILD",
        "datapointsCollectionPercentage": 0.0
    },
    {
        "securityProfileName": "MySPNoAlerts",
        "behaviorName": "Disconnects_ML_behavior",
        "modelStatus": "PENDING_BUILD",
        "datapointsCollectionPercentage": 0.0
    }
]
}

```

Untuk informasi selengkapnya, lihat [GetBehaviorModelTrainingSummaries \(Mendeteksi Perintah\)](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetBehaviorModelTrainingSummaries](#) di Referensi AWS CLI Perintah.

## get-cardinality

Contoh kode berikut menunjukkan cara menggunakan `get-cardinality`.

### AWS CLI

Untuk mengembalikan perkiraan jumlah nilai unik yang cocok dengan kueri

Anda dapat menggunakan skrip pengaturan berikut untuk membuat 10 hal yang mewakili 10 sensor suhu. Setiap hal baru memiliki 3 atribut.

```

# Bash script. If in other shells, type `bash` before running
Temperatures=(70 71 72 73 74 75 47 97 98 99)
Racks=(Rack1 Rack1 Rack2 Rack2 Rack3 Rack4 Rack5 Rack6 Rack6 Rack6)
IsNormal=(true true true true true true false false false false)
for ((i=0; i<10 ; i++))
do
    thing=$(aws iot create-thing --thing-name "TempSensor$i" --attribute-payload
attributes="{temperature=${Temperatures[i]},rackId=${Racks[i]},stateNormal=
${IsNormal[i]}}")
    aws iot describe-thing --thing-name "TempSensor$i"
done

```

Contoh output dari skrip pengaturan:

```
{
```

```

    "version": 1,
    "thingName": "TempSensor0",
    "defaultClientId": "TempSensor0",
    "attributes": {
      "rackId": "Rack1",
      "stateNormal": "true",
      "temperature": "70"
    },
    "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/TempSensor0",
    "thingId": "example1-90ab-cdef-fedc-ba987example"
  }

```

`get-cardinality` Contoh berikut menanyakan 10 sensor yang dibuat oleh skrip pengaturan dan mengembalikan jumlah rak yang memiliki sensor suhu yang melaporkan nilai suhu abnormal. Jika nilai suhu di bawah 60 atau di atas 80, sensor suhu dalam keadaan abnormal.

```

aws iot get-cardinality \
  --aggregation-field "attributes.rackId" \
  --query-string "thingName:TempSensor* AND attributes.stateNormal:false"

```

Output:

```

{
  "cardinality": 2
}

```

Untuk informasi selengkapnya, lihat Menanyakan Data Agregat < <https://docs.aws.amazon.com/iot/latest/developerguide/index-aggregate.html> > di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetCardinality](#) di Referensi AWS CLI Perintah.

## get-effective-policies

Contoh kode berikut menunjukkan cara menggunakan `get-effective-policies`.

### AWS CLI

Untuk membuat daftar kebijakan yang mempengaruhi sesuatu

`get-effective-policies` Contoh berikut mencantumkan kebijakan yang mempengaruhi hal yang ditentukan, termasuk kebijakan yang dilampirkan pada grup mana pun yang menjadi miliknya.

```
aws iot get-effective-policies \
  --thing-name TemperatureSensor-001 \
  --principal arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

### Output:

```
{
  "effectivePolicies": [
    {
      "policyName": "TemperatureSensorPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy",
      "policyDocument": "{
        \"Version\": \"2012-10-17\",
        \"Statement\": [
          {
            \"Effect\": \"Allow\",
            \"Action\": [
              \"iot:Publish\",
              \"iot:Receive\"
            ],
            \"Resource\": [
              \"arn:aws:iot:us-west-2:123456789012:topic/topic_1\",
              \"arn:aws:iot:us-west-2:123456789012:topic/topic_2\"
            ]
          },
          {
            \"Effect\": \"Allow\",
            \"Action\": [
              \"iot:Subscribe\"
            ],
            \"Resource\": [
              \"arn:aws:iot:us-west-2:123456789012:topicfilter/
topic_1\",
              \"arn:aws:iot:us-west-2:123456789012:topicfilter/
topic_2\"
            ]
          },
          {
            \"Effect\": \"Allow\",
            \"Action\": [
              \"iot:Connect\"
            ],
            \"Resource\": [
              \"arn:aws:iot:us-west-2:123456789012:device/
device_1\",
              \"arn:aws:iot:us-west-2:123456789012:device/
device_2\"
            ]
          }
        ]
      }
    }
  ]
}
```

```

    ],
    \"Resource\": [
      \"arn:aws:iot:us-west-2:123456789012:client/basicPubSub
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mendapatkan Kebijakan Efektif untuk Sesuatu](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetEffectivePolicies](#) di Referensi AWS CLI Perintah.

## get-indexing-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-indexing-configuration`.

### AWS CLI

Untuk mendapatkan konfigurasi pengindeksan hal

`get-indexing-configuration` Contoh berikut mendapatkan data konfigurasi saat ini untuk pengindeksan armada AWS IoT.

```
aws iot get-indexing-configuration
```

Output:

```

{
  "thingIndexingConfiguration": {
    "thingIndexingMode": "OFF",
    "thingConnectivityIndexingMode": "OFF"
  },
  "thingGroupIndexingConfiguration": {
    "thingGroupIndexingMode": "OFF"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola Pengindeksan Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetIndexingConfiguration](#) di Referensi AWS CLI Perintah.

## get-job-document

Contoh kode berikut menunjukkan cara menggunakan `get-job-document`.

### AWS CLI

Untuk mengambil dokumen untuk pekerjaan

`get-job-document` Contoh berikut menampilkan rincian tentang dokumen untuk pekerjaan yang ID adalah `example-job-01`.

```
aws iot get-job-document \
  --job-id "example-job-01"
```

Output:

```
{
  "document": "\n{\n  \"operation\": \"customJob\", \n  \"otherInfo\":\n  \"someValue\"\n}\n"
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetJobDocument](#) di Referensi AWS CLI Perintah.

## get-logging-options

Contoh kode berikut menunjukkan cara menggunakan `get-logging-options`.

### AWS CLI

Untuk mendapatkan opsi logging

`get-logging-options` Contoh berikut mendapatkan opsi logging saat ini untuk AWS akun Anda.

```
aws iot get-logging-options
```

Output:

```
{
  "roleArn": "arn:aws:iam::123456789012:role/service-role/iotLoggingRole",
  "logLevel": "ERROR"
}
```

Untuk informasi selengkapnya, lihat judul di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetLoggingOptions](#) di Referensi AWS CLI Perintah.

## get-ota-update

Contoh kode berikut menunjukkan cara menggunakan `get-ota-update`.

### AWS CLI

Untuk mengambil informasi tentang Pembaruan OTA

`get-ota-update` Contoh berikut menampilkan detail tentang Pembaruan OTA yang ditentukan.

```
aws iot get-ota-update \
  --ota-update-id ota12345
```

Output:

```
{
  "otaUpdateInfo": {
    "otaUpdateId": "ota12345",
    "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",
    "creationDate": 1557863215.995,
    "lastModifiedDate": 1557863215.995,
    "description": "A critical update needed right away.",
    "targets": [
      "device1",
      "device2",
      "device3",
      "device4"
    ]
  },
}
```



```

    "targetSelection": "SNAPSHOT",
    "protocols": ["HTTP"],
    "awsJobExecutionsRolloutConfig": {
      "maximumPerMinute": 10
    },
    "otaUpdateFiles": [
      {
        "fileName": "firmware.bin",
        "fileLocation": {
          "stream": {
            "streamId": "004",
            "fileId": 123
          }
        },
        "codeSigning": {
          "awsSignerJobId": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
        }
      }
    ],
    "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_role"
  },
  "otaUpdateStatus": "CREATE_COMPLETE",
  "awsIotJobId": "job54321",
  "awsIotJobArn": "arn:aws:iot:us-west-2:123456789012:job/job54321",
  "errorInfo": {
  }
}
}

```

Untuk informasi selengkapnya, lihat [getOtaUpdate](#) di Referensi API IoT.AWS

- Untuk detail API, lihat [GetOtaUpdated](#) di Referensi AWS CLI Perintah.

## get-percentiles

Contoh kode berikut menunjukkan cara menggunakan `get-percentiles`.

### AWS CLI

Untuk mengelompokkan nilai agregat yang cocok dengan kueri ke dalam pengelompokan persentil

Anda dapat menggunakan skrip pengaturan berikut untuk membuat 10 hal yang mewakili 10 sensor suhu. Setiap hal baru memiliki 1 atribut.

```
# Bash script. If in other shells, type `bash` before running
Temperatures=(70 71 72 73 74 75 47 97 98 99)
for ((i=0; i<10 ; i++))
do
    thing=$(aws iot create-thing --thing-name "TempSensor$i" --attribute-payload
attributes="{temperature=${Temperatures[i]}}")
    aws iot describe-thing --thing-name "TempSensor$i"
done
```

Contoh output dari skrip pengaturan:

```
{
  "version": 1,
  "thingName": "TempSensor0",
  "defaultClientId": "TempSensor0",
  "attributes": {
    "temperature": "70"
  },
  "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/TempSensor0",
  "thingId": "example1-90ab-cdef-fedc-ba987example"
}
```

`get-percentiles` Contoh berikut menanyakan 10 sensor yang dibuat oleh skrip pengaturan dan mengembalikan nilai untuk setiap grup persentil yang ditentukan. Grup persentil "10" berisi nilai bidang agregat yang terjadi di sekitar 10 persen dari nilai yang cocok dengan kueri. Dalam output berikut, {"persen": 10,0, "nilai": 67,7} berarti sekitar 10,0% dari nilai suhu di bawah 67,7.

```
aws iot get-percentiles \
  --aggregation-field "attributes.temperature" \
  --query-string "thingName:TempSensor*" \
  --percents 10 25 50 75 90
```

Output:

```
{
  "percentiles": [
    {
      "percent": 10.0,
      "value": 67.7
    },
    {
```

```
    "percent": 25.0,  
    "value": 71.25  
  },  
  {  
    "percent": 50.0,  
    "value": 73.5  
  },  
  {  
    "percent": 75.0,  
    "value": 91.5  
  },  
  {  
    "percent": 90.0,  
    "value": 98.1  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Menanyakan Data Agregat di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [GetPercentiles](#) di Referensi AWS CLI Perintah.

## get-policy-version

Contoh kode berikut menunjukkan cara menggunakan `get-policy-version`.

### AWS CLI

Untuk mendapatkan informasi tentang versi kebijakan tertentu

`get-policy-version` Contoh berikut mendapatkan informasi tentang versi pertama dari kebijakan yang ditentukan.

```
aws iot get-policy \  
  --policy-name UpdateDeviceCertPolicy  
  --policy-version-id "1"
```

Output:

```
{  
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
```

```

    "policyName": "UpdateDeviceCertPolicy",
    "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
  \"Allow\", \"Action\": \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
    "policyVersionId": "1",
    "isDefaultVersion": false,
    "creationDate": 1559925941.924,
    "lastModifiedDate": 1559926175.458,
    "generationId":
  "5066f1b6712ce9d2a1e56399771649a272d6a921762fead080e24fe52f24e042"
  }

```

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [GetPolicyVersion](#) di Referensi AWS CLI Perintah.

## get-policy

Contoh kode berikut menunjukkan cara menggunakan `get-policy`.

### AWS CLI

Untuk mendapatkan informasi tentang versi default kebijakan

`get-policy` Contoh berikut mengambil informasi tentang versi default dari kebijakan yang ditentukan.

```

aws iot get-policy \
  --policy-name UpdateDeviceCertPolicy

```

Output:

```

{
  "policyName": "UpdateDeviceCertPolicy",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
  "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
  \"Allow\", \"Action\": \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
  "defaultVersionId": "2",
  "creationDate": 1559925941.924,
  "lastModifiedDate": 1559925941.924,
  "generationId":
  "5066f1b6712ce9d2a1e56399771649a272d6a921762fead080e24fe52f24e042"
}

```

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetPolicy](#) di Referensi AWS CLI Perintah.

## get-registration-code

Contoh kode berikut menunjukkan cara menggunakan `get-registration-code`.

### AWS CLI

Untuk mendapatkan kode pendaftaran AWS khusus akun Anda

`get-registration-code` Contoh berikut mengambil kode pendaftaran AWS khusus akun Anda.

```
aws iot get-registration-code
```

Output:

```
{
  "registrationCode":
  "15c51ae5e36ba59ba77042df1115862076bea4bd15841c838fcb68d5010a614c"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Sertifikat Anda Sendiri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetRegistrationCode](#) di Referensi AWS CLI Perintah.

## get-statistics

Contoh kode berikut menunjukkan cara menggunakan `get-statistics`.

### AWS CLI

Untuk mencari indeks perangkat untuk data agregat

`get-statistics` Contoh berikut mengembalikan jumlah hal yang memiliki properti yang disebut `connectivity.connected` set ke `false` (yaitu, jumlah perangkat yang tidak terhubung) dalam bayangan perangkat mereka.

```
aws iot get-statistics \
```

```
--index-name AWS_Things \  
--query-string "connectivity.connected:false"
```

Output:

```
{  
  "statistics": {  
    "count": 6  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan Statistik Tentang Armada Perangkat Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetStatistics](#) di Referensi AWS CLI Perintah.

## get-topic-rule-destination

Contoh kode berikut menunjukkan cara menggunakan `get-topic-rule-destination`.

AWS CLI

Untuk mendapatkan tujuan aturan topik

`get-topic-rule-destination` Contoh berikut mendapatkan informasi tentang tujuan aturan topik.

```
aws iot get-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Output:

```
{  
  "topicRuleDestination": {  
    "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "status": "DISABLED",  
    "httpUrlProperties": {  
      "confirmationUrl": "https://example.com"  
    }  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tujuan aturan topik](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetTopicRuleDestination](#) di Referensi AWS CLI Perintah.

## get-topic-rule

Contoh kode berikut menunjukkan cara menggunakan `get-topic-rule`.

### AWS CLI

Untuk mendapatkan informasi tentang aturan

`get-topic-rule` Contoh berikut mendapat informasi tentang aturan yang ditentukan.

```
aws iot get-topic-rule \  
  --rule-name MyRPiLowMoistureAlertRule
```

Output:

```
{  
  "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/MyRPiLowMoistureAlertRule",  
  "rule": {  
    "ruleName": "MyRPiLowMoistureAlertRule",  
    "sql": "SELECT * FROM '$aws/things/MyRPi/shadow/update/accepted' WHERE  
state.reported.moisture = 'low'\n          ",  
    "description": "Sends an alert whenever soil moisture level readings are too  
low.",  
    "createdAt": 1558624363.0,  
    "actions": [  
      {  
        "sns": {  
          "targetArn": "arn:aws:sns:us-  
west-2:123456789012:MyRPiLowMoistureTopic",  
          "roleArn": "arn:aws:iam::123456789012:role/service-role/  
MyRPiLowMoistureTopicRole",  
          "messageFormat": "RAW"  
        }  
      }  
    ],  
    "ruleDisabled": false,  
  }  
}
```

```
    "awsIotSqlVersion": "2016-03-23"
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat Aturan Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetTopicRule](#) di Referensi AWS CLI Perintah.

## get-v2-logging-options

Contoh kode berikut menunjukkan cara menggunakan `get-v2-logging-options`.

### AWS CLI

Untuk membuat daftar opsi logging saat ini

`get-v2-logging-options` Contoh berikut mencantumkan opsi logging saat ini untuk AWS IoT.

```
aws iot get-v2-logging-options
```

Output:

```
{
  "roleArn": "arn:aws:iam::094249569039:role/service-role/iotLoggingRole",
  "defaultLogLevel": "WARN",
  "disableAllLogs": false
}
```

Untuk informasi selengkapnya, lihat judul di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetV2 LoggingOptions](#) di Referensi AWS CLI Perintah.

## list-active-violations

Contoh kode berikut menunjukkan cara menggunakan `list-active-violations`.

### AWS CLI

Untuk daftar pelanggaran aktif

`list-active-violations` Contoh berikut mencantumkan semua pelanggaran untuk profil keamanan yang ditentukan.



```
aws iot list-active-violations \  
--security-profile-name Testprofile
```

**Output:**

```
{  
  "activeViolations": [  
    {  
      "violationId": "174db59167fa474c80a652ad1583fd44",  
      "thingName": "iotconsole-1560269126751-1",  
      "securityProfileName": "Testprofile",  
      "behavior": {  
        "name": "Authorization",  
        "metric": "aws:num-authorization-failures",  
        "criteria": {  
          "comparisonOperator": "greater-than",  
          "value": {  
            "count": 10  
          },  
          "durationSeconds": 300,  
          "consecutiveDatapointsToAlarm": 1,  
          "consecutiveDatapointsToClear": 1  
        }  
      },  
      "lastViolationValue": {  
        "count": 0  
      },  
      "lastViolationTime": 1560293700.0,  
      "violationStartTime": 1560279000.0  
    },  
    {  
      "violationId": "c8a9466a093d3b7b35cd44ca58bdbeab",  
      "thingName": "TvnQoEoU",  
      "securityProfileName": "Testprofile",  
      "behavior": {  
        "name": "CellularBandwidth",  
        "metric": "aws:message-byte-size",  
        "criteria": {  
          "comparisonOperator": "greater-than",  
          "value": {  
            "count": 128  
          },  
          "consecutiveDatapointsToAlarm": 1,  
          "consecutiveDatapointsToClear": 1  
        }  
      }  
    }  
  ]  
}
```

```
        "consecutiveDatapointsToClear": 1
      }
    },
    "lastViolationValue": {
      "count": 110
    },
    "lastViolationTime": 1560369000.0,
    "violationStartTime": 1560276600.0
  },
  {
    "violationId": "74aa393adea02e6648f3ac362beed55e",
    "thingName": "iotconsole-1560269232412-2",
    "securityProfileName": "Testprofile",
    "behavior": {
      "name": "Authorization",
      "metric": "aws:num-authorization-failures",
      "criteria": {
        "comparisonOperator": "greater-than",
        "value": {
          "count": 10
        }
      },
      "durationSeconds": 300,
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1
    }
  },
  "lastViolationValue": {
    "count": 0
  },
  "lastViolationTime": 1560276600.0,
  "violationStartTime": 1560276600.0
},
{
  "violationId": "1e6ab5f7cf39a1466fcd154e1377e406",
  "thingName": "TvnQoEoU",
  "securityProfileName": "Testprofile",
  "behavior": {
    "name": "Authorization",
    "metric": "aws:num-authorization-failures",
    "criteria": {
      "comparisonOperator": "greater-than",
      "value": {
        "count": 10
      }
    }
  },
}
```

```

        "durationSeconds": 300,
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
      }
    },
    "lastViolationValue": {
      "count": 0
    },
    "lastViolationTime": 1560369000.0,
    "violationStartTime": 1560276600.0
  }
]
}

```

- Untuk detail API, lihat [ListActiveViolations](#) di Referensi AWS CLI Perintah.

## list-attached-policies

Contoh kode berikut menunjukkan cara menggunakan `list-attached-policies`.

### AWS CLI

Contoh 1: Untuk mencantumkan kebijakan yang dilampirkan ke grup

`list-attached-policies` Contoh berikut mencantumkan kebijakan yang dilampirkan ke grup tertentu.

```

aws iot list-attached-policies \
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"

```

Output:

```

{
  "policies": [
    {
      "policyName": "UpdateDeviceCertPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
UpdateDeviceCertPolicy"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

Contoh 2: Untuk mencantumkan kebijakan yang dilampirkan pada sertifikat perangkat

`list-attached-policies` Contoh berikut mencantumkan kebijakan AWS IoT yang dilampirkan pada sertifikat perangkat. Sertifikat diidentifikasi oleh ARN-nya.

```
aws iot list-attached-policies \  
  --target arn:aws:iot:us-  
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Output:

```
{  
  "policies": [  
    {  
      "policyName": "TemperatureSensorPolicy",  
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/  
TemperatureSensorPolicy"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListAttachedPolicies](#) di Referensi AWS CLI Perintah.

## **list-audit-findings**

Contoh kode berikut menunjukkan cara menggunakan `list-audit-findings`.

AWS CLI

Contoh 1: Untuk membuat daftar semua temuan dari audit

`list-audit-findings` Contoh berikut mencantumkan semua temuan dari audit AWS IoT Device Defender dengan ID tugas yang ditentukan.

```
aws iot list-audit-findings \  
  --task-id a3aea009955e501a31b764abe1bebd3d
```

Output:

```
{
  "findings": []
}
```

Contoh 2: Untuk membuat daftar temuan untuk jenis pemeriksaan audit

`list-audit-findings` Contoh berikut menunjukkan temuan dari audit AWS IoT Device Defender yang berlangsung antara 5 Juni 2019 dan 19 Juni 2019 di mana perangkat berbagi sertifikat perangkat. Ketika Anda menentukan nama cek, Anda harus memberikan waktu mulai dan berakhir.

```
aws iot list-audit-findings \
  --check-name DEVICE_CERTIFICATE_SHARED_CHECK \
  --start-time 1559747125 \
  --end-time 1560962028
```

Output:

```
{
  "findings": [
    {
      "taskId": "eeef61068b0eb03c456d746c5a26ee04",
      "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
      "taskStartTime": 1560161017.172,
      "findingTime": 1560161017.592,
      "severity": "CRITICAL",
      "nonCompliantResource": {
        "resourceType": "DEVICE_CERTIFICATE",
        "resourceIdentifier": {
          "deviceCertificateId":
            "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
        }
      },
      "relatedResources": [
        {
          "resourceType": "CLIENT_ID",
          "resourceIdentifier": {
            "clientId": "ZipxgAII"
          },
          "additionalInfo": {
            "CONNECTION_TIME": "1560086374068"
          }
        }
      ]
    }
  ]
}
```

```
    },
    {
      "resourceType": "CLIENT_ID",
      "resourceIdentifier": {
        "clientId": "ZipxgA11"
      },
      "additionalInfo": {
        "CONNECTION_TIME": "1560081552187",
        "DISCONNECTION_TIME": "1560086371552"
      }
    },
    {
      "resourceType": "CLIENT_ID",
      "resourceIdentifier": {
        "clientId": "ZipxgA11"
      },
      "additionalInfo": {
        "CONNECTION_TIME": "1559289863631",
        "DISCONNECTION_TIME": "1560081532716"
      }
    }
  ],
  "reasonForNonCompliance": "Certificate shared by one or more devices.",
  "reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
},
{
  "taskId": "bade6b5efd2e1b1569822f6021b39cf5",
  "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
  "taskStartTime": 1559988217.27,
  "findingTime": 1559988217.655,
  "severity": "CRITICAL",
  "nonCompliantResource": {
    "resourceType": "DEVICE_CERTIFICATE",
    "resourceIdentifier": {
      "deviceCertificateId":
        "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
    }
  },
  "relatedResources": [
    {
      "resourceType": "CLIENT_ID",
      "resourceIdentifier": {
        "clientId": "xShGENLW"
      }
    }
  ],
}
```

```

        "additionalInfo": {
            "CONNECTION_TIME": "1559972350825"
        }
    },
    {
        "resourceType": "CLIENT_ID",
        "resourceIdentifier": {
            "clientId": "xShGENLW"
        },
        "additionalInfo": {
            "CONNECTION_TIME": "1559255062002",
            "DISCONNECTION_TIME": "1559972350616"
        }
    }
],
"reasonForNonCompliance": "Certificate shared by one or more devices.",
"reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
},
{
    "taskId": "c23f6233ba2d35879c4bb2810fb5ffd6",
    "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
    "taskStartTime": 1559901817.31,
    "findingTime": 1559901817.767,
    "severity": "CRITICAL",
    "nonCompliantResource": {
        "resourceType": "DEVICE_CERTIFICATE",
        "resourceIdentifier": {
            "deviceCertificateId":
"b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
        }
    },
    "relatedResources": [
        {
            "resourceType": "CLIENT_ID",
            "resourceIdentifier": {
                "clientId": "TvnQoEoU"
            },
            "additionalInfo": {
                "CONNECTION_TIME": "1559826729768"
            }
        },
        {
            "resourceType": "CLIENT_ID",
            "resourceIdentifier": {

```

```

        "clientId": "TvnQoEoU"
      },
      "additionalInfo": {
        "CONNECTION_TIME": "1559345920964",
        "DISCONNECTION_TIME": "1559826728402"
      }
    ],
    "reasonForNonCompliance": "Certificate shared by one or more devices.",
    "reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListAuditFindings](#) di Referensi AWS CLI Perintah.

## list-audit-mitigation-actions-executions

Contoh kode berikut menunjukkan cara menggunakan `list-audit-mitigation-actions-executions`.

### AWS CLI

Untuk membuat daftar rincian pelaksanaan tindakan mitigasi audit

Tugas tindakan mitigasi audit menerapkan tindakan mitigasi untuk satu atau beberapa temuan dari audit IoT Device AWS Defender. `list-audit-mitigation-actions-executions` Contoh berikut mencantumkan rincian untuk tugas tindakan mitigasi dengan yang ditentukan `taskId` dan untuk temuan yang ditentukan.

```

aws iot list-audit-mitigation-actions-executions \
  --task-id myActionsTaskId \
  --finding-id 0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464

```

### Output:

```

{
  "actionsExecutions": [
    {
      "taskId": "myActionsTaskId",

```



```

    "findingId": "0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464",
    "actionName": "ResetPolicyVersionAction",
    "actionId": "1ea0b415-bef1-4a01-bd13-72fb63c59afb",
    "status": "COMPLETED",
    "startTime": "2019-12-10T15:19:13.279000-08:00",
    "endTime": "2019-12-10T15:19:13.337000-08:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [ListAuditMitigationActionsExecutions \(Perintah Tindakan Mitigasi\) di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListAuditMitigationActionsExecutions](#) di Referensi AWS CLI Perintah.

## list-audit-mitigation-actions-tasks

Contoh kode berikut menunjukkan cara menggunakan `list-audit-mitigation-actions-tasks`.

### AWS CLI

Untuk membuat daftar tugas tindakan mitigasi audit

`list-audit-mitigation-actions-tasks` Contoh berikut mencantumkan tindakan mitigasi yang diterapkan pada temuan dalam periode waktu yang ditentukan.

```

aws iot list-audit-mitigation-actions-tasks \
  --start-time 1594157400 \
  --end-time 1594157430

```

### Output:

```

{
  "tasks": [
    {
      "taskId": "0062f2d6-3999-488f-88c7-bef005414103",
      "startTime": "2020-07-07T14:30:15.172000-07:00",
      "taskStatus": "COMPLETED"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [ListAuditMitigationActionsTasks \(Perintah Tindakan Mitigasi\) di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListAuditMitigationActionsTasks](#) di Referensi AWS CLI Perintah.

## list-audit-suppressions

Contoh kode berikut menunjukkan cara menggunakan `list-audit-suppressions`.

### AWS CLI

Untuk membuat daftar semua penekanan temuan audit

`list-audit-suppressions` Contoh berikut mencantumkan semua penekanan pencarian audit aktif.

```
aws iot list-audit-suppressions
```

Output:

```
{
  "suppressions": [
    {
      "checkName": "DEVICE_CERTIFICATE_EXPIRING_CHECK",
      "resourceIdentifier": {
        "deviceCertificateId": "c7691e<shortened>"
      },
      "expirationDate": 1597881600.0,
      "suppressIndefinitely": false
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Audit menemukan penekanan di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListAuditSuppressions](#) di Referensi AWS CLI Perintah.

## list-audit-tasks

Contoh kode berikut menunjukkan cara menggunakan `list-audit-tasks`.

## AWS CLI

Untuk membuat daftar semua temuan dari audit

`list-audit-tasks` Contoh berikut mencantumkan tugas audit yang berlangsung antara 5 Juni 2019 dan 12 Juni 2019.

```
aws iot list-audit-tasks \  
  --start-time 1559747125 \  
  --end-time 1560357228
```

Output:

```
{  
  "tasks": [  
    {  
      "taskId": "a3aea009955e501a31b764abe1bebd3d",  
      "taskStatus": "COMPLETED",  
      "taskType": "ON_DEMAND_AUDIT_TASK"  
    },  
    {  
      "taskId": "f76b4b5102b632cd9ae38a279c266da1",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    },  
    {  
      "taskId": "51d9967d9f9ff4d26529505f6d2c444a",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    },  
    {  
      "taskId": "eeef61068b0eb03c456d746c5a26ee04",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    },  
    {  
      "taskId": "041c49557b7c7b04c079a49514b55589",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    },  
    {  
      "taskId": "82c7f2afac1562d18a4560be73998acc",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    }  
  ]  
}
```

```
    "taskType": "SCHEDULED_AUDIT_TASK"
  },
  {
    "taskId": "bade6b5efd2e1b1569822f6021b39cf5",
    "taskStatus": "COMPLETED",
    "taskType": "SCHEDULED_AUDIT_TASK"
  },
  {
    "taskId": "c23f6233ba2d35879c4bb2810fb5ffd6",
    "taskStatus": "COMPLETED",
    "taskType": "SCHEDULED_AUDIT_TASK"
  },
  {
    "taskId": "ac9086b7222a2f5e2e17bb6fd30b3aeb",
    "taskStatus": "COMPLETED",
    "taskType": "SCHEDULED_AUDIT_TASK"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListAuditTasks](#) di Referensi AWS CLI Perintah.

## list-authorizers

Contoh kode berikut menunjukkan cara menggunakan `list-authorizers`.

### AWS CLI

Untuk membuat daftar otorisasi kustom Anda

`list-authorizers` Contoh berikut mencantumkan otorisasi khusus di AWS akun Anda.

```
aws iot list-authorizers
```

Output:

```
{
  "authorizers": [
    {
      "authorizerName": "CustomAuthorizer",
```

```

        "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer"
      },
      {
        "authorizerName": "CustomAuthorizer2",
        "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer2"
      },
      {
        "authorizerName": "CustomAuthorizer3",
        "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer3"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [ListAuthorizers](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListAuthorizers](#) di Referensi AWS CLI Perintah.

## list-billing-groups

Contoh kode berikut menunjukkan cara menggunakan `list-billing-groups`.

### AWS CLI

Untuk mencantumkan grup penagihan untuk AWS akun dan wilayah Anda

`list-billing-groups` Contoh berikut mencantumkan semua grup penagihan yang ditentukan untuk AWS akun dan AWS Wilayah Anda.

```
aws iot list-billing-groups
```

Output:

```

{
  "billingGroups": [
    {
      "groupName": "GroupOne",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne"
    }
  ]
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListBillingGroups](#) di Referensi AWS CLI Perintah.

## **list-ca-certificates**

Contoh kode berikut menunjukkan cara menggunakan `list-ca-certificates`.

### AWS CLI

Untuk mencantumkan sertifikat CA yang terdaftar di AWS akun Anda

`list-ca-certificates` Contoh berikut mencantumkan sertifikat CA yang terdaftar di AWS akun Anda.

```
aws iot list-ca-certificates
```

Output:

```
{
  "certificates": [
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
      "certificateId":
"f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
      "status": "INACTIVE",
      "creationDate": 1569365372.053
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Sertifikat Anda Sendiri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListCaCertificates](#) di Referensi AWS CLI Perintah.

## **list-certificates-by-ca**

Contoh kode berikut menunjukkan cara menggunakan `list-certificates-by-ca`.

## AWS CLI

Untuk mencantumkan semua sertifikat perangkat yang ditandatangani dengan sertifikat CA

`list-certificates-by-ca` Contoh berikut mencantumkan semua sertifikat perangkat di AWS akun Anda yang ditandatangani dengan sertifikat CA yang ditentukan.

```
aws iot list-certificates-by-ca \
  --ca-certificate-id
  f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467
```

Output:

```
{
  "certificates": [
    {
      "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
      "certificateId":
"488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
      "status": "ACTIVE",
      "creationDate": 1569363250.557
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ListCertificatesByCA](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListCertificatesByCa](#) di Referensi AWS CLI Perintah.

## **list-certificates**

Contoh kode berikut menunjukkan cara menggunakan `list-certificates`.

### AWS CLI

Contoh 1: Untuk mencantumkan sertifikat yang terdaftar di AWS akun Anda

`list-certificates` Contoh berikut mencantumkan semua sertifikat yang terdaftar di akun Anda. Jika Anda memiliki lebih dari batas paging default 25, Anda dapat menggunakan nilai `nextMarker` respons dari perintah ini dan memasoknya ke perintah berikutnya untuk mendapatkan batch hasil berikutnya. Ulangi sampai `nextMarker` kembali tanpa nilai.

```
aws iot list-certificates
```

**Output:**

```
{
  "certificates": [
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36",
      "certificateId": "604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36",
      "status": "ACTIVE",
      "creationDate": 1556810537.617
    },
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/262a1ac8a7d8aa72f6e96e365480f7313aa9db74b8339ec65d34dc3074e1c31e",
      "certificateId": "262a1ac8a7d8aa72f6e96e365480f7313aa9db74b8339ec65d34dc3074e1c31e",
      "status": "ACTIVE",
      "creationDate": 1546447050.885
    },
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b",
      "certificateId": "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b",
      "status": "ACTIVE",
      "creationDate": 1546292258.322
    },
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/7aebeea3845d14a44ec80b06b8b78a89f3f8a706974b8b34d18f5adf0741db42",
      "certificateId": "7aebeea3845d14a44ec80b06b8b78a89f3f8a706974b8b34d18f5adf0741db42",
      "status": "ACTIVE",
      "creationDate": 1541457693.453
    },
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/54458aa39ebb3eb39c91ffbbdcc3a6ca1c7c094d1644b889f735a6fc2cd9a7e3",
      "certificateId": "54458aa39ebb3eb39c91ffbbdcc3a6ca1c7c094d1644b889f735a6fc2cd9a7e3",

```



```

        "status": "ACTIVE",
        "creationDate": 1541113568.611
    },
    {
        "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
        "certificateId":
"4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
        "status": "ACTIVE",
        "creationDate": 1541022751.983
    }
]
}

```

- Untuk detail API, lihat [ListCertificates](#) di Referensi AWS CLI Perintah.

## list-custom-metrics

Contoh kode berikut menunjukkan cara menggunakan `list-custom-metrics`.

### AWS CLI

Untuk membuat daftar metrik kustom

`list-custom-metrics` Contoh berikut mencantumkan semua metrik kustom Anda.

```
aws iot list-custom-metrics \
  --region us-east-1
```

Output:

```
{
  "metricNames": [
    "batteryPercentage"
  ]
}
```

Untuk informasi selengkapnya, lihat [Metrik khusus](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [ListCustomMetrics](#) di Referensi AWS CLI Perintah.

## list-dimensions

Contoh kode berikut menunjukkan cara menggunakan `list-dimensions`.

### AWS CLI

Untuk membuat daftar dimensi untuk AWS akun Anda

`list-dimensions` Contoh berikut mencantumkan semua dimensi AWS IoT Device Defender yang ditentukan AWS di akun Anda.

```
aws iot list-dimensions
```

Output:

```
{
  "dimensionNames": [
    "TopicFilterForAuthMessages",
    "TopicFilterForActivityMessages"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListDimensions](#) di Referensi AWS CLI Perintah.

## list-domain-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-domain-configurations`.

### AWS CLI

Untuk membuat daftar konfigurasi domain

`list-domain-configurations` Contoh berikut mencantumkan konfigurasi domain di AWS akun Anda yang memiliki jenis layanan yang ditentukan.

```
aws iot list-domain-configurations \
  --service-type "DATA"
```

Output:

```
{
  "domainConfigurations":
  [
    {
      "domainConfigurationName": "additionalDataDomain",
      "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh",
      "serviceType": "DATA"
    },
    {
      "domainConfigurationName": "iot:Jobs",
      "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:Jobs",
      "serviceType": "JOBS"
    },
    {
      "domainConfigurationName": "iot:Data-ATS",
      "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:Data-ATS",
      "serviceType": "DATA"
    },
    {
      "domainConfigurationName": "iot:CredentialProvider",
      "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:CredentialProvider",
      "serviceType": "CREDENTIAL_PROVIDER"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir yang Dapat Dikonfigurasi](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListDomainConfigurations](#) di Referensi AWS CLI Perintah.

## list-indices

Contoh kode berikut menunjukkan cara menggunakan `list-indices`.

### AWS CLI

Untuk mencantumkan indeks pencarian yang dikonfigurasi

`list-indices` Contoh berikut mencantumkan semua indeks pencarian yang dikonfigurasi di AWS akun Anda. Jika Anda belum mengaktifkan pengindeksan hal, Anda mungkin tidak memiliki indeks apa pun.

```
aws iot list-indices
```

Output:

```
{
  "indexNames": [
    "AWS_Things"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengindeksan Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListIndices](#) di Referensi AWS CLI Perintah.

## **list-job-executions-for-job**

Contoh kode berikut menunjukkan cara menggunakan `list-job-executions-for-job`.

AWS CLI

Untuk membuat daftar pekerjaan di AWS akun Anda

`list-job-executions-for-job` Contoh berikut mencantumkan semua eksekusi pekerjaan untuk pekerjaan di AWS akun Anda, yang ditentukan oleh JobId.

```
aws iot list-job-executions-for-job \
  --job-id my-ota-job
```

Output:

```
{
  "executionSummaries": [
    {
      "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/my_thing",
      "jobExecutionSummary": {
```

```
        "status": "QUEUED",
        "queuedAt": "2022-03-07T15:58:42.195000-08:00",
        "lastUpdatedAt": "2022-03-07T15:58:42.195000-08:00",
        "executionNumber": 1,
        "retryAttempt": 0
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListJobExecutionsForJob](#) di Referensi AWS CLI Perintah.

## list-job-executions-for-thing

Contoh kode berikut menunjukkan cara menggunakan `list-job-executions-for-thing`.

### AWS CLI

Untuk membuat daftar pekerjaan yang dieksekusi untuk suatu hal

`list-job-executions-for-thing` Contoh berikut mencantumkan semua pekerjaan yang dieksekusi untuk hal bernama `MyRaspberryPi`.

```
aws iot list-job-executions-for-thing \
  --thing-name "MyRaspberryPi"
```

Output:

```
{
  "executionSummaries": [
    {
      "jobId": "example-job-01",
      "jobExecutionSummary": {
        "status": "QUEUED",
        "queuedAt": 1560787023.636,
        "lastUpdatedAt": 1560787023.636,
        "executionNumber": 1
      }
    }
  ]
}
```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListJobExecutionsForThing](#) di Referensi AWS CLI Perintah.

## list-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-jobs`.

### AWS CLI

Untuk membuat daftar pekerjaan di AWS akun Anda

`list-jobs` Contoh berikut mencantumkan semua pekerjaan di AWS akun Anda, diurutkan berdasarkan status pekerjaan.

```
aws iot list-jobs
```

Output:

```
{  
  "jobs": [  
    {  
      "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",  
      "jobId": "example-job-01",  
      "targetSelection": "SNAPSHOT",  
      "status": "IN_PROGRESS",  
      "createdAt": 1560787022.733,  
      "lastUpdatedAt": 1560787026.294  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListJobs](#) di Referensi AWS CLI Perintah.

## list-mitigation-actions

Contoh kode berikut menunjukkan cara menggunakan `list-mitigation-actions`.

### AWS CLI

Untuk mencantumkan semua tindakan mitigasi yang ditentukan

`list-mitigation-actions` Contoh berikut mencantumkan semua tindakan mitigasi yang ditentukan untuk AWS akun dan Wilayah Anda. Untuk setiap tindakan, nama, ARN, dan tanggal pembuatan tercantum.

```
aws iot list-mitigation-actions
```

Output:

```
{
  "actionIdentifiers": [
    {
      "actionName": "DeactivateCACertAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/DeactivateCACertAction",
      "creationDate": "2019-12-10T11:12:47.574000-08:00"
    },
    {
      "actionName": "ResetPolicyVersionAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/ResetPolicyVersionAction",
      "creationDate": "2019-12-10T11:11:48.920000-08:00"
    },
    {
      "actionName": "PublishFindingToSNSAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/PublishFindingToSNSAction",
      "creationDate": "2019-12-10T11:10:49.546000-08:00"
    },
    {
      "actionName": "AddThingsToQuarantineGroupAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/AddThingsToQuarantineGroupAction",
      "creationDate": "2019-12-10T11:09:35.999000-08:00"
    },
    {
```

```

        "actionName": "UpdateDeviceCertAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
UpdateDeviceCertAction",
        "creationDate": "2019-12-10T11:08:44.263000-08:00"
    },
    {
        "actionName": "SampleMitigationAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
SampleMitigationAction",
        "creationDate": "2019-12-10T11:03:41.840000-08:00"
    }
]
}

```

Untuk informasi selengkapnya, lihat [ListMitigationActions \(Perintah Tindakan Mitigasi\)](#) di [Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListMitigationActions](#) di Referensi AWS CLI Perintah.

## list-mitigations-actions

Contoh kode berikut menunjukkan cara menggunakan `list-mitigations-actions`.

### AWS CLI

Untuk mencantumkan semua tindakan mitigasi yang ditentukan

`list-mitigations-actions` Contoh berikut mencantumkan semua tindakan mitigasi yang ditentukan untuk AWS akun dan Wilayah Anda. Untuk setiap tindakan, nama, ARN, dan tanggal pembuatan tercantum.

```
aws iot list-mitigation-actions
```

Output:

```

{
  "actionIdentifiers": [
    {
      "actionName": "DeactivateCACertAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
DeactivateCACertAction",

```



```

        "creationDate": "2019-12-10T11:12:47.574000-08:00"
    },
    {
        "actionName": "ResetPolicyVersionAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
ResetPolicyVersionAction",
        "creationDate": "2019-12-10T11:11:48.920000-08:00"
    },
    {
        "actionName": "PublishFindingToSNSAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
PublishFindingToSNSAction",
        "creationDate": "2019-12-10T11:10:49.546000-08:00"
    },
    {
        "actionName": "AddThingsToQuarantineGroupAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
        "creationDate": "2019-12-10T11:09:35.999000-08:00"
    },
    {
        "actionName": "UpdateDeviceCertAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
UpdateDeviceCertAction",
        "creationDate": "2019-12-10T11:08:44.263000-08:00"
    },
    {
        "actionName": "SampleMitigationAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
SampleMitigationAction",
        "creationDate": "2019-12-10T11:03:41.840000-08:00"
    }
]
}

```

Untuk informasi selengkapnya, lihat [ListMitigationActions \(Perintah Tindakan Mitigasi\)](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListMitigationsActions](#) di Referensi AWS CLI Perintah.

## list-ota-updates

Contoh kode berikut menunjukkan cara menggunakan `list-ota-updates`.

## AWS CLI

Untuk mencantumkan Pembaruan OTA untuk akun

`list-ota-updates` Contoh berikut mencantumkan pembaruan OTA yang tersedia.

```
aws iot list-ota-updates
```

Output:

```
{
  "otaUpdates": [
    {
      "otaUpdateId": "itsaupdate",
      "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",
      "creationDate": 1557863215.995
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Listotaupdates](#) di Referensi API IoT.AWS

- Untuk detail API, lihat [ListOtaUpdates](#) di Referensi AWS CLI Perintah.

## **list-outgoing-certificates**

Contoh kode berikut menunjukkan cara menggunakan `list-outgoing-certificates`.

### AWS CLI

Untuk membuat daftar sertifikat yang ditransfer ke AWS akun lain

`list-outgoing-certificates` Contoh berikut mencantumkan semua sertifikat perangkat yang sedang dalam proses ditransfer ke AWS akun yang berbeda menggunakan `transfer-certificate` perintah.

```
aws iot list-outgoing-certificates
```

Output:

```
{
```

```
"outgoingCertificates": [
  {
    "certificateArn": "arn:aws:iot:us-
west-2:030714055129:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
    "certificateId":
"488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
    "transferredTo": "030714055129",
    "transferDate": 1569427780.441,
    "creationDate": 1569363250.557
  }
]
```

Untuk informasi selengkapnya, lihat [ListOutgoingCertificates](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListOutgoingCertificates](#) di Referensi AWS CLI Perintah.

## list-policies

Contoh kode berikut menunjukkan cara menggunakan `list-policies`.

### AWS CLI

Untuk mencantumkan kebijakan yang ditentukan di AWS akun Anda

`list-policies` Contoh berikut mencantumkan semua kebijakan yang ditentukan di AWS akun Anda.

```
aws iot list-policies
```

Output:

```
{
  "policies": [
    {
      "policyName": "UpdateDeviceCertPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
UpdateDeviceCertPolicy"
    },
    {
      "policyName": "PlantIoTPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/PlantIoTPolicy"
    }
  ]
}
```

```
    },
    {
      "policyName": "MyPiGroup_Core-policy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/MyPiGroup_Core-
policy"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListPolicies](#) di Referensi AWS CLI Perintah.

## list-policy-versions

Contoh kode berikut menunjukkan cara menggunakan `list-policy-versions`.

### AWS CLI

Contoh 1: Untuk melihat semua versi kebijakan

`list-policy-versions` Contoh berikut mencantumkan semua versi kebijakan yang ditentukan dan tanggal pembuatannya.

```
aws iot list-policy-versions \
  --policy-name LightBulbPolicy
```

Output:

```
{
  "policyVersions": [
    {
      "versionId": "2",
      "isDefaultVersion": true,
      "createDate": 1559925941.924
    },
    {
      "versionId": "1",
      "isDefaultVersion": false,
      "createDate": 1559925941.924
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Kebijakan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListPolicyVersions](#) di Referensi AWS CLI Perintah.

## **list-principal-things**

Contoh kode berikut menunjukkan cara menggunakan `list-principal-things`.

### AWS CLI

Untuk membuat daftar hal-hal yang dilampirkan dengan kepala sekolah

`list-principal-things` Contoh berikut mencantumkan hal-hal yang dilampirkan pada prinsipal yang ditentukan oleh ARN.

```
aws iot list-principal-things \  
  --principal arn:aws:iot:us-  
west-2:123456789012:cert/2e1eb273792174ec2b9bf4e9b37e6c6c692345499506002a35159767055278e8
```

Output:

```
{  
  "things": [  
    "DeskLamp",  
    "TableLamp"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListPrincipalThings](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListPrincipalThings](#) di Referensi AWS CLI Perintah.

## **list-provisioning-template-versions**

Contoh kode berikut menunjukkan cara menggunakan `list-provisioning-template-versions`.

### AWS CLI

Untuk membuat daftar versi templat penyediaan

`list-provisioning-template-versions` Contoh berikut mencantumkan versi yang tersedia dari template penyediaan yang ditentukan.

```
aws iot list-provisioning-template-versions \  
  --template-name "widget-template"
```

Output:

```
{  
  "versions": [  
    {  
      "versionId": 1,  
      "creationDate": 1574800471.339,  
      "isDefaultVersion": true  
    },  
    {  
      "versionId": 2,  
      "creationDate": 1574801192.317,  
      "isDefaultVersion": false  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [ListProvisioningTemplateVersions](#) di Referensi AWS CLI Perintah.

## **list-provisioning-templates**

Contoh kode berikut menunjukkan cara menggunakan `list-provisioning-templates`.

AWS CLI

Untuk membuat daftar templat penyediaan

`list-provisioning-templates` Contoh berikut mencantumkan semua templat penyediaan di akun Anda AWS .

```
aws iot list-provisioning-templates
```

Output:

```
{
  "templates": [
    {
      "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/
widget-template",
      "templateName": "widget-template",
      "description": "A provisioning template for widgets",
      "creationDate": 1574800471.367,
      "lastModifiedDate": 1574801192.324,
      "enabled": false
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [ListProvisioningTemplates](#) di Referensi AWS CLI Perintah.

## **list-role-aliases**

Contoh kode berikut menunjukkan cara menggunakan `list-role-aliases`.

### AWS CLI

Untuk membuat daftar alias peran AWS IoT di akun Anda AWS

`list-role-aliases` Contoh berikut mencantumkan alias peran AWS IoT di akun Anda. AWS

```
aws iot list-role-aliases
```

Output:

```
{
  "roleAliases": [
    "ResidentAlias",
    "ElectricianAlias"
  ]
}
```

Untuk informasi selengkapnya, lihat [ListRoleAliases](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListRoleAliases](#) di Referensi AWS CLI Perintah.

## list-scheduled-audits

Contoh kode berikut menunjukkan cara menggunakan `list-scheduled-audits`.

### AWS CLI

Untuk membuat daftar audit terjadwal untuk akun Anda AWS

`list-scheduled-audits` Contoh berikut mencantumkan audit yang dijadwalkan untuk AWS akun Anda.

```
aws iot list-scheduled-audits
```

Output:

```
{
  "scheduledAudits": [
    {
      "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
      "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/AWSIoTDeviceDefenderDailyAudit",
      "frequency": "DAILY"
    },
    {
      "scheduledAuditName": "AWSDeviceDefenderWeeklyAudit",
      "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/AWSDeviceDefenderWeeklyAudit",
      "frequency": "WEEKLY",
      "dayOfWeek": "SUN"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListScheduledAudits](#) di Referensi AWS CLI Perintah.

## list-security-profiles-for-target

Contoh kode berikut menunjukkan cara menggunakan `list-security-profiles-for-target`.



## AWS CLI

Untuk membuat daftar profil keamanan yang dilampirkan pada target

`list-security-profiles-for-target` Contoh berikut mencantumkan profil keamanan AWS IoT Device Defender yang dilampirkan ke perangkat yang tidak terdaftar.

```
aws iot list-security-profiles-for-target \
  --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/
  unregistered-things"
```

Output:

```
{
  "securityProfileTargetMappings": [
    {
      "securityProfileIdentifier": {
        "name": "Testprofile",
        "arn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
Testprofile"
      },
      "target": {
        "arn": "arn:aws:iot:us-west-2:123456789012:all/unregistered-things"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListSecurityProfilesForTarget](#) di Referensi AWS CLI Perintah.

## **list-security-profiles**

Contoh kode berikut menunjukkan cara menggunakan `list-security-profiles`.

### AWS CLI

Untuk membuat daftar profil keamanan untuk AWS akun Anda

`list-security-profiles` Contoh berikut mencantumkan semua profil keamanan AWS IoT Device Defender yang ditentukan AWS di akun Anda.

```
aws iot list-security-profiles
```

Output:

```
{
  "securityProfileIdentifiers": [
    {
      "name": "Testprofile",
      "arn": "arn:aws:iot:us-west-2:123456789012:securityprofile/Testprofile"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListSecurityProfiles](#) di Referensi AWS CLI Perintah.

## list-streams

Contoh kode berikut menunjukkan cara menggunakan `list-streams`.

### AWS CLI

Untuk membuat daftar aliran di akun

`list-streams` Contoh berikut mencantumkan semua aliran di AWS akun Anda.

```
aws iot list-streams
```

Output:

```
{
  "streams": [
    {
      "streamId": "stream12345",
      "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
      "streamVersion": 1,
      "description": "This stream is used for Amazon FreeRTOS OTA Update
12345."
    },
    {
```

```

    "streamId": "stream54321",
    "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream54321",
    "streamVersion": 1,
    "description": "This stream is used for Amazon FreeRTOS OTA Update
54321."
  }
]
}

```

Untuk informasi selengkapnya, lihat [ListStreams](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListStreams](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk menampilkan tag dan nilainya yang terkait dengan sumber daya

`list-tags-for-resource` Contoh berikut menampilkan tag dan nilai-nilai yang terkait dengan kelompok `hallLightBulbs`.

```

aws iot list-tags-for-resource \
  --resource-arn "arn:aws:iot:us-west-2:094249569039:thinggroup/LightBulbs"

```

Output:

```

{
  "tags": [
    {
      "Key": "Assembly",
      "Value": "Fact1NW"
    },
    {
      "Key": "MyTag",
      "Value": "777"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya AWS IoT Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **list-targets-for-policy**

Contoh kode berikut menunjukkan cara menggunakan `list-targets-for-policy`.

### AWS CLI

Untuk membuat daftar prinsipal yang terkait dengan kebijakan IoT AWS

`list-targets-for-policy` Contoh berikut mencantumkan sertifikat perangkat yang dilampirkan kebijakan yang ditentukan.

```
aws iot list-targets-for-policy \
  --policy-name UpdateDeviceCertPolicy
```

Output:

```
{
  "targets": [
    "arn:aws:iot:us-west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
    "arn:aws:iot:us-west-2:123456789012:cert/d1eb269fb55a628552143c8f96eb3c258fcd5331ea113e766ba0c82bf225f0be"
  ]
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListTargetsForPolicy](#) di Referensi AWS CLI Perintah.

## **list-targets-for-security-profile**

Contoh kode berikut menunjukkan cara menggunakan `list-targets-for-security-profile`.

### AWS CLI

Untuk membuat daftar target di mana profil keamanan diterapkan

`list-targets-for-security-profile` Contoh berikut mencantumkan target yang diberi nama `PossibleIssue` profil keamanan AWS IoT Device Defender diterapkan.

```
aws iot list-targets-for-security-profile \  
  --security-profile-name Testprofile
```

Output:

```
{  
  "securityProfileTargets": [  
    {  
      "arn": "arn:aws:iot:us-west-2:123456789012:all/unregistered-things"  
    },  
    {  
      "arn": "arn:aws:iot:us-west-2:123456789012:all/registered-things"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListTargetsForSecurityProfile](#) di Referensi AWS CLI Perintah.

## **list-thing-groups-for-thing**

Contoh kode berikut menunjukkan cara menggunakan `list-thing-groups-for-thing`.

AWS CLI

Untuk membuat daftar kelompok-kelompok yang menjadi milik suatu benda

`list-thing-groups-for-thing` Contoh berikut mencantumkan kelompok-kelompok yang menjadi milik benda yang ditentukan.

```
aws iot list-thing-groups-for-thing \  
  --thing-name MyLightBulb
```

Output:

```
{
```

```
  "thingGroups": [
    {
      "groupName": "DeadBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/DeadBulbs"
    },
    {
      "groupName": "LightBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListThingGroupsForThing](#) di Referensi AWS CLI Perintah.

## list-thing-groups

Contoh kode berikut menunjukkan cara menggunakan `list-thing-groups`.

### AWS CLI

Untuk membuat daftar grup hal yang ditentukan di AWS akun Anda

`describe-thing-group` Contoh berikut mencantumkan semua grup hal yang ditentukan di AWS akun Anda.

```
aws iot list-thing-groups
```

Output:

```
{
  "thingGroups": [
    {
      "groupName": "HalogenBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs"
    },
    {
      "groupName": "LightBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListThingGroups](#) di Referensi AWS CLI Perintah.

## **list-thing-principals**

Contoh kode berikut menunjukkan cara menggunakan `list-thing-principals`.

### AWS CLI

Untuk membuat daftar kepala sekolah yang terkait dengan suatu hal

`list-thing-principals` Contoh berikut mencantumkan prinsipal (sertifikat X.509, pengguna IAM, grup, peran, identitas Amazon Cognito, atau identitas federasi) yang terkait dengan hal yang ditentukan.

```
aws iot list-thing-principals \  
  --thing-name MyRaspberryPi
```

Output:

```
{  
  "principals": [  
    "arn:aws:iot:us-  
west-2:123456789012:cert/33475ac865079a5ffd5ecd44240640349293facc760642d7d8d5dbb6b4c86893"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListThingPrincipals](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListThingPrincipals](#) di Referensi AWS CLI Perintah.

## **list-thing-types**

Contoh kode berikut menunjukkan cara menggunakan `list-thing-types`.

### AWS CLI

Untuk membuat daftar jenis hal yang ditentukan

`list-thing-types` Contoh berikut menampilkan daftar jenis hal yang didefinisikan dalam AWS akun Anda.

```
aws iot list-thing-types
```

Output:

```
{
  "thingTypes": [
    {
      "thingTypeName": "LightBulb",
      "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/
LightBulb",
      "thingTypeProperties": {
        "thingTypeDescription": "light bulb type",
        "searchableAttributes": [
          "model",
          "wattage"
        ]
      },
      "thingTypeMetadata": {
        "deprecated": false,
        "creationDate": 1559772562.498
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Tipe Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListThingTypes](#) di Referensi AWS CLI Perintah.

## **list-things-in-billing-group**

Contoh kode berikut menunjukkan cara menggunakan `list-things-in-billing-group`.

AWS CLI

Untuk membuat daftar hal-hal dalam grup penagihan

`list-things-in-billing-group` Contoh berikut mencantumkan hal-hal yang ada di grup penagihan yang ditentukan.



```
aws iot list-things-in-billing-group \  
  --billing-group-name GroupOne
```

Output:

```
{  
  "things": [  
    "MyOtherLightBulb",  
    "MyLightBulb"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListThingsInBillingGroup](#) di Referensi AWS CLI Perintah.

## list-things-in-thing-group

Contoh kode berikut menunjukkan cara menggunakan `list-things-in-thing-group`.

AWS CLI

Untuk membuat daftar hal-hal yang termasuk dalam grup

`list-things-in-thing-group` Contoh berikut mencantumkan hal-hal yang termasuk dalam kelompok hal yang ditentukan.

```
aws iot list-things-in-thing-group \  
  --thing-group-name LightBulbs
```

Output:

```
{  
  "things": [  
    "MyLightBulb"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListThingsInThingGroup](#) di Referensi AWS CLI Perintah.

## list-things

Contoh kode berikut menunjukkan cara menggunakan `list-things`.

### AWS CLI

Contoh 1: Untuk membuat daftar semua hal dalam registri

`list-things` Contoh berikut mencantumkan hal-hal (perangkat) yang didefinisikan dalam registri AWS IoT untuk akun Anda AWS .

```
aws iot list-things
```

Output:

```
{
  "things": [
    {
      "thingName": "ThirdBulb",
      "thingTypeName": "LightBulb",
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/ThirdBulb",
      "attributes": {
        "model": "123",
        "wattage": "75"
      },
      "version": 2
    },
    {
      "thingName": "MyOtherLightBulb",
      "thingTypeName": "LightBulb",
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb",
      "attributes": {
        "model": "123",
        "wattage": "75"
      },
      "version": 3
    },
    {
      "thingName": "MyLightBulb",
      "thingTypeName": "LightBulb",
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
      "attributes": {
        "model": "123",
```

```
        "wattage": "75"
      },
      "version": 1
    },
    {
      "thingName": "SampleIoTThing",
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/SampleIoTThing",
      "attributes": {},
      "version": 1
    }
  ]
}
```

Contoh 2: Untuk mencantumkan hal-hal yang ditentukan yang memiliki atribut tertentu

`list-things` Contoh berikut menampilkan daftar hal-hal yang memiliki atribut bernama `wattage`.

```
aws iot list-things \
  --attribute-name wattage
```

Output:

```
{
  "things": [
    {
      "thingName": "MyLightBulb",
      "thingTypeName": "LightBulb",
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
      "attributes": {
        "model": "123",
        "wattage": "75"
      },
      "version": 1
    },
    {
      "thingName": "MyOtherLightBulb",
      "thingTypeName": "LightBulb",
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb",
      "attributes": {
        "model": "123",
        "wattage": "75"
      },
    },
  ],
}
```

```
        "version": 3
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Cara Mengelola Sesuatu dengan Registri](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListThings](#) di Referensi AWS CLI Perintah.

## list-topic-rule-destinations

Contoh kode berikut menunjukkan cara menggunakan `list-topic-rule-destinations`.

### AWS CLI

Untuk mencantumkan tujuan aturan topik Anda

`list-topic-rule-destinations` Contoh berikut mencantumkan semua tujuan aturan topik yang telah Anda tetapkan di AWS Wilayah saat ini.

```
aws iot list-topic-rule-destinations
```

### Output:

```
{
  "destinationSummaries": [
    {
      "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "status": "ENABLED",
      "httpUrlSummary": {
        "confirmationUrl": "https://example.com"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan tujuan aturan topik](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListTopicRuleDestinations](#) di Referensi AWS CLI Perintah.

## list-topic-rules

Contoh kode berikut menunjukkan cara menggunakan `list-topic-rules`.

### AWS CLI

Untuk membuat daftar aturan Anda

`list-topic-rules` Contoh berikut mencantumkan semua aturan yang telah Anda tetapkan.

```
aws iot list-topic-rules
```

Output:

```
{
  "rules": [
    {
      "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/MyRPiLowMoistureAlertRule",
      "ruleName": "MyRPiLowMoistureAlertRule",
      "topicPattern": "$aws/things/MyRPi/shadow/update/accepted",
      "createdAt": 1558624363.0,
      "ruleDisabled": false
    },
    {
      "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/MyPlantPiMoistureAlertRule",
      "ruleName": "MyPlantPiMoistureAlertRule",
      "topicPattern": "$aws/things/MyPlantPi/shadow/update/accepted",
      "createdAt": 1541458459.0,
      "ruleDisabled": false
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Aturan Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListTopicRules](#) di Referensi AWS CLI Perintah.

## list-v2-logging-levels

Contoh kode berikut menunjukkan cara menggunakan `list-v2-logging-levels`.

## AWS CLI

Untuk daftar tingkat logging

`list-v2-logging-levels` Contoh berikut mencantumkan level logging yang dikonfigurasi. Jika tingkat logging tidak disetel, a `NotConfiguredException` terjadi ketika Anda menjalankan perintah ini.

```
aws iot list-v2-logging-levels
```

Output:

```
{
  "logTargetConfigurations": [
    {
      "logTarget": {
        "targetType": "DEFAULT"
      },
      "logLevel": "ERROR"
    }
  ]
}
```

- Untuk detail API, lihat [ListV2 LoggingLevels](#) di Referensi AWS CLI Perintah.

## `list-violation-events`

Contoh kode berikut menunjukkan cara menggunakan `list-violation-events`.

## AWS CLI

Untuk membuat daftar pelanggaran profil keamanan selama periode waktu

`list-violation-events` Contoh berikut mencantumkan pelanggaran yang terjadi antara 5 Juni 2019 dan 12 Juni 2019 untuk semua profil keamanan AWS IoT Device Defender untuk AWS akun dan Wilayah saat ini AWS .

```
aws iot list-violation-events \
  --start-time 1559747125 \
  --end-time 1560351925
```

## Output:

```
{
  "violationEvents": [
    {
      "violationId": "174db59167fa474c80a652ad1583fd44",
      "thingName": "iotconsole-1560269126751-1",
      "securityProfileName": "Testprofile",
      "behavior": {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
          "comparisonOperator": "greater-than",
          "value": {
            "count": 10
          },
          "durationSeconds": 300,
          "consecutiveDatapointsToAlarm": 1,
          "consecutiveDatapointsToClear": 1
        }
      },
      "metricValue": {
        "count": 0
      },
      "violationEventType": "in-alarm",
      "violationEventTime": 1560279000.0
    },
    {
      "violationId": "c8a9466a093d3b7b35cd44ca58bdbeab",
      "thingName": "TvnQoEoU",
      "securityProfileName": "Testprofile",
      "behavior": {
        "name": "CellularBandwidth",
        "metric": "aws:message-byte-size",
        "criteria": {
          "comparisonOperator": "greater-than",
          "value": {
            "count": 128
          },
          "consecutiveDatapointsToAlarm": 1,
          "consecutiveDatapointsToClear": 1
        }
      },
      "metricValue": {
```

```
        "count": 110
      },
      "violationEventType": "in-alarm",
      "violationEventTime": 1560276600.0
    },
    {
      "violationId": "74aa393adea02e6648f3ac362beed55e",
      "thingName": "iotconsole-1560269232412-2",
      "securityProfileName": "Testprofile",
      "behavior": {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
          "comparisonOperator": "greater-than",
          "value": {
            "count": 10
          },
          "durationSeconds": 300,
          "consecutiveDatapointsToAlarm": 1,
          "consecutiveDatapointsToClear": 1
        }
      },
      "metricValue": {
        "count": 0
      },
      "violationEventType": "in-alarm",
      "violationEventTime": 1560276600.0
    },
    {
      "violationId": "1e6ab5f7cf39a1466fcd154e1377e406",
      "thingName": "TvnQoEoU",
      "securityProfileName": "Testprofile",
      "behavior": {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
          "comparisonOperator": "greater-than",
          "value": {
            "count": 10
          },
          "durationSeconds": 300,
          "consecutiveDatapointsToAlarm": 1,
          "consecutiveDatapointsToClear": 1
        }
      }
    }
  ]
}
```



```

    },
    "metricValue": {
      "count": 0
    },
    "violationEventType": "in-alarm",
    "violationEventTime": 1560276600.0
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListViolationEvents](#) di Referensi AWS CLI Perintah.

## register-ca-certificate

Contoh kode berikut menunjukkan cara menggunakan `register-ca-certificate`.

### AWS CLI

Untuk mendaftarkan sertifikat otoritas sertifikat (CA)

`register-ca-certificate` Contoh berikut mendaftarkan sertifikat CA. Perintah ini menyediakan sertifikat CA dan sertifikat verifikasi kunci yang membuktikan bahwa Anda memiliki kunci pribadi yang terkait dengan sertifikat CA.

```

aws iot register-ca-certificate \
  --ca-certificate file://rootCA.pem \
  --verification-cert file://verificationCert.pem

```

Output:

```

{
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
  "certificateId":
"f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467"
}

```

Untuk informasi selengkapnya, lihat [RegisterCACertificate di Referensi](#) API IoT.AWS

- Untuk detail API, lihat [RegisterCaCertificate](#) di Referensi AWS CLI Perintah.

## register-certificate

Contoh kode berikut menunjukkan cara menggunakan `register-certificate`.

### AWS CLI

Untuk mendaftarkan sertifikat perangkat yang ditandatangani sendiri

`register-certificate` Contoh berikut mendaftarkan sertifikat `deviceCert.pem` perangkat yang ditandatangani oleh sertifikat `rootCA.pem` CA. Sertifikat CA harus didaftarkan sebelum Anda menggunakannya untuk mendaftarkan sertifikat perangkat yang ditandatangani sendiri. Sertifikat yang ditandatangani sendiri harus ditandatangani oleh sertifikat CA yang sama yang Anda berikan ke perintah ini.

```
aws iot register-certificate \  
  --certificate-pem file://deviceCert.pem \  
  --ca-certificate-pem file://rootCA.pem
```

Output:

```
{  
  "certificateArn": "arn:aws:iot:us-  
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",  
  "certificateId":  
  "488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142"  
}
```

Untuk informasi selengkapnya, lihat [RegisterCertificate](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [RegisterCertificate](#) di Referensi AWS CLI Perintah.

## register-thing

Contoh kode berikut menunjukkan cara menggunakan `register-thing`.

### AWS CLI

Untuk mendaftarkan sesuatu

`register-thing` Contoh berikut mendaftarkan sesuatu menggunakan template penyedia.

```
aws iot register-thing \
  --template-body '{"Parameters":{"ThingName":
{"Type":"String"},"AWS::IoT::Certificate::Id":{"Type":"String"},"Resources":
{"certificate":{"Properties":{"CertificateId":
{"Ref":"AWS::IoT::Certificate::Id"},"Status":"Active"},"Type":"AWS::IoT::Certificate"},"poli
{"Properties":{"PolicyName":"MyIotPolicy"},"Type":"AWS::IoT::Policy"},"thing":
{"OverrideSettings":
{"AttributePayload":"MERGE","ThingGroups":"DO_NOTHING","ThingTypeName":"REPLACE"},"Propertie
{"AttributePayload":{},"ThingGroups":[],"ThingName":
{"Ref":"ThingName"},"ThingTypeName":"VirtualThings"},"Type":"AWS::IoT::Thing"}}}' \
  --parameters '{"ThingName":"Register-thing-
trial-1","AWS::IoT::Certificate::Id":"799a9ea048a1e6aea42b55EXAMPLEf8697b4bafcd77a318a3068e3
```

### Output:

```
{
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCCAKGgAwIBAgIUYLk81I35cIppobpw
Hi0J2jNjboIwDQYJKoZIhvcNAQEL
\nBQAwTTFLEkGA1UECwxQW1hem9uIFd1YiBTZXJ2aWNlcyBPPUftYXpvbi
5jb20g\nSW5jLiBMPVNlYXR0bGUgU1Q9V2FzaGluZ3RvbiBDPVVTMB4XDTIwMDcyMzE2NDUw
\n0VoXDTQ5MTIzMT
IzNTk10VowHjEcMBoGA1UEAwTQVdTIElvVCBDZXJ0aWZpY2F0\nZTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBA071uADhdBajqTmgrMV5\nmCFfBZQRMo1MdtVoZr2X+M4MzL
+RARrtUzH9a2SMAckeX8Keb1I0TKz0RI
RDXnyE
\n6lV0wjgAsd0ku22rFxex4eG2ikha7pYYkvuToqA7L3TxItRvfKrxRI4ZfJoFPip4\nKqiuBJVNOGKTcQ
Hd1RN0rddwwu6kFJLeKDMEXAMPLEdUF0N+qfR9yKnZQkm
+g6Q2\nGXu7u0W3hn6n1RN8qVoka0uW12p53xM7oHVz
Gf+cxKBx1b0hGkp6yCfTskUBm3Sp\n9zLw35kiHXVm4EVpWgNlnk6XcIGIkw8a/iy4pzmVUgAANY1/uU/
zgCjymw
ZT5S30\nBV0CAwEAAaNgMF4wHwYDVR0jBBgwFoAUGx0tCcU3q2n1WXAuUCv6hugXjKswHQYD
\nVR00BBYEF0VtvZ
9Aj2RYFnkX7Iu01XTRUdxgMAwGA1UdEwEB/wQMAAwDgYDVR0P\nAQH/
BAQDAgeAMA0GCSqGSIb3DQEBCwUAA4IB
AQXCQCcp0tubS5ft0sDMTcP/jNX
\nDHyaRxmjpSc2aCdm7WX591TKWyAdxGAvqaDVWqTo0oXI7tZ8w7aINlGi5
pXnifx\n3SBebMUoBbTktrC97yUaeL025mCFv8emDnTR/fe7PTsBKjW0g/rrfpwBxZLXDFwN
\nnqkQjy3EDfifj2
6j0xYIqqWMPogyn4sr0CKynS5wMJuQZ1HQ0nabVwnwK4Y0Mf1p
\np9+4susFUR9aT3BT1AcIwqSpzh1Khh4Iz7ND
kRn4amsUT210jg/z0010w+BTHcVQ\nJly8XDuoCWSu04q6SnaBzHmlySIajxuRTP/AdfRouP10Xe
+q1bPOBcvVvF
```

```
8o\n-----END CERTIFICATE-----\n",
  "resourceArns": {
    "certificate": "arn:aws:iot:us-
west-2:571032923833:cert/799a9ea048a1e6aea42b55EXAMPLEf8697b4bafcd77a318a3068e30404b9233c",
    "thing": "arn:aws:iot:us-west-2:571032923833:thing/Register-thing-trial-1"
  }
}
```

Untuk informasi selengkapnya, lihat [Penyediaan oleh pengguna tepercaya di Panduan Pengembang Inti AWS IoT](#).

- Untuk detail API, lihat [RegisterThing](#) di Referensi AWS CLI Perintah.

## reject-certificate-transfer

Contoh kode berikut menunjukkan cara menggunakan `reject-certificate-transfer`.

### AWS CLI

Untuk menolak transfer sertifikat

`reject-certificate-transfer` Contoh berikut menolak transfer sertifikat perangkat yang ditentukan dari AWS akun lain.

```
aws iot reject-certificate-transfer \
  --certificate-id
  f0f33678c7c9a046e5cc87b2b1a58dfa0beec26db78addd5e605d630e05c7fc8
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mentransfer sertifikat ke akun lain](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [RejectCertificateTransfer](#) di Referensi AWS CLI Perintah.

## remove-thing-from-billing-group

Contoh kode berikut menunjukkan cara menggunakan `remove-thing-from-billing-group`.

### AWS CLI

Untuk menghapus sesuatu dari grup penagihan

`remove-thing-from-billing-group` Contoh berikut menghapus hal yang ditentukan dari grup penagihan.

```
aws iot remove-thing-from-billing-group \  
  --billing-group-name GroupOne \  
  --thing-name MyOtherLightBulb
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [RemoveThingFromBillingGroup](#) di Referensi AWS CLI Perintah.

## **remove-thing-from-thing-group**

Contoh kode berikut menunjukkan cara menggunakan `remove-thing-from-thing-group`.

### AWS CLI

Untuk menghapus sesuatu dari grup benda

`remove-thing-from-thing-group` Contoh berikut menghapus hal yang ditentukan dari kelompok hal.

```
aws iot remove-thing-from-thing-group \  
  --thing-name bulb7 \  
  --thing-group-name DeadBulbs
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat Thing Groups < <https://docs.aws.amazon.com/iot/latest/developerguide/thing-groups.html> > di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [RemoveThingFromThingGroup](#) di Referensi AWS CLI Perintah.

## **replace-topic-rule**

Contoh kode berikut menunjukkan cara menggunakan `replace-topic-rule`.

### AWS CLI

Untuk memperbarui definisi aturan topik

`replace-topic-rule` Contoh berikut memperbarui aturan yang ditentukan untuk mengirim peringatan SNS ketika pembacaan tingkat kelembaban tanah terlalu rendah.

```
aws iot replace-topic-rule \  
  --rule-name MyRPiLowMoistureAlertRule \  
  --topic-rule-payload "{\"sql\": \"SELECT * FROM '$aws/things/MyRPi/shadow/  
update/accepted' WHERE state.reported.moisture = 'low'\", \"description\": \"Sends  
an alert when soil moisture level readings are too low.\", \"actions\": [{\"sns  
\": {\"targetArn\": \"arn:aws:sns:us-west-2:123456789012:MyRPiLowMoistureTopic\",  
\"roleArn\": \"arn:aws:iam::123456789012:role/service-role/MyRPiLowMoistureTopicRole  
\", \"messageFormat\": \"RAW\"}}], \"ruleDisabled\": false, \"awsIotSqlVersion\":  
\"2016-03-23\"}"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Aturan AWS IoT di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ReplaceTopicRule](#) di Referensi AWS CLI Perintah.

## search-index

Contoh kode berikut menunjukkan cara menggunakan `search-index`.

### AWS CLI

Untuk menanyakan indeks benda

`search-index` Contoh berikut query `AWS_Things` indeks untuk hal-hal yang memiliki jenis `LightBulb`

```
aws iot search-index \  
  --index-name "AWS_Things" \  
  --query-string "thingTypeName:LightBulb"
```

Output:

```
{  
  "things": [  
    {  
      "thingName": "MyLightBulb",  
      "thingId": "40da2e73-c6af-406e-b415-15acae538797",
```

```
    "thingTypeName": "LightBulb",
    "thingGroupNames": [
      "LightBulbs",
      "DeadBulbs"
    ],
    "attributes": {
      "model": "123",
      "wattage": "75"
    },
    "connectivity": {
      "connected": false
    }
  },
  {
    "thingName": "ThirdBulb",
    "thingId": "615c8455-33d5-40e8-95fd-3ee8b24490af",
    "thingTypeName": "LightBulb",
    "attributes": {
      "model": "123",
      "wattage": "75"
    },
    "connectivity": {
      "connected": false
    }
  },
  {
    "thingName": "MyOtherLightBulb",
    "thingId": "6dae0d3f-40c1-476a-80c4-1ed24ba6aa11",
    "thingTypeName": "LightBulb",
    "attributes": {
      "model": "123",
      "wattage": "75"
    },
    "connectivity": {
      "connected": false
    }
  }
]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Pengindeksan Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [SearchIndex](#) di Referensi AWS CLI Perintah.

## set-default-authorizer

Contoh kode berikut menunjukkan cara menggunakan `set-default-authorizer`.

### AWS CLI

Untuk menetapkan otorisasi default

`set-default-authorizer` Contoh berikut menetapkan authorizer kustom bernama `CustomAuthorizer` sebagai authorizer default.

```
aws iot set-default-authorizer \  
  --authorizer-name CustomAuthorizer
```

Output:

```
{  
  "authorizerName": "CustomAuthorizer",  
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/  
CustomAuthorizer"  
}
```

Untuk informasi selengkapnya, lihat [CreateDefaultAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [SetDefaultAuthorizer](#) di Referensi AWS CLI Perintah.

## set-default-policy-version

Contoh kode berikut menunjukkan cara menggunakan `set-default-policy-version`.

### AWS CLI

Untuk menyetel versi default untuk kebijakan

`set-default-policy-version` Contoh berikut menetapkan versi default 2 untuk kebijakan bernama `UpdateDeviceCertPolicy`.

```
aws iot set-default-policy-version \  
  --policy-name UpdateDeviceCertPolicy \  
  --policy-version-id 2
```



Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetDefaultPolicyVersion](#) di Referensi AWS CLI Perintah.

## set-v2-logging-level

Contoh kode berikut menunjukkan cara menggunakan `set-v2-logging-level`.

### AWS CLI

Untuk mengatur tingkat logging untuk grup sesuatu

`set-v2-logging-level` Contoh berikut menetapkan tingkat logging untuk log peringatan untuk kelompok hal yang ditentukan.

```
aws iot set-v2-logging-level \  
  --log-target "{\"targetType\":\"THING_GROUP\",\"targetName\":\"LightBulbs\"}" \  
  --log-level WARN
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetV2 LoggingLevel](#) di Referensi AWS CLI Perintah.

## set-v2-logging-options

Contoh kode berikut menunjukkan cara menggunakan `set-v2-logging-options`.

### AWS CLI

Untuk mengatur opsi pencatatan

`set-v2-logging-options` Contoh berikut menetapkan tingkat verbositas logging default ke ERROR dan menentukan ARN yang akan digunakan untuk logging.

```
aws iot set-v2-logging-options \  
  --default-log-level ERROR \  
  --role-arn "arn:aws:iam::094249569039:role/service-role/iotLoggingRole"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetV2 LoggingOptions](#) di Referensi AWS CLI Perintah.

## start-audit-mitigation-actions-task

Contoh kode berikut menunjukkan cara menggunakan `start-audit-mitigation-actions-task`.

### AWS CLI

Untuk menerapkan tindakan mitigasi terhadap temuan dari audit

`start-audit-mitigation-actions-task` Contoh berikut menerapkan `ResetPolicyVersionAction` tindakan (yang menghapus kebijakan) ke temuan tunggal yang ditentukan.

```
aws iot start-audit-mitigation-actions-task \
  --task-id "myActionsTaskId" \
  --target "findingIds=[\"0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464\"]" \
  --audit-check-to-actions-mapping
  "IOT_POLICY_OVERLY_PERMISSIVE_CHECK=[\"ResetPolicyVersionAction\"]" \
  --client-request-token "adhadhahda"
```

Output:

```
{
  "taskId": "myActionsTaskId"
}
```

Untuk informasi selengkapnya, lihat [StartAuditMitigationActionsTask \(Perintah Tindakan Mitigasi\) di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [StartAuditMitigationActionsTask](#) di Referensi AWS CLI Perintah.

## start-on-demand-audit-task

Contoh kode berikut menunjukkan cara menggunakan `start-on-demand-audit-task`.

### AWS CLI

Untuk segera memulai audit

`start-on-demand-audit-task` Contoh berikut memulai audit AWS IoT Device Defender dan melakukan tiga pemeriksaan sertifikat.

```
aws iot start-on-demand-audit-task \  
  --target-check-names CA_CERTIFICATE_EXPIRING_CHECK  
  DEVICE_CERTIFICATE_EXPIRING_CHECK REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK
```

Output:

```
{  
  "taskId": "a3aea009955e501a31b764abe1bebd3d"  
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [StartOnDemandAuditTask](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menentukan kunci tag dan nilai untuk sumber daya

`tag-resource` Contoh berikut menerapkan tag dengan kunci Assembly dan nilai Fact1NW untuk kelompok `halLightBulbs`.

```
aws iot tag-resource \  
  --tags Key=Assembly,Value="Fact1NW" \  
  --resource-arn "arn:aws:iot:us-west-2:094249569039:thinggroup/LightBulbs"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya AWS IoT Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## test-authorization

Contoh kode berikut menunjukkan cara menggunakan `test-authorization`.

## AWS CLI

Untuk menguji kebijakan AWS IoT Anda

`test-authorization` Contoh berikut menguji kebijakan AWS IoT yang terkait dengan prinsipal yang ditentukan.

```
aws iot test-authorization \  
  --auth-infos actionType=CONNECT,resources=arn:aws:iot:us-  
east-1:123456789012:client/client1 \  
  --principal arn:aws:iot:us-west-2:123456789012:cert/  
aab1068f7f43ac3e3cae4b3a8aa3f308d2a750e6350507962e32c1eb465d9775
```

Output:

```
{  
  "authResults": [  
    {  
      "authInfo": {  
        "actionType": "CONNECT",  
        "resources": [  
          "arn:aws:iot:us-east-1:123456789012:client/client1"  
        ]  
      },  
      "allowed": {  
        "policies": [  
          {  
            "policyName": "TestPolicyAllowed",  
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/  
TestPolicyAllowed"  
          }  
        ]  
      },  
      "denied": {  
        "implicitDeny": {  
          "policies": [  
            {  
              "policyName": "TestPolicyDenied",  
              "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/  
TestPolicyDenied"  
            }  
          ]  
        }  
      },  
    ]  
  },  
}
```

```

        "explicitDeny": {
            "policies": [
                {
                    "policyName": "TestPolicyExplicitDenied",
                    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TestPolicyExplicitDenied"
                }
            ]
        },
        "authDecision": "IMPLICIT_DENY",
        "missingContextValues": []
    }
]
}

```

Untuk informasi selengkapnya, lihat [TestAuthorization](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [TestAuthorization](#) di Referensi AWS CLI Perintah.

## test-invoke-authorizer

Contoh kode berikut menunjukkan cara menggunakan `test-invoke-authorizer`.

### AWS CLI

Untuk menguji otorisasi kustom Anda

`test-invoke-authorizer` Contoh berikut menguji otorisasi kustom Anda.

```

aws iot test-invoke-authorizer \
  --authorizer-name IoTAuthorizer \
  --token allow \
  --token-signature "mE0GvaHqy9nER/
FdgtJX5lXYEJ3b3vE7t1gEszc0TKGgLKWXTnPk2AbKn0AZ8lGyoN5dVtWDWVmr25m7+
+zjbYIMk2TBvyGXh0mvKFBPkdgyA43KL6SiZy0cTq1PMcQDsP7VX2rXr7CTowCxSNKphGXdQe0/
I5dQ+J06KUaHwCmupt0/MejKtaNwiiA064j6wpr0AUwG5S1IYFuRd0X
+wfo8pb0DubAIX1Ua705kuhRUcTx4SxUSHEYKmN4IDEvLB6FsIr0B2wvB7y4iPmcajxzG102ExvyCUNctCV9dY1RRGJj

```

Output:

```
{
```

```

    "isAuthenticated": true,
    "principalId": "principalId",
    "policyDocuments": [
      {"Version": "2012-10-17", "Statement":
[{"Action": "iot:Publish", "Effect": "Allow", "Resource": "arn:aws:iot:us-
west-2:123456789012:topic/customauthtesting"}]}]
    ],
    "refreshAfterInSeconds": 600,
    "disconnectAfterInSeconds": 3600
  }

```

Untuk informasi selengkapnya, lihat [TestInvokeAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [TestInvokeAuthorizer](#) di Referensi AWS CLI Perintah.

## transfer-certificate

Contoh kode berikut menunjukkan cara menggunakan `transfer-certificate`.

### AWS CLI

Untuk mentransfer sertifikat perangkat ke AWS akun lain

`transfer-certificate` Contoh berikut mentransfer sertifikat perangkat ke AWS akun lain. Sertifikat dan AWS akun diidentifikasi oleh ID.

```

aws iot transfer-certificate \
  --certificate-id
488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142 \
  --target-aws-account 030714055129

```

Output:

```

{
  "transferredCertificateArn": "arn:aws:iot:us-
west-2:030714055129:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142"
}

```

Untuk informasi selengkapnya, lihat [Mentransfer sertifikat ke akun lain](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [TransferCertificate](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus kunci tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag `MyTag` dan nilainya dari kelompok `hallLightBulbs`.

```
command
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya AWS IoT Anda](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-account-audit-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-account-audit-configuration`.

### AWS CLI

Contoh 1: Untuk mengaktifkan notifikasi Amazon SNS untuk notifikasi audit

`update-account-audit-configuration` Contoh berikut memungkinkan notifikasi Amazon SNS untuk AWS notifikasi audit IoT Device Defender, menentukan target dan peran yang digunakan untuk menulis ke target tersebut.

```
aws iot update-account-audit-configuration \  
  --audit-notification-target-configurations "SNS={targetArn=\"arn:aws:sns:us-  
west-2:123456789012:ddauidits\",roleArn=\"arn:aws:iam::123456789012:role/service-  
role/AWSIoTDeviceDefenderAudit\",enabled=true}"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk mengaktifkan pemeriksaan audit

`update-account-audit-configuration` Contoh berikut memungkinkan pemeriksaan audit AWS IoT Device Defender bernama.

AUTHENTICATED\_COGNITO\_ROLE\_OVERLY\_PERMISSIVE\_CHECK Anda tidak dapat menonaktifkan pemeriksaan audit jika itu adalah bagian dari `targetCheckNames` untuk satu atau beberapa audit terjadwal untuk AWS akun tersebut.

```
aws iot update-account-audit-configuration \  
  --audit-check-configurations \  
  "{\"AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK\":{\"enabled\":true}}"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateAccountAuditConfiguration](#) di Referensi AWS CLI Perintah.

## update-audit-suppression

Contoh kode berikut menunjukkan cara menggunakan `update-audit-suppression`.

### AWS CLI

Untuk memperbarui penindasan temuan audit

`update-audit-suppression` Contoh berikut memperbarui tanggal kedaluwarsa penindasan temuan audit ke 2020-09-21.

```
aws iot update-audit-suppression \  
  --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \  
  --resource-identifier deviceCertificateId=c7691e<shortened> \  
  --no-suppress-indefinitely \  
  --expiration-date 2020-09-21
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Audit menemukan penekanan di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateAuditSuppression](#) di Referensi AWS CLI Perintah.

## update-authorizer

Contoh kode berikut menunjukkan cara menggunakan `update-authorizer`.



## AWS CLI

Untuk memperbarui otorisasi kustom

`update-authorizer` Contoh berikut dia nyatakan `CustomAuthorizer2` untuk `INACTIVE`.

```
aws iot update-authorizer \  
  --authorizer-name CustomAuthorizer2 \  
  --status INACTIVE
```

Output:

```
{  
  "authorizerName": "CustomAuthorizer2",  
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/  
CustomAuthorizer2"  
}
```

Untuk informasi selengkapnya, lihat [UpdateAuthorizer](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [UpdateAuthorizer](#) di Referensi AWS CLI Perintah.

## update-billing-group

Contoh kode berikut menunjukkan cara menggunakan `update-billing-group`.

## AWS CLI

Untuk memperbarui informasi tentang grup penagihan

`update-billing-group` Contoh berikut memperbarui deskripsi untuk grup penagihan yang ditentukan.

```
aws iot update-billing-group \  
  --billing-group-name GroupOne \  
  --billing-group-properties "billingGroupDescription=\"Primary bulb billing group  
\""
```

Output:

```
{
```

```
"version": 2
}
```

Untuk informasi selengkapnya, lihat [Grup Penagihan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateBillingGroup](#) di Referensi AWS CLI Perintah.

## update-ca-certificate

Contoh kode berikut menunjukkan cara menggunakan `update-ca-certificate`.

### AWS CLI

Untuk memperbarui sertifikat otoritas sertifikat (CA)

`update-ca-certificate` Contoh berikut menetapkan sertifikat CA yang ditentukan ke status AKTIF.

```
aws iot update-ca-certificate \
  --certificate-id
  f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467 \
  --new-status ACTIVE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UpdateCaCertificate di Referensi API IoT.AWS](#)

- Untuk detail API, lihat [UpdateCaCertificate](#) di Referensi AWS CLI Perintah.

## update-certificate

Contoh kode berikut menunjukkan cara menggunakan `update-certificate`.

### AWS CLI

Untuk memperbarui sertifikat perangkat

`update-certificate` Contoh berikut menetapkan sertifikat perangkat yang ditentukan ke status INACTIVE.

```
aws iot update-certificate \
```

```
--certificate-id
d1eb269fb55a628552143c8f96eb3c258fcd5331ea113e766ba0c82bf225f0be \
--new-status INACTIVE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UpdateCertificate](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [UpdateCertificate](#) di Referensi AWS CLI Perintah.

## update-custom-metric

Contoh kode berikut menunjukkan cara menggunakan `update-custom-metric`.

### AWS CLI

Untuk memperbarui metrik kustom

`update-custom-metric` Contoh berikut memperbarui metrik khusus untuk memiliki yang baru `display-name`.

```
aws iot update-custom-metric \
  --metric-name batteryPercentage \
  --display-name 'remaining battery percentage on device' \
  --region us-east-1
```

Output:

```
{
  "metricName": "batteryPercentage",
  "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/
batteryPercentage",
  "metricType": "number",
  "displayName": "remaining battery percentage on device",
  "creationDate": "2020-11-17T23:01:35.110000-08:00",
  "lastModifiedDate": "2020-11-17T23:02:12.879000-08:00"
}
```

Untuk informasi selengkapnya, lihat [Metrik khusus](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [UpdateCustomMetric](#) di Referensi AWS CLI Perintah.

## update-dimension

Contoh kode berikut menunjukkan cara menggunakan `update-dimension`.

### AWS CLI

Untuk memperbarui dimensi

`update-dimension` Contoh berikut memperbarui dimensi.

```
aws iot update-dimension \  
  --name TopicFilterForAuthMessages \  
  --string-values device/${iot:ClientId}/auth
```

Output:

```
{  
  "name": "TopicFilterForAuthMessages",  
  "lastModifiedDate": 1585866222.317,  
  "stringValues": [  
    "device/${iot:ClientId}/auth"  
  ],  
  "creationDate": 1585854500.474,  
  "type": "TOPIC_FILTER",  
  "arn": "arn:aws:iot:us-west-2:1234564789012:dimension/  
TopicFilterForAuthMessages"  
}
```

Untuk informasi selengkapnya, lihat [Metrik pelingkupan di profil keamanan menggunakan dimensi di Panduan Pengembang](#) Inti AWS IoT.

- Untuk detail API, lihat [UpdateDimension](#) di Referensi AWS CLI Perintah.

## update-domain-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-domain-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi domain

`update-domain-configuration` Contoh berikut menonaktifkan konfigurasi domain yang ditentukan.

```
aws iot update-domain-configuration \  
  --domain-configuration-name "additionalDataDomain" \  
  --domain-configuration-status "DISABLED"
```

Output:

```
{  
  "domainConfigurationName": "additionalDataDomain",  
  "domainConfigurationArn": "arn:aws:iot:us-  
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh"  
}
```

Untuk informasi selengkapnya, lihat [Titik Akhir yang Dapat Dikonfigurasi](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateDomainConfiguration](#) di Referensi AWS CLI Perintah.

## update-dynamic-thing-group

Contoh kode berikut menunjukkan cara menggunakan `update-dynamic-thing-group`.

### AWS CLI

Untuk memperbarui grup hal dinamis

`update-dynamic-thing-group` Contoh berikut memperbarui kelompok hal dinamis yang ditentukan. Ini memberikan deskripsi dan memperbarui string kueri untuk mengubah kriteria keanggotaan grup.

```
aws iot update-dynamic-thing-group \  
  --thing-group-name "RoomTooWarm"  
  --thing-group-properties "thingGroupDescription=\"This thing group contains  
rooms warmer than 65F.\""  
  --query-string "attributes.temperature>65"
```

Output:

```
{  
  "version": 2  
}
```

Untuk informasi selengkapnya, lihat [Dynamic Thing Groups](#) di AWS Panduan Pengembang IoT.

- Untuk detail API, lihat [UpdateDynamicThingGroup](#) di Referensi AWS CLI Perintah.

## update-event-configurations

Contoh kode berikut menunjukkan cara menggunakan `update-event-configurations`.

### AWS CLI

Untuk menunjukkan jenis acara mana yang diterbitkan

`update-event-configurations` Contoh berikut memperbarui konfigurasi untuk mengaktifkan pesan saat sertifikat CA ditambahkan, diperbarui, atau dihapus.

```
aws iot update-event-configurations \  
  --event-configurations "{\"CA_CERTIFICATE\":{\"Enabled\":true}}"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pesan Acara](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateEventConfigurations](#) di Referensi AWS CLI Perintah.

## update-indexing-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-indexing-configuration`.

### AWS CLI

Untuk mengaktifkan pengindeksan hal

`update-indexing-configuration` Contoh berikut memungkinkan pengindeksan hal untuk mendukung pencarian data registri, data bayangan, dan status konektivitas benda menggunakan indeks `AWS_Things`.

```
aws iot update-indexing-configuration \  
  --thing-indexing-configuration \  
  thingIndexingMode=REGISTRY_AND_SHADOW,thingConnectivityIndexingMode=STATUS
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Pengindeksan Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateIndexingConfiguration](#) di Referensi AWS CLI Perintah.

## update-job

Contoh kode berikut menunjukkan cara menggunakan `update-job`.

### AWS CLI

Untuk mendapatkan status terperinci untuk suatu pekerjaan

`update-job` Contoh berikut mendapatkan status rinci untuk pekerjaan yang ID-nya `example-job-01`.

```
aws iot describe-job \
  --job-id "example-job-01"
```

Output:

```
{
  "job": {
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
    "jobId": "example-job-01",
    "targetSelection": "SNAPSHOT",
    "status": "IN_PROGRESS",
    "targets": [
      "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi"
    ],
    "description": "example job test",
    "presignedUrlConfig": {},
    "jobExecutionsRolloutConfig": {},
    "createdAt": 1560787022.733,
    "lastUpdatedAt": 1560787026.294,
    "jobProcessDetails": {
      "numberOfCanceledThings": 0,
      "numberOfSucceededThings": 0,
      "numberOfFailedThings": 0,
      "numberOfRejectedThings": 0,
      "numberOfQueuedThings": 1,
      "numberOfInProgressThings": 0,
    }
  }
}
```

```

        "numberOfRemovedThings": 0,
        "numberOfTimedOutThings": 0
    },
    "timeoutConfig": {}
}

```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Pekerjaan \(CLI\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [UpdateJob](#) di Referensi AWS CLI Perintah.

## update-mitigation-action

Contoh kode berikut menunjukkan cara menggunakan `update-mitigation-action`.

### AWS CLI

Untuk memperbarui tindakan mitigasi

`update-mitigation-action` Contoh berikut memperbarui tindakan mitigasi yang ditentukan bernama `AddThingsToQuarantineGroupAction`, mengubah nama grup hal, dan menetapkan `overrideDynamicGroups` ke `false`. Anda dapat memverifikasi perubahan Anda dengan menggunakan `describe-mitigation-action` perintah.

```

aws iot update-mitigation-action \
  --cli-input-json "{ \"actionName\": \"AddThingsToQuarantineGroupAction\",
  \"actionParams\": { \"addThingsToThingGroupParams\": {\"thingGroupNames\":
  [\"QuarantineGroup2\"],\"overrideDynamicGroups\": false}}}"

```

Output:

```

{
  "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
  AddThingsToQuarantineGroupAction",
  "actionId": "2fd2726d-98e1-4abf-b10f-09465ccd6bfa"
}

```

Untuk informasi selengkapnya, lihat [UpdateMitigationAction \(Perintah Tindakan Mitigasi\) di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [UpdateMitigationAction](#) di Referensi AWS CLI Perintah.



## update-provisioning-template

Contoh kode berikut menunjukkan cara menggunakan `update-provisioning-template`.

### AWS CLI

Untuk memperbarui templat penyediaan

`update-provisioning-template` Contoh berikut memodifikasi deskripsi dan peran arn untuk template penyediaan yang ditentukan dan mengaktifkan template.

```
aws iot update-provisioning-template \  
  --template-name widget-template \  
  --enabled \  
  --description "An updated provisioning template for widgets" \  
  --provisioning-role-arn arn:aws:iam::504350838278:role/Provision_role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [AWS IoT Secure Tunneling di IoT Core Developer AWS Guide](#).

- Untuk detail API, lihat [UpdateProvisioningTemplate](#) di Referensi AWS CLI Perintah.

## update-role-alias

Contoh kode berikut menunjukkan cara menggunakan `update-role-alias`.

### AWS CLI

Untuk memperbarui alias peran

`update-role-alias` Contoh berikut memperbarui alias `LightBulbRole` peran.

```
aws iot update-role-alias \  
  --role-alias LightBulbRole \  
  --role-arn arn:aws:iam::123456789012:role/lightbulbrole-001
```

Output:

```
{  
  "roleAlias": "LightBulbRole",  
  "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/LightBulbRole"
```

```
}
```

Untuk informasi selengkapnya, lihat [UpdateRoleAlias](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [UpdateRoleAlias](#) di Referensi AWS CLI Perintah.

## update-scheduled-audit

Contoh kode berikut menunjukkan cara menggunakan `update-scheduled-audit`.

### AWS CLI

Untuk memperbarui definisi audit terjadwal

`update-scheduled-audit` Contoh berikut mengubah nama pemeriksaan target untuk audit terjadwal AWS IoT Device Defender.

```
aws iot update-scheduled-audit \  
  --scheduled-audit-name WednesdayCertCheck \  
  --target-check-names CA_CERTIFICATE_EXPIRING_CHECK  
  DEVICE_CERTIFICATE_EXPIRING_CHECK REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK
```

Output:

```
{  
  "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/  
  WednesdayCertCheck"  
}
```

Untuk informasi selengkapnya, lihat [Perintah Audit](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateScheduledAudit](#) di Referensi AWS CLI Perintah.

## update-security-profile

Contoh kode berikut menunjukkan cara menggunakan `update-security-profile`.

### AWS CLI

Untuk mengubah profil keamanan

`update-security-profile` Contoh berikut memperbarui deskripsi dan perilaku untuk profil keamanan AWS IoT Device Defender.

```
aws iot update-security-profile \
  --security-profile-name PossibleIssue \
  --security-profile-description "Check to see if authorization fails 12 times in
  5 minutes or if cellular bandwidth exceeds 128" \
  --behaviors "[{"name":"CellularBandwidth","metric":"aws:message-byte-size
  \","criteria":{"comparisonOperator":"greater-than","value":{"count":128},
  \\"consecutiveDatapointsToAlarm":1,\"consecutiveDatapointsToClear":1}},{"name
  \":"Authorization","metric":"aws:num-authorization-failures","criteria":
  {"comparisonOperator":"less-than","value":{"count":12},"durationSeconds
  \":300,\"consecutiveDatapointsToAlarm":1,\"consecutiveDatapointsToClear":1}}]"
```

### Output:

```
{
  "securityProfileName": "PossibleIssue",
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
  PossibleIssue",
  "securityProfileDescription": "check to see if authorization fails 12 times in 5
  minutes or if cellular bandwidth exceeds 128",
  "behaviors": [
    {
      "name": "CellularBandwidth",
      "metric": "aws:message-byte-size",
      "criteria": {
        "comparisonOperator": "greater-than",
        "value": {
          "count": 128
        },
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
      }
    },
    {
      "name": "Authorization",
      "metric": "aws:num-authorization-failures",
      "criteria": {
        "comparisonOperator": "less-than",
        "value": {
          "count": 12
        },
        "durationSeconds": 300,
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
      }
    }
  ]
}
```

```
    }
  }
],
"version": 2,
"creationDate": 1560278102.528,
"lastModifiedDate": 1560352711.207
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateSecurityProfile](#) di Referensi AWS CLI Perintah.

## update-stream

Contoh kode berikut menunjukkan cara menggunakan `update-stream`.

### AWS CLI

Untuk memperbarui aliran

`update-stream` Contoh berikut memperbarui aliran yang ada. Versi streaming bertambah satu.

```
aws iot update-stream \
  --cli-input-json file://update-stream.json
```

Isi dari `update-stream.json`:

```
{
  "streamId": "stream12345",
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "files": [
    {
      "fileId": 123,
      "s3Location": {
        "bucket": "codesign-ota-bucket",
        "key": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
      }
    }
  ]
  "roleArn": "arn:aws:iam:us-west-2:123456789012:role/service-role/my_ota_stream_role"
}
```

**Output:**

```
{
  "streamId": "stream12345",
  "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "streamVersion": 2
}
```

Untuk informasi selengkapnya, lihat [UpdateStream](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [UpdateStream](#) di Referensi AWS CLI Perintah.

**update-thing-group**

Contoh kode berikut menunjukkan cara menggunakan `update-thing-group`.

**AWS CLI**

Untuk memperbarui definisi untuk grup sesuatu

`update-thing-group` Contoh berikut memperbarui definisi untuk kelompok hal yang ditentukan, mengubah deskripsi dan dua atribut.

```
aws iot update-thing-group \
  --thing-group-name HalogenBulbs \
  --thing-group-properties "thingGroupDescription=\"Halogen bulb group\"",
  attributePayload={attributes={Manufacturer=AnyCompany,wattage=60}}"
```

**Output:**

```
{
  "version": 2
}
```

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateThingGroup](#) di Referensi AWS CLI Perintah.

**update-thing-groups-for-thing**

Contoh kode berikut menunjukkan cara menggunakan `update-thing-groups-for-thing`.

## AWS CLI

Untuk mengubah kelompok-kelompok yang menjadi milik suatu benda

`update-thing-groups-for-thing` Contoh berikut menghapus hal yang dinamai `MyLightBulb` dari grup bernama `DeadBulbs` dan menambahkannya ke grup bernama `replaceableItems` pada saat yang sama.

```
aws iot update-thing-groups-for-thing \  
  --thing-name MyLightBulb \  
  --thing-groups-to-add "replaceableItems" \  
  --thing-groups-to-remove "DeadBulbs"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Thing Groups](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateThingGroupsForThing](#) di Referensi AWS CLI Perintah.

## update-thing

Contoh kode berikut menunjukkan cara menggunakan `update-thing`.

## AWS CLI

Untuk mengasosiasikan sesuatu dengan tipe benda

`update-thing` Contoh berikut mengaitkan sesuatu dalam registri AWS IoT dengan tipe benda. Saat Anda membuat asosiasi, Anda memberikan nilai untuk atribut yang ditentukan oleh tipe benda.

```
aws iot update-thing \  
  --thing-name "MyOtherLightBulb" \  
  --thing-type-name "LightBulb" \  
  --attribute-payload '{"attributes": {"wattage": "75", "model": "123"}}'
```

Perintah ini tidak menghasilkan output. Gunakan `describe-thing` perintah untuk melihat hasilnya.

Untuk informasi selengkapnya, lihat [Tipe Hal](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateThing](#) di Referensi AWS CLI Perintah.

## update-topic-rule-destination

Contoh kode berikut menunjukkan cara menggunakan `update-topic-rule-destination`.

### AWS CLI

Contoh 1: Untuk mengaktifkan tujuan aturan topik

`update-topic-rule-destination` Contoh berikut memungkinkan lalu lintas ke tujuan aturan topik.

```
aws iot update-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \  
  --status ENABLED
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan tujuan aturan topik](#) di Panduan Pengembang AWS IoT.

Contoh 2: Untuk menonaktifkan tujuan aturan topik

`update-topic-rule-destination` Contoh berikut menonaktifkan lalu lintas ke tujuan aturan topik.

```
aws iot update-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \  
  --status DISABLED
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan tujuan aturan topik di Panduan](#) Pengembang AWS IoT.

Contoh 3: Untuk mengirim pesan konfirmasi baru

`update-topic-rule-destination` Contoh berikut mengirimkan pesan konfirmasi baru untuk tujuan aturan topik.

```
aws iot update-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \  
  --status ENABLED
```

```
--arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \
--status IN_PROGRESS
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengirim pesan konfirmasi baru](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateTopicRuleDestination](#) di Referensi AWS CLI Perintah.

## validate-security-profile-behaviors

Contoh kode berikut menunjukkan cara menggunakan `validate-security-profile-behaviors`.

### AWS CLI

Contoh 1: Untuk memvalidasi parameter perilaku untuk profil keamanan

`validate-security-profile-behaviors` Contoh berikut memvalidasi serangkaian perilaku yang terbentuk dengan baik dan benar untuk profil keamanan IoT AWS Device Defender.

```
aws iot validate-security-profile-behaviors \
  --behaviors "[{"name":"CellularBandwidth","metric":"aws:message-byte-size",
  "criteria":{"comparisonOperator":"greater-than","value":{"count":128},
  "consecutiveDatapointsToAlarm":1,"consecutiveDatapointsToClear":1}}, {"name
  ":"Authorization","metric":"aws:num-authorization-failures","criteria":
  {"comparisonOperator":"greater-than","value":{"count":12},"durationSeconds
  ":300,"consecutiveDatapointsToAlarm":1,"consecutiveDatapointsToClear":1}]]"
```

Output:

```
{
  "valid": true,
  "validationErrors": []
}
```

Contoh 2: Untuk memvalidasi parameter perilaku yang salah untuk profil keamanan

`validate-security-profile-behaviors` Contoh berikut memvalidasi serangkaian perilaku yang berisi kesalahan untuk profil keamanan AWS IoT Device Defender.



```
aws iot validate-security-profile-behaviors \
  --behaviors "[{\\"name\\":\\"CellularBandwidth\\",\\"metric\\":\\"aws:message-byte-size\\",\\"criteria\\":{\\"comparisonOperator\\":\\"greater-than\\",\\"value\\":{\\"count\\":128},\\"consecutiveDatapointsToAlarm\\":1,\\"consecutiveDatapointsToClear\\":1}},{\\"name\\":\\"Authorization\\",\\"metric\\":\\"aws:num-authorization-failures\\",\\"criteria\\":{\\"comparisonOperator\\":\\"greater-than\\",\\"value\\":{\\"count\\":12},\\"durationSeconds\\":300,\\"consecutiveDatapointsToAlarm\\":100000,\\"consecutiveDatapointsToClear\\":1}}]"
```

Output:

```
{
  "valid": false,
  "validationErrors": [
    {
      "errorMessage": "Behavior Authorization is malformed.
consecutiveDatapointsToAlarm 100000 should be in range[1,10]"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Perintah](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ValidateSecurityProfileBehaviors](#) di Referensi AWS CLI Perintah.

## AWS IoT 1-Click Contoh perangkat menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS IoT 1-Click perangkat AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **claim-devices-by-claim-code**

Contoh kode berikut menunjukkan cara menggunakan `claim-devices-by-claim-code`.

#### AWS CLI

Untuk mengklaim satu atau beberapa perangkat AWS IoT 1-Klik menggunakan kode klaim

`claim-devices-by-claim-code` Contoh berikut mengklaim perangkat AWS IoT 1-Klik yang ditentukan menggunakan kode klaim (bukan ID perangkat).

```
aws iot1click-devices claim-devices-by-claim-code \  
  --claim-code C-123EXAMPLE
```

#### Output:

```
{  
  "Total": 9  
  "ClaimCode": "C-123EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ClaimDevicesByClaimCode](#) di Referensi AWS CLI Perintah.

### **describe-device**

Contoh kode berikut menunjukkan cara menggunakan `describe-device`.

#### AWS CLI

Untuk menggambarkan perangkat

`describe-device` Contoh berikut menjelaskan perangkat yang ditentukan.

```
aws iot1click-devices describe-device \  
  --device-id C-123EXAMPLE
```

```
--device-id G030PM0123456789
```

Output:

```
{
  "DeviceDescription": {
    "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/G030PM0123456789",
    "Attributes": {
      "projectRegion": "us-west-2",
      "projectName": "AnytownDumpsters",
      "placementName": "customer217",
      "deviceTemplateName": "empty-dumpster-request"
    },
    "DeviceId": "G030PM0123456789",
    "Enabled": false,
    "RemainingLife": 99.9,
    "Type": "button",
    "Tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [DescribeDevice](#) di Referensi AWS CLI Perintah.

## **finalize-device-claim**

Contoh kode berikut menunjukkan cara menggunakan `finalize-device-claim`.

### AWS CLI

Untuk menyelesaikan permintaan klaim untuk perangkat IoT AWS 1-Klik menggunakan ID perangkat

`finalize-device-claim` Contoh berikut menyelesaikan permintaan klaim untuk perangkat IoT AWS 1-Klik yang ditentukan menggunakan ID perangkat (bukan kode klaim).

```
aws iot1click-devices finalize-device-claim \
  --device-id G030PM0123456789
```

Output:

```
{
  "State": "CLAIMED"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [FinalizeDeviceClaim](#) di Referensi AWS CLI Perintah.

## get-device-methods

Contoh kode berikut menunjukkan cara menggunakan `get-device-methods`.

### AWS CLI

Untuk membuat daftar metode yang tersedia untuk perangkat

`get-device-methods` Contoh berikut mencantumkan metode yang tersedia untuk perangkat.

```
aws iot1click-devices get-device-methods \
  --device-id G030PM0123456789
```

### Output:

```
{
  "DeviceMethods": [
    {
      "MethodName": "getDeviceHealthParameters"
    },
    {
      "MethodName": "setDeviceHealthMonitorCallback"
    },
    {
      "MethodName": "getDeviceHealthMonitorCallback"
    },
    {
      "MethodName": "setOnClickCallback"
    },
    {
      "MethodName": "getOnClickCallback"
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [GetDeviceMethods](#) di Referensi AWS CLI Perintah.

## **initiate-device-claim**

Contoh kode berikut menunjukkan cara menggunakan `initiate-device-claim`.

### AWS CLI

Untuk memulai permintaan klaim untuk perangkat IoT AWS 1-Klik menggunakan ID perangkat

`initiate-device-claim` Contoh berikut memulai permintaan klaim untuk perangkat IoT AWS 1-Klik yang ditentukan menggunakan ID perangkat (bukan kode klaim).

```
aws iot1click-devices initiate-device-claim \  
  --device-id G030PM0123456789
```

Output:

```
{  
  "State": "CLAIM_INITIATED"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [InitiateDeviceClaim](#) di Referensi AWS CLI Perintah.

## **invoke-device-method**

Contoh kode berikut menunjukkan cara menggunakan `invoke-device-method`.

### AWS CLI

Untuk memanggil metode perangkat di perangkat

`invoke-device-method` Contoh berikut memanggil metode yang ditentukan pada perangkat.

```
aws iot1click-devices invoke-device-method \  
  --cli-input-json file://invoke-device-method.json
```

Isi dari `invoke-device-method.json`:

```
{  
  "DeviceId": "G030PM0123456789",  
  "DeviceMethod": {  
    "DeviceType": "device",  
    "MethodName": "getDeviceHealthParameters"  
  }  
}
```

Output:

```
{  
  "DeviceMethodResponse": "{\"remainingLife\": 99.8}"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [InvokeDeviceMethod](#) di Referensi AWS CLI Perintah.

## list-device-events

Contoh kode berikut menunjukkan cara menggunakan `list-device-events`.

### AWS CLI

Untuk membuat daftar peristiwa perangkat untuk rentang waktu tertentu

`list-device-events` Contoh berikut mencantumkan peristiwa perangkat yang ditentukan untuk rentang waktu yang ditentukan.

```
aws iot1click-devices list-device-events \  
  --device-id G030PM0123456789 \  
  --from-time-stamp 2019-07-17T15:45:12.880Z --to-time-stamp  
  2019-07-19T15:45:12.880Z
```

Output:

```
{
  "Events": [
    {
      "Device": {
        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
      },
      "StdEvent": "{\"clickType\": \"SINGLE\"",
      "\"reportedTime\": \"2019-07-18T23:47:55.015Z\", \"certificateId\": \"fe8798a6c97c62ef8756b80eeefdcd2280f3352f82faa8080c74cc4f4a4d1811\", \"remainingLife\": 99.85000000000001, \"testMode\": false}"
    },
    {
      "Device": {
        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
      },
      "StdEvent": "{\"clickType\": \"DOUBLE\"",
      "\"reportedTime\": \"2019-07-19T00:14:41.353Z\", \"certificateId\": \"fe8798a6c97c62ef8756b80eeefdcd2280f3352f82faa8080c74cc4f4a4d1811\", \"remainingLife\": 99.8, \"testMode\": false}"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ListDeviceEvents](#) di Referensi AWS CLI Perintah.

## list-devices

Contoh kode berikut menunjukkan cara menggunakan `list-devices`.

### AWS CLI

Untuk membuat daftar perangkat dari jenis tertentu

`list-devices` Contoh berikut mencantumkan perangkat dari jenis tertentu.

```
aws iot1click-devices list-devices \
```

```
--device-type button
```

Perintah ini tidak menghasilkan output.

Output:

```
{
  "Devices": [
    {
      "remainingLife": 99.9,
      "attributes": {
        "arn": "arn:aws:iot1click:us-west-2:123456789012:devices/
G030PM0123456789",
        "type": "button",
        "deviceId": "G030PM0123456789",
        "enabled": false
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ListDevices](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

AWS CLI

Untuk membuat daftar tag untuk perangkat

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk perangkat yang ditentukan.

```
aws iot1click-devices list-tags-for-resource \
  --resource-arn "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789"
```

Output:



```
{
  "Tags": {
    "Driver Phone": "123-555-0199",
    "Driver": "Jorge Souza"
  }
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber AWS daya perangkat

`tag-resource` Contoh berikut menambahkan dua tag ke sumber daya yang ditentukan.

```
aws iot1click-devices tag-resource \
  --cli-input-json file://devices-tag-resource.json
```

Isi dari `devices-tag-resource.json`:

```
{
  "ResourceArn": "arn:aws:iot1click:us-west-2:123456789012:devices/
G030PM0123456789",
  "Tags": {
    "Driver": "Jorge Souza",
    "Driver Phone": "123-555-0199"
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **unclaim-device**

Contoh kode berikut menunjukkan cara menggunakan `unclaim-device`.

### AWS CLI

Untuk membatalkan klaim (deregister) perangkat dari akun Anda AWS

`unclaim-device` Contoh berikut membatalkan klaim (deregisters) perangkat yang ditentukan dari akun Anda. AWS

```
aws iot1click-devices unclaim-device \  
  --device-id G030PM0123456789
```

Output:

```
{  
  "State": "UNCLAIMED"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [UnclaimDevice](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber AWS daya perangkat

`untag-resource` Contoh berikut menghapus tag dengan nama `Driver Phone` dan `Driver` dari sumber daya perangkat yang ditentukan.

```
aws iot1click-devices untag-resource \  
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/  
  AnytownDumpsters" \  
  --tag-key DriverPhone
```

```
--tag-keys "Driver Phone" "Driver"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-device-state

Contoh kode berikut menunjukkan cara menggunakan `update-device-state`.

### AWS CLI

Untuk memperbarui status ``diaktifkan`` untuk perangkat

Berikut ini `update-device-state` menetapkan status perangkat yang ditentukan ke `enabled`.

```
aws iot1click-devices update-device-state \  
  --device-id G030PM0123456789 \  
  --enabled
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [UpdateDeviceState](#) di Referensi AWS CLI Perintah.

## AWS IoT 1-Click Contoh proyek menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT 1-Click Projects.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **associate-device-with-placement**

Contoh kode berikut menunjukkan cara menggunakan `associate-device-with-placement`.

AWS CLI

Untuk mengaitkan perangkat AWS IoT 1-Klik dengan penempatan yang ada

`associate-device-with-placement` Contoh berikut mengaitkan perangkat IoT AWS 1-Klik yang ditentukan dengan penempatan yang ada.

```
aws iot1click-projects associate-device-with-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217 \  
  --device-template-name empty-dumpster-request \  
  --device-id G030PM0123456789
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [AssociateDeviceWithPlacement](#) di Referensi AWS CLI Perintah.

## **create-placement**

Contoh kode berikut menunjukkan cara menggunakan `create-placement`.

AWS CLI

Untuk membuat penempatan 1-Klik AWS IoT untuk sebuah proyek

`create-placement` Contoh berikut membuat penempatan AWS IoT 1-Klik untuk proyek yang ditentukan.

```
aws iot1click-projects create-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217 \  
  --attributes '{"location": "123 Any Street Anytown, USA 10001", "phone":  
  "123-456-7890"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [CreatePlacement](#) di Referensi AWS CLI Perintah.

## create-project

Contoh kode berikut menunjukkan cara menggunakan `create-project`.

### AWS CLI

Untuk membuat proyek AWS IoT 1-Click untuk penempatan nol atau lebih

`create-project` Contoh berikut membuat proyek AWS IoT 1-Click untuk penempatan.

```
aws iot1click-projects create-project -- file: //create-project.json cli-input-json
```

Isi dari `create-project.json`:

```
{  
  "projectName": "AnytownDumpsters",  
  "description": "All dumpsters in the Anytown region.",  
  "placementTemplate": {  
    "defaultAttributes": {  
      "City" : "Anytown"  
    },  
    "deviceTemplates": {  
      "empty-dumpster-request" : {  
        "deviceType": "button"  
      }  
    }  
  }  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [CreateProject](#) di Referensi AWS CLI Perintah.

## **delete-placement**

Contoh kode berikut menunjukkan cara menggunakan `delete-placement`.

### AWS CLI

Untuk menghapus penempatan dari proyek

`delete-placement` Contoh berikut menghapus penempatan yang ditentukan dari proyek.

```
aws iot1click-projects delete-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [DeletePlacement](#) di Referensi AWS CLI Perintah.

## **delete-project**

Contoh kode berikut menunjukkan cara menggunakan `delete-project`.

### AWS CLI

Untuk menghapus proyek dari AWS akun Anda

`delete-project` Contoh berikut menghapus proyek yang ditentukan dari AWS akun Anda.

```
aws iot1click-projects delete-project \  
  --project-name AnytownDumpsters
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [DeleteProject](#) di Referensi AWS CLI Perintah.

## describe-placement

Contoh kode berikut menunjukkan cara menggunakan `describe-placement`.

### AWS CLI

Untuk menggambarkan penempatan untuk sebuah proyek

`describe-placement` Contoh berikut menjelaskan penempatan untuk proyek yang ditentukan.

```
aws iot1click-projects describe-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217
```

Output:

```
{  
  "placement": {  
    "projectName": "AnytownDumpsters",  
    "placementName": "customer217",  
    "attributes": {  
      "phone": "123-555-0110",  
      "location": "123 Any Street Anytown, USA 10001"  
    },  
    "createdDate": 1563488454,  
    "updatedDate": 1563488454  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [DescribePlacement](#) di Referensi AWS CLI Perintah.

## describe-project

Contoh kode berikut menunjukkan cara menggunakan `describe-project`.

## AWS CLI

Untuk mendeskripsikan proyek AWS IoT 1-Click

`describe-project` Contoh berikut menjelaskan proyek AWS IoT 1-Click yang ditentukan.

```
aws iot1click-projects describe-project \  
--project-name AnytownDumpsters
```

Output:

```
{  
  "project": {  
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/AnytownDumpsters",  
    "projectName": "AnytownDumpsters",  
    "description": "All dumpsters in the Anytown region.",  
    "createdDate": 1563483100,  
    "updatedAt": 1563483100,  
    "placementTemplate": {  
      "defaultAttributes": {  
        "City": "Anytown"  
      },  
      "deviceTemplates": {  
        "empty-dumpster-request": {  
          "deviceType": "button",  
          "callbackOverrides": {}  
        }  
      }  
    },  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [DescribeProject](#) di Referensi AWS CLI Perintah.

## **disassociate-device-from-placement**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-device-from-placement`.



## AWS CLI

Untuk memisahkan perangkat dari penempatan

`disassociate-device-from-placement` Contoh berikut memisahkan perangkat yang ditentukan dari penempatan.

```
aws iot1click-projects disassociate-device-from-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217 \  
  --device-template-name empty-dumpster-request
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [DisassociateDeviceFromPlacement](#) di Referensi AWS CLI Perintah.

## get-devices-in-placement

Contoh kode berikut menunjukkan cara menggunakan `get-devices-in-placement`.

### AWS CLI

Untuk mencantumkan semua perangkat dalam penempatan yang terdapat dalam proyek

`get-devices-in-placement` Contoh berikut mencantumkan semua perangkat dalam penempatan tertentu yang terkandung dalam proyek yang ditentukan.

```
aws iot1click-projects get-devices-in-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217
```

Output:

```
{  
  "devices": {  
    "empty-dumpster-request": "G030PM0123456789"  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [GetDevicesInPlacement](#) di Referensi AWS CLI Perintah.

## list-placements

Contoh kode berikut menunjukkan cara menggunakan `list-placements`.

### AWS CLI

Untuk mencantumkan semua AWS penempatan 1-Klik IoT untuk sebuah proyek

`list-placements` Contoh berikut mencantumkan semua penempatan AWS IoT 1-Klik untuk proyek yang ditentukan.

```
aws iot1click-projects list-placements \  
  --project-name AnytownDumpsters
```

Output:

```
{  
  "placements": [  
    {  
      "projectName": "AnytownDumpsters",  
      "placementName": "customer217",  
      "createdDate": 1563488454,  
      "updatedDate": 1563488454  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ListPlacements](#) di Referensi AWS CLI Perintah.

## list-projects

Contoh kode berikut menunjukkan cara menggunakan `list-projects`.

## AWS CLI

Untuk membuat daftar semua proyek AWS IoT 1-Klik

`list-projects` Contoh berikut mencantumkan semua proyek AWS IoT 1-Klik di akun Anda.

```
aws iot1click-projects list-projects
```

Output:

```
{
  "projects": [
    {
      "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/
AnytownDumpsters",
      "projectName": "AnytownDumpsters",
      "createdDate": 1563483100,
      "updatedDate": 1563483100,
      "tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ListProjects](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk membuat daftar tag untuk sumber daya proyek

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk sumber daya proyek yang ditentukan.

```
aws iot1click-projects list-tags-for-resource \
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/
AnytownDumpsters"
```

**Output:**

```
{
  "tags": {
    "Manager": "Li Juan",
    "Account": "45215"
  }
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

**tag-resource**

Contoh kode berikut menunjukkan cara menggunakan tag-resource.

**AWS CLI**

Untuk menambahkan tag ke sumber daya proyek

tag-resource Contoh berikut menambahkan dua tag ke sumber daya proyek yang ditentukan.

```
aws iot1click-projects tag-resource \
  --cli-input-json file:///devices-tag-resource.json
```

Isi dari devices-tag-resource.json:

```
{
  "resourceArn": "arn:aws:iot1click:us-west-2:123456789012:projects/
AnytownDumpsters",
  "tags": {
    "Account": "45215",
    "Manager": "Li Juan"
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya proyek

`untag-resource` Contoh berikut menghapus tag dengan nama kunci `Manager` dari proyek yang ditentukan.

```
aws iot1click-projects untag-resource \  
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/  
AnytownDumpsters" \  
  --tag-keys "Manager"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-placement

Contoh kode berikut menunjukkan cara menggunakan `update-placement`.

### AWS CLI

Untuk memperbarui pasangan nilai kunci “atribut” penempatan

`update-placement` Contoh berikut memperbarui “atribut” pasangan kunci-nilai penempatan.

```
aws iot1click-projects update-placement \  
  --cli-input-json file://update-placement.json
```

Isi dari `update-placement.json`:

```
{
```

```
"projectName": "AnytownDumpsters",
"placementName": "customer217",
"attributes": {
  "phone": "123-456-7890",
  "location": "123 Any Street Anytown, USA 10001"
}
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [UpdatePlacement](#) di Referensi AWS CLI Perintah.

## update-project

Contoh kode berikut menunjukkan cara menggunakan `update-project`.

### AWS CLI

Untuk memperbarui pengaturan untuk proyek

`update-project` Contoh berikut memperbarui deskripsi untuk sebuah proyek.

```
aws iot1click-projects update-project \
  --project-name AnytownDumpsters \
  --description "All dumpsters (yard waste, recycling, garbage) in the Anytown
region."
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan AWS IoT 1-Klik dengan AWS CLI di Panduan Pengembang](#) 1-Klik AWS IoT.

- Untuk detail API, lihat [UpdateProject](#) di Referensi AWS CLI Perintah.

## AWS IoT Analytics contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Analytics.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **batch-put-message**

Contoh kode berikut menunjukkan cara menggunakan `batch-put-message`.

AWS CLI

Untuk mengirim pesan ke saluran

`batch-put-message` Contoh berikut mengirimkan pesan ke saluran yang ditentukan.

```
aws iotanalytics batch-put-message \  
  --cli-binary-format raw-in-base64-out \  
  --cli-input-json file://batch-put-message.json
```

Isi dari `batch-put-message.json`:

```
{  
  "channelName": "mychannel",  
  "messages": [  
    {  
      "messageId": "0001",  
      "payload": "eyJhdGVtcGVyYXR1cmUiOiAyMCB9"  
    }  
  ]  
}
```

Output:

```
{
  "batchPutMessageErrorEntries": []
}
```

Untuk informasi selengkapnya, lihat [BatchPutMessage](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [BatchPutMessage](#) di Referensi AWS CLI Perintah.

## cancel-pipeline-reprocessing

Contoh kode berikut menunjukkan cara menggunakan `cancel-pipeline-reprocessing`.

### AWS CLI

Untuk membatalkan pemrosesan ulang data melalui pipa

`cancel-pipeline-reprocessing` Contoh berikut membatalkan pemrosesan ulang data melalui pipeline yang ditentukan.

```
aws iotanalytics cancel-pipeline-reprocessing \
  --pipeline-name mypipeline \
  --reprocessing-id "6ad2764f-fb13-4de3-b101-4e74af03b043"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [CancelPipelineReprocessing](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [CancelPipelineReprocessing](#) di Referensi AWS CLI Perintah.

## create-channel

Contoh kode berikut menunjukkan cara menggunakan `create-channel`.

### AWS CLI

Untuk membuat saluran

`create-channel` Contoh berikut membuat saluran dengan konfigurasi yang ditentukan. Saluran mengumpulkan data dari topik MQTT dan mengarsipkan pesan mentah dan belum diproses sebelum menerbitkan data ke alur.



```
aws iotanalytics create-channel \  
  --cli-input-json file://create-channel.json
```

Isi dari `create-channel.json`:

```
{  
  "channelName": "mychannel",  
  "retentionPeriod": {  
    "unlimited": true  
  },  
  "tags": [  
    {  
      "key": "Environment",  
      "value": "Production"  
    }  
  ]  
}
```

Output:

```
{  
  "channelArn": "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel",  
  "channelName": "mychannel",  
  "retentionPeriod": {  
    "unlimited": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [CreateChannel](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [CreateChannel](#) di Referensi AWS CLI Perintah.

## **create-dataset-content**

Contoh kode berikut menunjukkan cara menggunakan `create-dataset-content`.

### AWS CLI

Untuk membuat konten dari dataset

`create-dataset-content` Contoh berikut membuat konten dari dataset tertentu dengan menerapkan `queryAction` (query SQL) atau `containerAction` (mengeksekusi aplikasi containerized).

```
aws iotanalytics create-dataset-content \  
  --dataset-name mydataset
```

Output:

```
{  
  "versionId": "d494b416-9850-4670-b885-ca22f1e89d62"  
}
```

Untuk informasi selengkapnya, lihat [CreateDatasetContent](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [CreateDatasetContent](#) di Referensi AWS CLI Perintah.

## **create-dataset**

Contoh kode berikut menunjukkan cara menggunakan `create-dataset`.

### AWS CLI

Untuk membuat dataset

`create-dataset` Contoh berikut membuat dataset. Sebuah dataset menyimpan data yang diambil dari penyimpanan data dengan menerapkan `queryAction` (query SQL) atau `containerAction` (mengeksekusi aplikasi containerized). Operasi ini menciptakan kerangka dataset. Anda dapat mengisi kumpulan data secara manual dengan menelepon `CreateDatasetContent` atau secara otomatis sesuai dengan yang `trigger` Anda tentukan.

```
aws iotanalytics create-dataset \  
  --cli-input-json file://create-dataset.json
```

Isi dari `create-dataset.json`:

```
{  
  "datasetName": "mydataset",  
  "actions": [  
    {  
      "actionType": "query",  
      "query": "SELECT * FROM mydataset"  
    }  
  ]  
}
```

```
{
  "actionName": "myDatasetAction",
  "queryAction": {
    "sqlQuery": "SELECT * FROM mydatastore"
  }
},
"retentionPeriod": {
  "unlimited": true
},
"tags": [
  {
    "key": "Environment",
    "value": "Production"
  }
]
}
```

Output:

```
{
  "datasetName": "mydataset",
  "retentionPeriod": {
    "unlimited": true
  },
  "datasetArn": "arn:aws:iotanalytics:us-west-2:123456789012:dataset/mydataset"
}
```

Untuk informasi selengkapnya, lihat [CreateDataset](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [CreateDataset](#) di Referensi AWS CLI Perintah.

## create-datastore

Contoh kode berikut menunjukkan cara menggunakan `create-datastore`.

### AWS CLI

Untuk membuat penyimpanan data

`create-datastore` Contoh berikut membuat penyimpanan data, yang merupakan repositori untuk pesan.

```
aws iotanalytics create-datastore \  
  --cli-input-json file://create-datastore.json
```

Isi dari `create-datastore.json`:

```
{  
  "datastoreName": "mydatastore",  
  "retentionPeriod": {  
    "numberOfDays": 90  
  },  
  "tags": [  
    {  
      "key": "Environment",  
      "value": "Production"  
    }  
  ]  
}
```

Output:

```
{  
  "datastoreName": "mydatastore",  
  "datastoreArn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/  
mydatastore",  
  "retentionPeriod": {  
    "numberOfDays": 90,  
    "unlimited": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [CreateDatastore](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [CreateDatastore](#) di Referensi AWS CLI Perintah.

## create-pipeline

Contoh kode berikut menunjukkan cara menggunakan `create-pipeline`.

### AWS CLI

#### Membuat Pipeline IoT Analytics

`create-pipeline` Contoh berikut membuat pipeline. Pipeline mengkonsumsi pesan dari saluran dan memungkinkan Anda memproses pesan sebelum menyimpannya di penyimpanan data. Anda harus menentukan saluran dan aktivitas penyimpanan data dan, secara opsional, sebanyak 23 aktivitas tambahan dalam `pipelineActivities` array.

```
aws iotanalytics create-pipeline \  
  --cli-input-json file://create-pipeline.json
```

Isi dari `create-pipeline.json`:

```
{  
  "pipelineName": "mypipeline",  
  "pipelineActivities": [  
    {  
      "channel": {  
        "name": "myChannelActivity",  
        "channelName": "mychannel",  
        "next": "myMathActivity"  
      }  
    },  
    {  
      "datastore": {  
        "name": "myDatastoreActivity",  
        "datastoreName": "mydatastore"  
      }  
    },  
    {  
      "math": {  
        "name": "myMathActivity",  
        "math": "((temp - 32) * 5.0) / 9.0",  
        "attribute": "tempC",  
        "next": "myDatastoreActivity"  
      }  
    }  
  ],  
  "tags": [  
    {  
      "key": "Environment",  
      "value": "Beta"  
    }  
  ]  
}
```

## Output:

```
{
  "pipelineArn": "arn:aws:iotanalytics:us-west-2:123456789012:pipeline/
mypipeline",
  "pipelineName": "mypipeline"
}
```

Untuk informasi selengkapnya, lihat [CreatePipeline](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [CreatePipeline](#) di Referensi AWS CLI Perintah.

## **delete-channel**

Contoh kode berikut menunjukkan cara menggunakan `delete-channel`.

### AWS CLI

Menghapus Saluran IoT Analytics

`delete-channel` Contoh berikut menghapus saluran yang ditentukan.

```
aws iotanalytics delete-channel \
  --channel-name mychannel
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteChannel](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DeleteChannel](#) di Referensi AWS CLI Perintah.

## **delete-dataset-content**

Contoh kode berikut menunjukkan cara menggunakan `delete-dataset-content`.

### AWS CLI

Untuk menghapus konten kumpulan data

`delete-dataset-content` Contoh berikut menghapus konten dari dataset yang ditentukan.

```
aws iotanalytics delete-dataset-content \
```

```
--dataset-name mydataset
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteDatasetContent](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DeleteDatasetContent](#) di Referensi AWS CLI Perintah.

## **delete-dataset**

Contoh kode berikut menunjukkan cara menggunakan `delete-dataset`.

### AWS CLI

Untuk menghapus dataset

`delete-dataset` Contoh berikut menghapus dataset yang ditentukan. Anda tidak perlu menghapus konten kumpulan data sebelum melakukan operasi ini.

```
aws iotanalytics delete-dataset \  
  --dataset-name mydataset
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteDataset](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DeleteDataset](#) di Referensi AWS CLI Perintah.

## **delete-datastore**

Contoh kode berikut menunjukkan cara menggunakan `delete-datastore`.

### AWS CLI

Untuk menghapus penyimpanan data

`delete-datastore` Contoh berikut menghapus penyimpanan data yang ditentukan.

```
aws iotanalytics delete-datastore \  
  --datastore-name mydatastore
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteDatastore](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DeleteDatastore](#) di Referensi AWS CLI Perintah.

## **delete-pipeline**

Contoh kode berikut menunjukkan cara menggunakan `delete-pipeline`.

### AWS CLI

Untuk menghapus pipeline

`delete-pipeline` Contoh berikut menghapus pipeline yang ditentukan.

```
aws iotanalytics delete-pipeline \  
  --pipeline-name mypipeline
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeletePipeline](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DeletePipeline](#) di Referensi AWS CLI Perintah.

## **describe-channel**

Contoh kode berikut menunjukkan cara menggunakan `describe-channel`.

### AWS CLI

Untuk mengambil informasi tentang saluran

`describe-channel` Contoh berikut menampilkan detail, termasuk statistik, untuk saluran yang ditentukan.

```
aws iotanalytics describe-channel \  
  --channel-name mychannel \  
  --include-statistics
```

Output:

```
{
```



```
"statistics": {
  "size": {
    "estimatedSizeInBytes": 402.0,
    "estimatedOn": 1561504380.0
  }
},
"channel": {
  "status": "ACTIVE",
  "name": "mychannel",
  "lastUpdateTime": 1557860351.001,
  "creationTime": 1557860351.001,
  "retentionPeriod": {
    "unlimited": true
  },
  "arn": "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
}
}
```

Untuk informasi selengkapnya, lihat [DescribeChannel](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DescribeChannel](#) di Referensi AWS CLI Perintah.

## describe-dataset

Contoh kode berikut menunjukkan cara menggunakan `describe-dataset`.

### AWS CLI

Untuk mengambil informasi tentang dataset

`describe-dataset` Contoh berikut menampilkan rincian untuk dataset tertentu.

```
aws iotanalytics describe-dataset \
  --dataset-name mydataset
```

Output:

```
{
  "dataset": {
    "status": "ACTIVE",
    "contentDeliveryRules": [],
    "name": "mydataset",
```

```

    "lastUpdateTime": 1557859240.658,
    "triggers": [],
    "creationTime": 1557859240.658,
    "actions": [
      {
        "actionName": "query_32",
        "queryAction": {
          "sqlQuery": "SELECT * FROM mydatastore",
          "filters": []
        }
      }
    ],
    "retentionPeriod": {
      "numberOfDays": 90,
      "unlimited": false
    },
    "arn": "arn:aws:iotanalytics:us-west-2:123456789012:dataset/mydataset"
  }
}

```

Untuk informasi selengkapnya, lihat [DescribeDataset](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DescribeDataset](#) di Referensi AWS CLI Perintah.

## describe-datastore

Contoh kode berikut menunjukkan cara menggunakan `describe-datastore`.

### AWS CLI

Untuk mengambil informasi tentang penyimpanan data

`describe-datastore` Contoh berikut menampilkan detail, termasuk statistik, untuk penyimpanan data yang ditentukan.

```

aws iotanalytics describe-datastore \
  --datastore-name mydatastore \
  --include-statistics

```

Output:

```
{
```

```
"datastore": {
  "status": "ACTIVE",
  "name": "mydatastore",
  "lastUpdateTime": 1557858971.02,
  "creationTime": 1557858971.02,
  "retentionPeriod": {
    "unlimited": true
  },
  "arn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/mydatastore"
},
"statistics": {
  "size": {
    "estimatedSizeInBytes": 397.0,
    "estimatedOn": 1561592040.0
  }
}
}
```

Untuk informasi selengkapnya, lihat [DescribeDatastore](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DescribeDatastore](#) di Referensi AWS CLI Perintah.

## describe-logging-options

Contoh kode berikut menunjukkan cara menggunakan `describe-logging-options`.

### AWS CLI

Untuk mengambil opsi logging saat ini

`describe-logging-options` Contoh berikut menampilkan opsi logging AWS IoT Analytics saat ini.

```
aws iotanalytics describe-logging-options
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/service-role/myIoTAnalyticsRole",
    "enabled": true,
    "level": "ERROR"
  }
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [DescribeLoggingOptions](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DescribeLoggingOptions](#) di Referensi AWS CLI Perintah.

## describe-pipeline

Contoh kode berikut menunjukkan cara menggunakan `describe-pipeline`.

### AWS CLI

Untuk mengambil informasi tentang pipa

`describe-pipeline` Contoh berikut menampilkan rincian untuk pipeline yang ditentukan.

```
aws iotanalytics describe-pipeline \  
  --pipeline-name mypipeline
```

Output:

```
{  
  "pipeline": {  
    "activities": [  
      {  
        "channel": {  
          "channelName": "mychannel",  
          "name": "mychannel_28",  
          "next": "mydatastore_29"  
        }  
      },  
      {  
        "datastore": {  
          "datastoreName": "mydatastore",  
          "name": "mydatastore_29"  
        }  
      }  
    ],  
    "name": "mypipeline",  
    "lastUpdateTime": 1561676362.515,  
    "creationTime": 1557859124.432,  
    "reprocessingSummaries": [  

```

```

    {
      "status": "SUCCEEDED",
      "creationTime": 1561676362.189,
      "id": "6ad2764f-fb13-4de3-b101-4e74af03b043"
    }
  ],
  "arn": "arn:aws:iotanalytics:us-west-2:123456789012:pipeline/mypipeline"
}

```

Untuk informasi selengkapnya, lihat [DescribePipeline](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [DescribePipeline](#) di Referensi AWS CLI Perintah.

## get-dataset-content

Contoh kode berikut menunjukkan cara menggunakan `get-dataset-content`.

### AWS CLI

Untuk mengambil isi dari dataset

`get-dataset-content` Contoh berikut mengambil isi dari dataset sebagai URI presigned.

```
aws iotanalytics get-dataset-content --dataset-name mydataset
```

Output:

```

{
  "status": {
    "state": "SUCCEEDED"
  },
  "timestamp": 1557863215.995,
  "entries": [
    {
      "dataURI": "https://aws-radiant-
dataset-12345678-1234-1234-1234-123456789012.s3.us-west-2.amazonaws.com/
results/12345678-e8b3-46ba-b2dd-efe8d86cf385.csv?X-Amz-Security-Token=...-Amz-
Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20190628T173437Z&X-Amz-SignedHeaders=host&X-
Amz-Expires=7200&X-Amz-Credential=...F20190628%2Fus-west-2%2Fs3%2Faws4_request&X-
Amz-Signature=..."
    }
  ]
}

```

```
}
```

Untuk informasi lebih lanjut, lihat [GetDatasetContent](#) di panduan.

- Untuk detail API, lihat [GetDatasetContent](#) di Referensi AWS CLI Perintah.

## list-channels

Contoh kode berikut menunjukkan cara menggunakan `list-channels`.

### AWS CLI

Untuk mengambil daftar saluran

`list-channels` Contoh berikut menampilkan informasi ringkasan untuk saluran yang tersedia.

```
aws iotanalytics list-channels
```

Output:

```
{
  "channelSummaries": [
    {
      "status": "ACTIVE",
      "channelName": "mychannel",
      "creationTime": 1557860351.001,
      "lastUpdateTime": 1557860351.001
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ListChannels](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [ListChannels](#) di Referensi AWS CLI Perintah.

## list-dataset-contents

Contoh kode berikut menunjukkan cara menggunakan `list-dataset-contents`.

### AWS CLI

Untuk membuat daftar informasi tentang konten dataset

`list-dataset-contents` Contoh berikut mencantumkan informasi tentang konten dataset yang telah dibuat.

```
aws iotanalytics list-dataset-contents \  
  --dataset-name mydataset
```

Output:

```
{  
  "datasetContentSummaries": [  
    {  
      "status": {  
        "state": "SUCCEEDED"  
      },  
      "scheduleTime": 1557863215.995,  
      "version": "b10ea2a9-66c1-4d99-8d1f-518113b738d0",  
      "creationTime": 1557863215.995  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListDatasetContents](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [ListDatasetContents](#) di Referensi AWS CLI Perintah.

## list-datasets

Contoh kode berikut menunjukkan cara menggunakan `list-datasets`.

AWS CLI

Untuk mengambil informasi tentang dataset

`list-datasets` Contoh berikut mencantumkan informasi ringkasan tentang kumpulan data yang tersedia.

```
aws iotanalytics list-datasets
```

Output:

```
{
```

```
"datasetSummaries": [  
  {  
    "status": "ACTIVE",  
    "datasetName": "mydataset",  
    "lastUpdateTime": 1557859240.658,  
    "triggers": [],  
    "creationTime": 1557859240.658,  
    "actions": [  
      {  
        "actionName": "query_32",  
        "actionType": "QUERY"  
      }  
    ]  
  }  
]
```

Untuk informasi selengkapnya, lihat [ListDatasets](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [ListDatasets](#) di Referensi AWS CLI Perintah.

## list-datastores

Contoh kode berikut menunjukkan cara menggunakan `list-datastores`.

### AWS CLI

Untuk mengambil daftar penyimpanan data

`list-datastores` Contoh berikut menampilkan informasi ringkasan tentang penyimpanan data yang tersedia.

```
aws iotanalytics list-datastores
```

Output:

```
{  
  "datastoreSummaries": [  
    {  
      "status": "ACTIVE",  
      "datastoreName": "mydatastore",  
      "creationTime": 1557858971.02,  
      "triggers": [],  
      "actions": []  
    }  
  ]  
}
```



```
        "lastUpdateTime": 1557858971.02
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [ListDatastores](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [ListDatastores](#) di Referensi AWS CLI Perintah.

## list-pipelines

Contoh kode berikut menunjukkan cara menggunakan `list-pipelines`.

### AWS CLI

Untuk mengambil daftar saluran pipa

`list-pipelines` Contoh berikut menampilkan daftar pipeline yang tersedia.

```
aws iotanalytics list-pipelines
```

Output:

```
{
  "pipelineSummaries": [
    {
      "pipelineName": "mypipeline",
      "creationTime": 1557859124.432,
      "lastUpdateTime": 1557859124.432,
      "reprocessingSummaries": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ListPipelines](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [ListPipelines](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut Daftar tag yang telah Anda lampirkan ke sumber daya yang ditentukan.

```
aws iotanalytics list-tags-for-resource \  
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
```

Output:

```
{  
  "tags": [  
    {  
      "value": "bar",  
      "key": "foo"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListTagsForResource](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-logging-options

Contoh kode berikut menunjukkan cara menggunakan `put-logging-options`.

### AWS CLI

Untuk mengatur atau memperbarui opsi pencatatan

`put-logging-options` Contoh berikut menetapkan atau memperbarui opsi pencatatan AWS IoT Analytics. Jika Anda memperbarui nilai `loggingOptions` bidang apa pun, perlu waktu hingga satu menit agar perubahan diterapkan. Selain itu, jika Anda mengubah kebijakan yang dilampirkan pada peran yang Anda tentukan di bidang “roLearn” (misalnya, untuk memperbaiki kebijakan yang tidak valid), perubahan tersebut dapat memakan waktu hingga lima menit agar perubahan tersebut diterapkan.

```
aws iotanalytics put-logging-options \  
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
```

```
--cli-input-json file://put-logging-options.json
```

Isi dari `put-logging-options.json`:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/service-role/myIoTAnalyticsRole",
    "level": "ERROR",
    "enabled": true
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [PutLoggingOptions](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [PutLoggingOptions](#) di Referensi AWS CLI Perintah.

## run-pipeline-activity

Contoh kode berikut menunjukkan cara menggunakan `run-pipeline-activity`.

### AWS CLI

Untuk mensimulasikan aktivitas pipa

`run-pipeline-activity` Contoh berikut mensimulasikan hasil menjalankan aktivitas pipeline pada payload pesan.

```
aws iotanalytics run-pipeline-activity \
  --pipeline-activity file://maths.json \
  --payloads file://payloads.json
```

Isi dari `maths.json`:

```
{
  "math": {
    "name": "MyMathActivity",
    "math": "((temp - 32) * 5.0) / 9.0",
    "attribute": "tempC"
  }
}
```

```
}

```

Isi dari `payloads.json`:

```
[
  {"humidity": 52, "temp": 68 },
  {"humidity": 52, "temp": 32 }
]
```

Output:

```
{
  "logResult": "",
  "payloads": [
    "eyJodW1pZG10eSI6NTIsInRlbXAiOiY4LCJ0ZW1wQyI6MjB9",
    "eyJodW1pZG10eSI6NTIsInRlbXAiOiMyLCJ0ZW1wQyI6MH0="
  ]
}
```

Untuk informasi selengkapnya, lihat [RunPipelineActivity](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [RunPipelineActivity](#) di Referensi AWS CLI Perintah.

## sample-channel-data

Contoh kode berikut menunjukkan cara menggunakan `sample-channel-data`.

### AWS CLI

Untuk mengambil contoh pesan dari saluran

`sample-channel-data` Contoh berikut mengambil sampel pesan dari saluran tertentu yang dicerna selama jangka waktu yang ditentukan. Anda dapat mengambil hingga 10 pesan.

```
aws iotanalytics sample-channel-data \
  --channel-name mychannel
```

Output:

```
{
  "payloads": [
```

```

    "eyJmGVtcGVyYXR1cmUiOiAyMCA9",
    "eyJmZm9vIjogImJhciIgfQ=="
  ]
}

```

Untuk informasi selengkapnya, lihat [SampleChannelData](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [SampleChannelData](#) di Referensi AWS CLI Perintah.

## start-pipeline-reprocessing

Contoh kode berikut menunjukkan cara menggunakan `start-pipeline-reprocessing`.

### AWS CLI

Untuk memulai pemrosesan ulang pipa

`start-pipeline-reprocessing` Contoh berikut memulai pemrosesan ulang data pesan mentah melalui pipeline yang ditentukan.

```

aws iotanalytics start-pipeline-reprocessing \
  --pipeline-name mypipeline

```

Output:

```

{
  "reprocessingId": "6ad2764f-fb13-4de3-b101-4e74af03b043"
}

```

Untuk informasi selengkapnya, lihat [StartPipelineReprocessing](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [StartPipelineReprocessing](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambah atau memodifikasi tag untuk sumber daya

`tag-resource` Contoh berikut menambah atau memodifikasi tag yang dilampirkan ke sumber daya yang ditentukan.

```
aws iotanalytics tag-resource \  
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel" \  
  --tags "[{"key": "Environment", "value": "Production"}]"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [TagResource](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dengan nama kunci yang ditentukan dari sumber daya yang ditentukan.

```
aws iotanalytics untag-resource \  
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel" \  
  --tag-keys ["Environment"]
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UntagResource < https://docs.aws.amazon.com/iotanalytics/latest/APIReference/API\\_UntagResource .html >](https://docs.aws.amazon.com/iotanalytics/latest/APIReference/API_UntagResource.html) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-channel**

Contoh kode berikut menunjukkan cara menggunakan `update-channel`.

### AWS CLI

Untuk memodifikasi saluran

`update-channel` Contoh berikut memodifikasi pengaturan untuk saluran yang ditentukan.

```
aws iotanalytics update-channel \  
  --cli-input-json file://update-channel.json
```

Isi dari `update-channel.json`:

```
{  
  "channelName": "mychannel",  
  "retentionPeriod": {  
    "numberOfDays": 92  
  }  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UpdateChannel](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [UpdateChannel](#) di Referensi AWS CLI Perintah.

## update-dataset

Contoh kode berikut menunjukkan cara menggunakan `update-dataset`.

### AWS CLI

Untuk memperbarui kumpulan data

`update-dataset` Contoh berikut memodifikasi pengaturan dataset yang ditentukan.

```
aws iotanalytics update-dataset \  
  --cli-input-json file://update-dataset.json
```

Isi dari `update-dataset.json`:

```
{  
  "datasetName": "mydataset",  
  "actions": [  
    {  
      "actionName": "myDatasetUpdateAction",  
      "queryAction": {
```

```
        "sqlQuery": "SELECT * FROM mydatastore"
      }
    ],
    "retentionPeriod": {
      "numberOfDays": 92
    }
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat `UpdateDataset` < [https://docs.aws.amazon.com/iotanalytics/latest/APIReference/API\\_UpdateDataset.html](https://docs.aws.amazon.com/iotanalytics/latest/APIReference/API_UpdateDataset.html) > di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [UpdateDataset](#) di Referensi AWS CLI Perintah.

## update-datastore

Contoh kode berikut menunjukkan cara menggunakan `update-datastore`.

### AWS CLI

Untuk memperbarui penyimpanan data

`update-datastore` Contoh berikut memodifikasi pengaturan penyimpanan data yang ditentukan.

```
aws iotanalytics update-datastore \
  --cli-input-json file://update-datastore.json
```

Isi `update-datastore.json`:

```
{
  "datastoreName": "mydatastore",
  "retentionPeriod": {
    "numberOfDays": 93
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UpdateDatastore](#) di Referensi API AWS IoT Analytics.



- Untuk detail API, lihat [UpdateDatastore](#) di Referensi AWS CLI Perintah.

## update-pipeline

Contoh kode berikut menunjukkan cara menggunakan `update-pipeline`.

### AWS CLI

Untuk memperbarui pipeline

`update-pipeline` Contoh berikut memodifikasi pengaturan pipa yang ditentukan. Anda harus menentukan saluran dan aktivitas penyimpanan data dan, secara opsional, sebanyak 23 aktivitas tambahan, dalam `pipelineActivities` larik.

```
aws iotanalytics update-pipeline \  
  --cli-input-json file://update-pipeline.json
```

Isi `update-pipeline.json`:

```
{  
  "pipelineName": "mypipeline",  
  "pipelineActivities": [  
    {  
      "channel": {  
        "name": "myChannelActivity",  
        "channelName": "mychannel",  
        "next": "myMathActivity"  
      }  
    },  
    {  
      "datastore": {  
        "name": "myDatastoreActivity",  
        "datastoreName": "mydatastore"  
      }  
    },  
    {  
      "math": {  
        "name": "myMathActivity",  
        "math": "(((temp - 32) * 5.0) / 9.0) + 273.15",  
        "attribute": "tempK",  
        "next": "myDatastoreActivity"  
      }  
    }  
  ]  
}
```

```
}  
  ]  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UpdatePipeline](#) di Referensi API AWS IoT Analytics.

- Untuk detail API, lihat [UpdatePipeline](#) di Referensi AWS CLI Perintah.

## Contoh Device Advisor menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Device Advisor.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-suite-definition**

Contoh kode berikut menunjukkan cara menggunakan `create-suite-definition`.

AWS CLI

Contoh 1: Untuk membuat rangkaian pengujian IoT Device Advisor

`create-suite-definition` Contoh berikut membuat rangkaian pengujian penasihat perangkat di AWS IoT dengan konfigurasi definisi suite yang ditentukan.

```
aws iotdeviceadvisor create-suite-definition \
  --suite-definition-configuration '{ \
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/MyIotThing"}], \
    "intendedForQualification": false, \
    "rootGroup": "{\"configuration\":{\"tests\":[{\"name\":\"MQTT Connect\"}, \
    \"configuration\":{\"EXECUTION_TIMEOUT\":120},\"tests\":[{\"name\":\"MQTT_Connect\"}, \
    \"configuration\":{\"test\":{\"id\":\"MQTT_Connect\"},\"testCase\":null,\"version\": \"0.0.0\"}]}]}\" \
    \"devicePermissionRoleArn\": \"arn:aws:iam::123456789012:role/Myrole\"}'
```

**Output:**

```
{
  "suiteDefinitionId": "0jtsigio7yenu",
  "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/0jtsigio7yenu",
  "suiteDefinitionName": "TestSuiteName",
  "createdAt": "2022-12-02T11:38:13.263000-05:00"
}
```

Untuk informasi selengkapnya, lihat [Membuat definisi rangkaian pengujian](#) di Panduan Pengembang Inti AWS IoT.

**Contoh 2: Untuk membuat rangkaian tes Kualifikasi Terbaru IoT Device Advisor**

`create-suite-definition` Contoh berikut membuat rangkaian pengujian kualifikasi penasihat perangkat dengan versi terbaru di AWS IoT dengan konfigurasi definisi suite yang ditentukan.

```
aws iotdeviceadvisor create-suite-definition \
  --suite-definition-configuration '{ \
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/MyIotThing"}], \
    "intendedForQualification": true, \
    "rootGroup": "", \
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole\"}'
```

**Output:**

```
{
```

```
"suiteDefinitionId": "txgsuolk2myj",
"suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-
east-1:123456789012:suitedefinition/txgsuolk2myj",
"suiteDefinitionName": "TestSuiteName",
"createdAt": "2022-12-02T11:38:13.263000-05:00"
}
```

Untuk informasi selengkapnya, lihat [Membuat definisi rangkaian pengujian](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [CreateSuiteDefinition](#) di Referensi AWS CLI Perintah.

## delete-suite-definition

Contoh kode berikut menunjukkan cara menggunakan `delete-suite-definition`.

### AWS CLI

Untuk menghapus rangkaian pengujian IoT Device Advisor

`delete-suite-definition` Contoh berikut menghapus rangkaian pengujian penasihat perangkat dengan ID definisi suite yang ditentukan.

```
aws iotdeviceadvisor delete-suite-definition \
  --suite-definition-id 0jtsgio7yenu
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteSuiteDefinition](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [DeleteSuiteDefinition](#) di Referensi AWS CLI Perintah.

## get-endpoint

Contoh kode berikut menunjukkan cara menggunakan `get-endpoint`.

### AWS CLI

Contoh 1: Untuk mendapatkan informasi tentang titik akhir tingkat Akun IoT Device Advisor

`get-endpoint` Contoh berikut mendapatkan informasi tentang titik akhir pengujian tingkat Akun penasihat perangkat.

```
aws iotdeviceadvisor get-endpoint
```

Output:

```
{
  "endpoint": "t6y4c143x9sfo.deviceadvisor.iot.us-east-1.amazonaws.com"
}
```

Contoh 2: Untuk mendapatkan informasi tentang titik akhir tingkat Perangkat IoT Device Advisor

`get-endpoint` Contoh berikut mendapatkan informasi tentang titik akhir pengujian tingkat perangkat penasihat perangkat dengan `thing-arn` atau `certificate-arn` yang ditentukan.

```
aws iotdeviceadvisor get-endpoint \
  --thing-arn arn:aws:iot:us-east-1:123456789012:thing/MyIotThing
```

Output:

```
{
  "endpoint": "tdb7719be5t6y4c143x9sfo.deviceadvisor.iot.us-east-1.amazonaws.com"
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan titik akhir pengujian](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [GetEndpoint](#) di Referensi AWS CLI Perintah.

## get-suite-definition

Contoh kode berikut menunjukkan cara menggunakan `get-suite-definition`.

### AWS CLI

Untuk mendapatkan informasi tentang rangkaian pengujian IoT Device Advisor

`get-suite-definition` Contoh berikut mendapatkan informasi tentang aevice advisor test suite dengan ID definisi suite yang ditentukan.

```
aws iotdeviceadvisor get-suite-definition \
```

```
--suite-definition-id qqcsmtyyjabl
```

### Output:

```
{
  "suiteDefinitionId": "qqcsmtyyjabl",
  "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/qqcsmtyyjabl",
  "suiteDefinitionVersion": "v1",
  "latestVersion": "v1",
  "suiteDefinitionConfiguration": {
    "suiteDefinitionName": "MQTT connection",
    "devices": [],
    "intendedForQualification": false,
    "isLongDurationTest": false,
    "rootGroup": "{\"configuration\":{},\"tests\":[{\"id\":\"uta5d9j1kvwc\",
    \"name\":\"Test group 1\",\"configuration\":{},\"tests\":[{\"id\":\"awr8pq5vc9yp\",
    \"name\":\"MQTT Connect\",\"configuration\":{},\"test\":{\"id\":\"MQTT_Connect\",
    \"testCase\":null,\"version\":\"0.0.0\"}]}]}]}",
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole",
    "protocol": "MqttV3_1_1"
  },
  "createdAt": "2022-11-11T22:28:52.389000-05:00",
  "lastModifiedAt": "2022-11-11T22:28:52.389000-05:00",
  "tags": {}
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan definisi rangkaian pengujian](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [GetSuiteDefinition](#) di Referensi AWS CLI Perintah.

## get-suite-run-report

Contoh kode berikut menunjukkan cara menggunakan `get-suite-run-report`.

### AWS CLI

Untuk mendapatkan informasi tentang laporan uji coba yang memenuhi syarat IoT Device Advisor `get-suite-run-report` Contoh berikut mendapatkan tautan unduhan laporan untuk rangkaian pengujian kualifikasi penasihat perangkat yang berhasil dijalankan dengan ID definisi suite dan ID run suite yang ditentukan.

```
aws iotdeviceadvisor get-suite-run-report \  
  --suite-definition-id ztvb5aek4w4x \  
  --suite-run-id p6awv83nre6v
```

Output:

```
{  
  "qualificationReportDownloadUrl": "https://senate-apn-reports-us-east-1-  
prod.s3.amazonaws.com/report.downloadlink"  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan laporan kualifikasi untuk rangkaian pengujian kualifikasi yang berhasil dijalankan](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [GetSuiteRunReport](#) di Referensi AWS CLI Perintah.

## get-suite-run

Contoh kode berikut menunjukkan cara menggunakan `get-suite-run`.

### AWS CLI

Untuk mendapatkan informasi tentang status run suite pengujian IoT Device Advisor

`get-suite-run` Contoh berikut mendapatkan informasi tentang status run suite pengujian penasihat perangkat dengan ID definisi suite dan ID run suite yang ditentukan.

```
aws iotdeviceadvisor get-suite-run \  
  --suite-definition-id qqcsmtyyjabl \  
  --suite-run-id nzlfyhaa18oa
```

Output:

```
{  
  "suiteDefinitionId": "qqcsmtyyjabl",  
  "suiteDefinitionVersion": "v1",  
  "suiteRunId": "nzlfyhaa18oa",  
  "suiteRunArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suiterun/  
qqcsmtyyjabl/nzlfyhaa18oa",  
  "suiteRunConfiguration": {
```

```

    "primaryDevice": {
      "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyIotThing",
      "certificateArn": "arn:aws:iot:us-east-1:123456789012:cert/certFile"
    },
    "parallelRun": false
  },
  "testResult": {
    "groups": [
      {
        "groupId": "uta5d9j1kvwc",
        "groupName": "Test group 1",
        "tests": [
          {
            "testCaseRunId": "2ve2twrqyr0s",
            "testCaseDefinitionId": "awr8pq5vc9yp",
            "testCaseDefinitionName": "MQTT Connect",
            "status": "PASS",
            "startTime": "2022-11-12T00:01:53.693000-05:00",
            "endTime": "2022-11-12T00:02:15.443000-05:00",
            "logUrl": "https://console.aws.amazon.com/cloudwatch/home?region=us-east-1#logEventViewer:group=/aws/iot/deviceadvisor/qqcsmtyyjabl;stream=nzlfyhaa18oa_2ve2twrqyr0s",
            "warnings": "null",
            "failure": "null"
          }
        ]
      }
    ]
  },
  "startTime": "2022-11-12T00:01:52.673000-05:00",
  "endTime": "2022-11-12T00:02:16.496000-05:00",
  "status": "PASS",
  "tags": {}
}

```

Untuk informasi selengkapnya, lihat [Menjalankan rangkaian pengujian](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [GetSuiteRundi Referensi AWS CLI Perintah](#).

## list-suite-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-suite-definitions`.



## AWS CLI

Contoh 1: Untuk mencantumkan rangkaian pengujian IoT Device Advisor yang Anda buat

`list-suite-definitions` Contoh berikut mencantumkan hingga 25 rangkaian pengujian penasihat perangkat yang Anda buat di AWS IoT. Jika Anda memiliki lebih dari 25 rangkaian pengujian, "NextToken" akan ditampilkan di output. Anda dapat menggunakan "NextToken" ini untuk menunjukkan rangkaian pengujian lainnya yang Anda buat.

```
aws iotdeviceadvisor list-suite-definitions
```

Output:

```
{
  "suiteDefinitionInformationList": [
    {
      "suiteDefinitionId": "3hsn88h4p2g5",
      "suiteDefinitionName": "TestSuite1",
      "defaultDevices": [
        {
          "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"
        }
      ],
      "intendedForQualification": false,
      "isLongDurationTest": false,
      "protocol": "MqttV3_1_1",
      "createdAt": "2022-11-17T14:15:56.830000-05:00"
    },
    {
      .....
    }
  ],
  "nextToken": "nextTokenValue"
}
```

Contoh 2: Untuk mencantumkan rangkaian pengujian IoT Device Advisor yang Anda buat dengan pengaturan yang ditentukan

`list-suite-definitions` Contoh berikut mencantumkan rangkaian pengujian penasihat perangkat yang Anda buat di AWS IoT dengan nomor hasil maksimal yang ditentukan. Jika Anda memiliki lebih banyak rangkaian pengujian daripada angka maks, "NextToken" akan

ditampilkan di output. Jika Anda memiliki “NextToken”, Anda dapat menggunakan “NextToken” untuk menampilkan rangkaian pengujian yang Anda buat yang tidak ditampilkan sebelumnya.

```
aws iotdeviceadvisor list-suite-definitions \  
  --max-result 1 \  
  --next-token "nextTokenValue"
```

Output:

```
{  
  "suiteDefinitionInformationList": [  
    {  
      "suiteDefinitionId": "ztlv5aew4w4x",  
      "suiteDefinitionName": "TestSuite2",  
      "defaultDevices": [],  
      "intendedForQualification": true,  
      "isLongDurationTest": false,  
      "protocol": "MqttV3_1_1",  
      "createdAt": "2022-11-17T14:15:56.830000-05:00"  
    }  
  ],  
  "nextToken": "nextTokenValue"  
}
```

Untuk informasi selengkapnya, lihat [ListSuiteDefinitions](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListSuiteDefinitions](#) di Referensi AWS CLI Perintah.

## list-suite-runs

Contoh kode berikut menunjukkan cara menggunakan `list-suite-runs`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua informasi tentang rangkaian pengujian IoT Device Advisor yang ditentukan menjalankan status

`list-suite-runs` Contoh berikut mencantumkan semua informasi tentang rangkaian pengujian penasihat perangkat menjalankan status dengan ID definisi suite yang ditentukan. Jika Anda menjalankan lebih dari 25 rangkaian pengujian, “NextToken” akan ditampilkan di output. Anda dapat menggunakan “NextToken” ini untuk menunjukkan sisa rangkaian pengujian yang berjalan.

```
aws iotdeviceadvisor list-suite-runs \  
  --suite-definition-id ztvb5aew4w4x
```

Output:

```
{  
  "suiteRunsList": [  
    {  
      "suiteDefinitionId": "ztvb5aew4w4x",  
      "suiteDefinitionVersion": "v1",  
      "suiteDefinitionName": "TestSuite",  
      "suiteRunId": "p6awv89nre6v",  
      "createdAt": "2022-12-01T16:33:14.212000-05:00",  
      "startedAt": "2022-12-01T16:33:15.710000-05:00",  
      "endAt": "2022-12-01T16:42:03.323000-05:00",  
      "status": "PASS",  
      "passed": 6,  
      "failed": 0  
    }  
  ]  
}
```

Contoh 2: Untuk mencantumkan informasi tentang rangkaian pengujian IoT Device Advisor yang ditentukan menjalankan status dengan pengaturan yang ditentukan

`list-suite-runs` Contoh berikut mencantumkan informasi tentang rangkaian pengujian penasihat perangkat yang menjalankan status dengan ID definisi suite yang ditentukan dan nomor hasil maksimal yang ditentukan. Jika Anda memiliki lebih banyak rangkaian pengujian daripada angka maks, "NextToken" akan ditampilkan di output. Jika Anda memiliki "NextToken", Anda dapat menggunakan "NextToken" untuk menunjukkan rangkaian pengujian yang tidak ditampilkan sebelumnya.

```
aws iotdeviceadvisor list-suite-runs \  
  --suite-definition-id qqcsmtyyjaml \  
  --max-result 1 \  
  --next-token "nextTokenValue"
```

Output:

```
{  
  "suiteRunsList": [  
    {
```

```

    {
      "suiteDefinitionId": "qqcsmtyyjaml",
      "suiteDefinitionVersion": "v1",
      "suiteDefinitionName": "MQTT connection",
      "suiteRunId": "gz9vm2s6d2jy",
      "createdAt": "2022-12-01T20:10:27.079000-05:00",
      "startedAt": "2022-12-01T20:10:28.003000-05:00",
      "endAt": "2022-12-01T20:10:45.084000-05:00",
      "status": "STOPPED",
      "passed": 0,
      "failed": 0
    }
  ],
  "nextToken": "nextTokenValue"
}

```

Untuk informasi selengkapnya, lihat [ListSuiteRuns](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [ListSuiteRuns](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke sumber daya IoT Device Advisor

`list-tags-for-resource` Contoh berikut mencantumkan tag yang dilampirkan ke sumber penasihat perangkat. Sumber daya penasihat perangkat dapat berupa `Suitedefinition-Arn` atau `Suiterun-Arn`.

```

aws iotdeviceadvisor list-tags-for-resource \
  --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/
ba0uyjpg38ny

```

Output:

```

{
  "tags": {
    "TestTagKey": "TestTagValue"
  }
}

```

```
}
```

Untuk informasi selengkapnya, lihat [ListTagsForResource](#) di Referensi API AWS IoT dan [tipe Sumber Daya yang ditentukan oleh AWS IoT Core Device Advisor](#) dalam Referensi Otorisasi Layanan.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## start-suite-run

Contoh kode berikut menunjukkan cara menggunakan `start-suite-run`.

### AWS CLI

Untuk memulai rangkaian pengujian IoT Device Advisor

`start-suite-run` Contoh berikut mencantumkan widget yang tersedia di AWS akun Anda.

```
aws iotdeviceadvisor start-suite-run \
  --suite-definition-id qqcsmtyyjabl \
  --suite-definition-version v1 \
  --suite-run-configuration '{"primaryDevice":{"thingArn": "arn:aws:iot:us-
east-1:123456789012:thing/MyIotThing","certificateArn":"arn:aws:iot:us-
east-1:123456789012:cert/certFile"}}'
```

Output:

```
{
  "suiteRunId": "pwmucgw7lt9s",
  "suiteRunArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suiterun/
qqcsmtyyjabl/pwmucgw7lk9s",
  "createdAt": "2022-12-02T15:43:05.581000-05:00"
}
```

Untuk informasi selengkapnya, lihat [Memulai rangkaian pengujian yang dijalankan](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [StartSuiteRun](#) di Referensi AWS CLI Perintah.

## stop-suite-run

Contoh kode berikut menunjukkan cara menggunakan `stop-suite-run`.

## AWS CLI

Untuk menghentikan rangkaian pengujian IoT Device Advisor yang sedang berjalan

`stop-suite-run` Contoh berikut menghentikan rangkaian pengujian penasihat perangkat yang saat ini berjalan dengan ID definisi suite dan ID run suite yang ditentukan.

```
aws iotdeviceadvisor stop-suite-run \  
  --suite-definition-id qqcsmtyyjabl \  
  --suite-run-id nzlfyhaa18oa
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghentikan rangkaian pengujian yang dijalankan](#) di Panduan Pengembang Inti AWS IoT.

- Untuk detail API, lihat [StopSuiteRundi Referensi AWS CLI Perintah](#).

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menambah dan memodifikasi tag yang ada dari sumber daya IoT Device Advisor

`tag-resource` Contoh berikut menambah dan memodifikasi tag yang ada dari sumber daya penasihat perangkat dengan arn dan tag sumber daya yang ditentukan. Sumber daya penasihat perangkat dapat berupa `SuiteDefinition-Arn` atau `SuiteRun-Arn`.

```
aws iotdeviceadvisor tag-resource \  
  --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/  
ba0uyjpg38ny \  
  --tags '{"TagKey": "TagValue"}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [TagResource](#) di Referensi API AWS IoT dan [tipe Sumber Daya yang ditentukan oleh AWS IoT Core Device Advisor](#) dalam Referensi Otorisasi Layanan.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag yang ada dari sumber daya IoT Device Advisor

`untag-resource` Contoh berikut menghapus tag yang ada dari sumber penasihat perangkat dengan arn sumber daya dan kunci tag yang ditentukan. Sumber daya penasihat perangkat dapat berupa `SuiteDefinition-Arn` atau `SuiteRun-Arn`.

```
aws iotdeviceadvisor untag-resource \
  --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/
ba0uyjpg38ny \
  --tag-keys "TagKey"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UntagResource](#) di Referensi API AWS IoT dan [tipe Sumber Daya yang ditentukan oleh AWS IoT Core Device Advisor](#) dalam Referensi Otorisasi Layanan.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-suite-definition

Contoh kode berikut menunjukkan cara menggunakan `update-suite-definition`.

### AWS CLI

Contoh 1: Untuk memperbarui rangkaian pengujian IoT Device Advisor

`update-suite-definition` Contoh berikut memperbarui rangkaian pengujian penasihat perangkat di AWS IoT dengan ID definisi suite dan konfigurasi definisi suite yang ditentukan.

```
aws iotdeviceadvisor update-suite-definition \
  --suite-definition-id 3hsn88h4p2g5 \
  --suite-definition-configuration '{ \
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn": "arn:aws:iot:us-east-1:123456789012:thing/
MyIoTThing"}], \
```

```

      "intendedForQualification": false, \
      "rootGroup": "{\"configuration\":{},\"tests\":[{\"name\":\"MQTT Connect\",
\\\"configuration\\\":{\\\"EXECUTION_TIMEOUT\\\":120},\\\"tests\\\":[{\\\"name\\\":\\\"MQTT_Connect\\\",
\\\"configuration\\\":{}},\\\"test\\\":{\\\"id\\\":\\\"MQTT_Connect\\\",\\\"testCase\\\":null,\\\"version
\\\":\\\"0.0.0\\\"}]}]}]}", \
      "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"'

```

### Output:

```

{
  "suiteDefinitionId": "3hsn88h4p2g5",
  "suiteDefinitionName": "TestSuiteName",
  "suiteDefinitionVersion": "v3",
  "createdAt": "2022-11-17T14:15:56.830000-05:00",
  "lastUpdatedAt": "2022-12-02T16:02:45.857000-05:00"
}

```

### Contoh 2: Untuk memperbarui rangkaian tes Kualifikasi Penasihat Perangkat IoT

update-suite-definitionContoh berikut memperbarui rangkaian pengujian kualifikasi penasihat perangkat di AWS IoT dengan ID definisi suite dan konfigurasi definisi suite yang ditentukan.

```

aws iotdeviceadvisor update-suite-definition \
  --suite-definition-id txgsuolk2myj \
  --suite-definition-configuration '{
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/
MyIoTThing"}], \
    "intendedForQualification": true, \
    "rootGroup": "", \
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"}'

```

### Output:

```

{
  "suiteDefinitionId": "txgsuolk2myj",
  "suiteDefinitionName": "TestSuiteName",
  "suiteDefinitionVersion": "v3",
  "createdAt": "2022-11-17T14:15:56.830000-05:00",
  "lastUpdatedAt": "2022-12-02T16:02:45.857000-05:00"
}

```



```
}
```

Untuk informasi selengkapnya, lihat [UpdateSuiteDefinition](#) di Referensi API AWS IoT.

- Untuk detail API, lihat [UpdateSuiteDefinition](#) di Referensi AWS CLI Perintah.

## AWS IoT data contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT data.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-thing-shadow**

Contoh kode berikut menunjukkan cara menggunakan `delete-thing-shadow`.

AWS CLI

Untuk menghapus dokumen bayangan perangkat

`delete-thing-shadow` Contoh berikut menghapus seluruh dokumen bayangan untuk perangkat bernama `MyRPi`.

```
aws iot-data delete-thing-shadow \  
  --thing-name MyRPi \  
  "output.txt"
```

Perintah tidak menghasilkan output pada layar, tetapi `output.txt` berisi informasi yang mengonfirmasi versi dan stempel waktu dokumen bayangan yang Anda hapus.

```
{"version":2,"timestamp":1560270384}
```

Untuk informasi selengkapnya, lihat [Menggunakan Bayangan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteThingShadow](#) di Referensi AWS CLI Perintah.

## get-thing-shadow

Contoh kode berikut menunjukkan cara menggunakan `get-thing-shadow`.

### AWS CLI

Untuk mendapatkan dokumen bayangan sesuatu

`get-thing-shadow` Contoh berikut mendapatkan dokumen bayangan benda untuk hal IoT yang ditentukan.

```
aws iot-data get-thing-shadow \  
  --thing-name MyRPi \  
  output.txt
```

Perintah tidak menghasilkan output pada layar, tetapi berikut ini menunjukkan isi `output.txt`:

```
{  
  "state":{  
    "reported":{  
      "moisture":"low"  
    }  
  },  
  "metadata":{  
    "reported":{  
      "moisture":{  
        "timestamp":1560269319  
      }  
    }  
  },  
  "version":1,"timestamp":1560269405  
}
```

Untuk informasi selengkapnya, lihat [Alur Data Layanan Device Shadow](#) di AWS Panduan Pengembang IoT.

- Untuk detail API, lihat [GetThingShadow](#) di Referensi AWS CLI Perintah.

## update-thing-shadow

Contoh kode berikut menunjukkan cara menggunakan `update-thing-shadow`.

### AWS CLI

Untuk memperbarui bayangan sesuatu

`update-thing-shadow` Contoh berikut memodifikasi keadaan bayangan perangkat saat ini untuk hal yang ditentukan dan menyimpannya ke file `output.txt`.

```
aws iot-data update-thing-shadow \  
  --thing-name MyRPi \  
  --payload '{"state":{"reported":{"moisture":"okay"}}}' \  
  "output.txt"
```

Perintah tidak menghasilkan output pada layar, tetapi berikut ini menunjukkan isi `output.txt`:

```
{  
  "state": {  
    "reported": {  
      "moisture": "okay"  
    }  
  },  
  "metadata": {  
    "reported": {  
      "moisture": {  
        "timestamp": 1560270036  
      }  
    }  
  },  
  "version": 2,  
  "timestamp": 1560270036  
}
```

Untuk informasi selengkapnya, lihat [Alur Data Layanan Device Shadow](#) di AWS Panduan Pengembang IoT.

- Untuk detail API, lihat [UpdateThingShadow](#) di Referensi AWS CLI Perintah.

## AWS IoT Events contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Events.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-put-message**

Contoh kode berikut menunjukkan cara menggunakan `batch-put-message`.

AWS CLI

Untuk mengirim pesan (input) ke AWS IoT Events

`batch-put-message` Contoh berikut mengirimkan satu set pesan ke sistem AWS IoT Events. Setiap payload pesan diubah menjadi input yang Anda tentukan (`inputName`) dan dicerna ke detektor apa pun yang memantau input tersebut. Jika beberapa pesan dikirim, urutan pemrosesan pesan tidak dijamin. Untuk menjamin pemesanan, Anda harus mengirim pesan satu per satu dan menunggu respons yang berhasil.

```
aws iotevents-data batch-put-message \  
  --cli-input-json file://highPressureMessage.json
```

Isi dari `highPressureMessage.json`:

```
{
  "messages": [
    {
      "messageId": "00001",
      "inputName": "PressureInput",
      "payload": "{\"motorid\": \"Fulton-A32\", \"sensorData\": {\"pressure\": 80, \"temperature\": 39} }"
    }
  ]
}
```

Output:

```
{
  "BatchPutMessageErrorEntries": []
}
```

Untuk informasi selengkapnya, lihat [BatchPutMessage](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [BatchPutMessage](#) di Referensi AWS CLI Perintah.

## batch-update-detector

Contoh kode berikut menunjukkan cara menggunakan `batch-update-detector`.

### AWS CLI

Untuk memperbarui detektor (contoh)

`batch-update-detector` Contoh berikut memperbarui status, nilai variabel, dan pengaturan timer dari satu atau lebih detektor (instance) dari model detektor tertentu.

```
aws iotevents-data batch-update-detector \
  --cli-input-json file://budFulton-A32.json
```

Isi dari `budFulton-A32.json`:

```
{
  "detectors": [
    {
```

```

    "messageId": "00001",
    "detectorModelName": "motorDetectorModel",
    "keyValue": "Fulton-A32",
    "state": {
      "stateName": "Normal",
      "variables": [
        {
          "name": "pressureThresholdBreached",
          "value": "0"
        }
      ],
      "timers": [
      ]
    }
  ]
}

```

Output:

```

{
  "batchUpdateDetectorErrorEntries": []
}

```

Untuk informasi selengkapnya, lihat [BatchUpdateDetector](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [BatchUpdateDetector](#) di Referensi AWS CLI Perintah.

## create-detector-model

Contoh kode berikut menunjukkan cara menggunakan `create-detector-model`.

### AWS CLI

Untuk membuat model detektor

`create-detector-model` Contoh berikut membuat model detektor dengan konfigurasi yang ditentukan oleh file parameter.

```

aws iotevents create-detector-model \
  --cli-input-json file://motorDetectorModel.json

```

## Isi dari motorDetectorModel.json:

```

{
  "detectorModelName": "motorDetectorModel",
  "detectorModelDefinition": {
    "states": [
      {
        "stateName": "Normal",
        "onEnter": {
          "events": [
            {
              "eventName": "init",
              "condition": "true",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value": "0"
                  }
                }
              ]
            }
          ]
        },
        "onInput": {
          "transitionEvents": [
            {
              "eventName": "Overpressurized",
              "condition": "$input.PressureInput.sensorData.pressure
&gt; 70",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value":
"$variable.pressureThresholdBreached + 3"
                  }
                }
              ],
              "nextState": "Dangerous"
            }
          ]
        }
      ]
    }
  },

```

```

    {
      "stateName": "Dangerous",
      "onEnter": {
        "events": [
          {
            "eventName": "Pressure Threshold Breached",
            "condition": "$variable.pressureThresholdBreach
1",
            "actions": [
              {
                "sns": {
                  "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                }
              }
            ]
          }
        ],
      },
      "onInput": {
        "events": [
          {
            "eventName": "Overpressurized",
            "condition": "$input.PressureInput.sensorData.pressure
&gt; 70",
            "actions": [
              {
                "setVariable": {
                  "variableName": "pressureThresholdBreach",
                  "value": "3"
                }
              }
            ]
          }
        ],
      },
      {
        "eventName": "Pressure Okay",
        "condition": "$input.PressureInput.sensorData.pressure
&lt;= 70",
        "actions": [
          {
            "setVariable": {
              "variableName": "pressureThresholdBreach",
              "value":
"$variable.pressureThresholdBreach - 1"

```



```

    }
  }
]
},
"transitionEvents": [
  {
    "eventName": "BackToNormal",
    "condition": "$input.PressureInput.sensorData.pressure
&lt;= 70 &amp;&amp; $variable.pressureThresholdBreached &lt;= 1",
    "nextState": "Normal"
  }
],
"onExit": {
  "events": [
    {
      "eventName": "Normal Pressure Restored",
      "condition": "true",
      "actions": [
        {
          "sns": {
            "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
          }
        }
      ]
    }
  ]
}
},
"initialStateName": "Normal"
},
"key": "motorid",
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

**Output:**

```

{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",

```

```
    "lastUpdateTime": 1560796816.077,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560796816.077,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
}
```

Untuk informasi selengkapnya, lihat [CreateDetectorModel](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [CreateDetectorModel](#) di Referensi AWS CLI Perintah.

## create-input

Contoh kode berikut menunjukkan cara menggunakan `create-input`.

### AWS CLI

Untuk membuat masukan

`create-input` Contoh berikut menciptakan masukan.

```
aws iotevents create-input \
  --cli-input-json file://pressureInput.json
```

Isi dari `pressureInput.json`:

```
{
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor",
  "inputDefinition": {
    "attributes": [
      { "jsonPath": "sensorData.pressure" },
      { "jsonPath": "motorid" }
    ]
  }
}
```

Output:

```
{
  "inputConfiguration": {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1560795312.542,
    "creationTime": 1560795312.542,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Untuk informasi selengkapnya, lihat [CreateInput](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [CreateInput](#) di Referensi AWS CLI Perintah.

## **delete-detector-model**

Contoh kode berikut menunjukkan cara menggunakan `delete-detector-model`.

### AWS CLI

Untuk menghapus model detektor

`delete-detector-model` Contoh berikut menghapus model detektor yang ditentukan. Setiap contoh aktif dari model detektor juga dihapus.

```
aws iotevents delete-detector-model \
  --detector-model-name motorDetectorModel
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteDetectorModel](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [DeleteDetectorModel](#) di Referensi AWS CLI Perintah.

## **delete-input**

Contoh kode berikut menunjukkan cara menggunakan `delete-input`.

### AWS CLI

Untuk menghapus input

`delete-input` Contoh berikut menghapus input yang ditentukan.

```
aws iotevents delete-input \  
  --input-name PressureInput
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeletelInput](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [DeletelInput](#) di Referensi AWS CLI Perintah.

## **describe-detector-model**

Contoh kode berikut menunjukkan cara menggunakan `describe-detector-model`.

### AWS CLI

Untuk mendapatkan informasi tentang model detektor

`describe-detector-model` Contoh berikut menampilkan rincian untuk model detektor yang ditentukan. Karena `version` parameter tidak ditentukan, informasi tentang versi terbaru dikembalikan.

```
aws iotevents describe-detector-model \  
  --detector-model-name motorDetectorModel
```

Output:

```
{  
  "detectorModel": {  
    "detectorModelConfiguration": {  
      "status": "ACTIVE",  
      "lastUpdateTime": 1560796816.077,  
      "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",  
      "creationTime": 1560796816.077,  
      "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/motorDetectorModel",  
      "key": "motorid",  
      "detectorModelName": "motorDetectorModel",  
      "detectorModelVersion": "1"  
    },  
    "detectorModelDefinition": {  
      "states": [  

```

```

    {
      "onInput": {
        "transitionEvents": [
          {
            "eventName": "Overpressurized",
            "actions": [
              {
                "setVariable": {
                  "variableName":
"pressureThresholdBreached",
                  "value":
"$variable.pressureThresholdBreached + 3"
                }
              },
            ],
            "condition":
"$input.PressureInput.sensorData.pressure > 70",
            "nextState": "Dangerous"
          }
        ],
        "events": []
      },
      "stateName": "Normal",
      "onEnter": {
        "events": [
          {
            "eventName": "init",
            "actions": [
              {
                "setVariable": {
                  "variableName":
"pressureThresholdBreached",
                  "value": "0"
                }
              }
            ],
            "condition": "true"
          }
        ]
      },
      "onExit": {
        "events": []
      }
    },
  ],
}

```

```

    {
      "onInput": {
        "transitionEvents": [
          {
            "eventName": "BackToNormal",
            "actions": [],
            "condition":
"$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreached <= 1",
            "nextState": "Normal"
          }
        ],
        "events": [
          {
            "eventName": "Overpressurized",
            "actions": [
              {
                "setVariable": {
                  "variableName":
"pressureThresholdBreached",
                  "value": "3"
                }
              }
            ],
            "condition":
"$input.PressureInput.sensorData.pressure > 70"
          },
          {
            "eventName": "Pressure Okay",
            "actions": [
              {
                "setVariable": {
                  "variableName":
"pressureThresholdBreached",
                  "value":
"$variable.pressureThresholdBreached - 1"
                }
              }
            ],
            "condition":
"$input.PressureInput.sensorData.pressure <= 70"
          }
        ]
      },

```

```

    "stateName": "Dangerous",
    "onEnter": {
      "events": [
        {
          "eventName": "Pressure Threshold Breached",
          "actions": [
            {
              "sns": {
                "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
              }
            }
          ],
          "condition": "$variable.pressureThresholdBreach >
1"
        }
      ],
      "onExit": {
        "events": [
          {
            "eventName": "Normal Pressure Restored",
            "actions": [
              {
                "sns": {
                  "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                }
              }
            ],
            "condition": "true"
          }
        ]
      }
    ],
    "initialStateName": "Normal"
  }
}

```

Untuk informasi selengkapnya, lihat [DescribeDetectorModel](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [DescribeDetectorModel](#) di Referensi AWS CLI Perintah.

## describe-detector

Contoh kode berikut menunjukkan cara menggunakan `describe-detector`.

### AWS CLI

Untuk mendapatkan informasi tentang detektor (instance).

`describe-detector` Contoh berikut menampilkan rincian untuk detektor tertentu (contoh).

```
aws iotevents-data describe-detector \  
  --detector-model-name motorDetectorModel \  
  --key-value "Fulton-A32"
```

### Output:

```
{  
  "detector": {  
    "lastUpdateTime": 1560797852.776,  
    "creationTime": 1560797852.775,  
    "state": {  
      "variables": [  
        {  
          "name": "pressureThresholdBreached",  
          "value": "3"  
        }  
      ],  
      "stateName": "Dangerous",  
      "timers": []  
    },  
    "keyValue": "Fulton-A32",  
    "detectorModelName": "motorDetectorModel",  
    "detectorModelVersion": "1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [DescribeDetector](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [DescribeDetector](#) di Referensi AWS CLI Perintah.

## describe-input

Contoh kode berikut menunjukkan cara menggunakan `describe-input`.



## AWS CLI

Untuk mendapatkan informasi tentang masukan

`describe-input` Contoh berikut menampilkan rincian untuk input yang ditentukan.

```
aws iotevents describe-input \  
  --input-name PressureInput
```

Output:

```
{  
  "input": {  
    "inputConfiguration": {  
      "status": "ACTIVE",  
      "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/  
PressureInput",  
      "lastUpdateTime": 1560795312.542,  
      "creationTime": 1560795312.542,  
      "inputName": "PressureInput",  
      "inputDescription": "Pressure readings from a motor"  
    },  
    "inputDefinition": {  
      "attributes": [  
        {  
          "jsonPath": "sensorData.pressure"  
        },  
        {  
          "jsonPath": "motorid"  
        }  
      ]  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [DescribeInput](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [DescribeInput](#) di Referensi AWS CLI Perintah.

### **describe-logging-options**

Contoh kode berikut menunjukkan cara menggunakan `describe-logging-options`.

## AWS CLI

Untuk mendapatkan informasi tentang pengaturan logging

`describe-logging-options` Contoh berikut mengambil pengaturan saat ini dari opsi logging Peristiwa AWS IoT.

```
aws iotevents describe-logging-options
```

Output:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "enabled": false,
    "level": "ERROR"
  }
}
```

Untuk informasi selengkapnya, lihat [DescribeLoggingOptions](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [DescribeLoggingOptions](#) di Referensi AWS CLI Perintah.

## **list-detector-model-versions**

Contoh kode berikut menunjukkan cara menggunakan `list-detector-model-versions`.

## AWS CLI

Untuk mendapatkan informasi tentang versi model detektor

`list-detector-model-versions` Contoh berikut Daftar semua versi model detektor. Hanya metadata yang terkait dengan setiap versi model detektor yang dikembalikan.

```
aws iotevents list-detector-model-versions \
  --detector-model-name motorDetectorModel
```

Output:

```
{
```

```
"detectorModelVersionSummaries": [
  {
    "status": "ACTIVE",
    "lastUpdateTime": 1560796816.077,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560796816.077,
    "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
]
```

Untuk informasi selengkapnya, lihat [ListDetectorModelVersions](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [ListDetectorModelVersions](#) di Referensi AWS CLI Perintah.

## list-detector-models

Contoh kode berikut menunjukkan cara menggunakan `list-detector-models`.

### AWS CLI

Untuk mendapatkan daftar model detektor Anda

`list-detector-models` Contoh berikut Daftar model detektor yang telah Anda buat. Hanya metadata yang terkait dengan setiap model detektor yang dikembalikan.

```
aws iotevents list-detector-models
```

Output:

```
{
  "detectorModelSummaries": [
    {
      "detectorModelName": "motorDetectorModel",
      "creationTime": 1552072424.212
      "detectorModelDescription": "Detect overpressure in a motor."
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ListDetectorModels](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [ListDetectorModels](#) di Referensi AWS CLI Perintah.

## list-detectors

Contoh kode berikut menunjukkan cara menggunakan `list-detectors`.

### AWS CLI

Untuk mendapatkan daftar detektor untuk model detektor

`list-detectors` Contoh berikut mencantumkan detektor (contoh model detektor) di akun Anda.

```
aws iotevents-data list-detectors \  
  --detector-model-name motorDetectorModel
```

Output:

```
{  
  "detectorSummaries": [  
    {  
      "lastUpdateTime": 1558129925.2,  
      "creationTime": 1552073155.527,  
      "state": {  
        "stateName": "Normal"  
      },  
      "keyValue": "Fulton-A32",  
      "detectorModelName": "motorDetectorModel",  
      "detectorModelVersion": "1"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListDetectors](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [ListDetectors](#) di Referensi AWS CLI Perintah.

## list-inputs

Contoh kode berikut menunjukkan cara menggunakan `list-inputs`.

## AWS CLI

Untuk daftar input

`list-inputs` Contoh berikut mencantumkan input yang telah Anda buat di akun Anda.

```
aws iotevents list-inputs
```

Perintah ini tidak menghasilkan output. Output:

```
{
  {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1551742986.768,
    "creationTime": 1551742986.768,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Untuk informasi selengkapnya, lihat [ListInputs](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [ListInputs](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk mencantumkan tag yang ditetapkan ke sumber daya.

`list-tags-for-resource` Contoh berikut mencantumkan nama kunci tag dan nilai yang telah Anda tetapkan ke sumber daya.

```
aws iotevents list-tags-for-resource \
  --resource-arn "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput"
```

Output:

```
{
```

```
"tags": [  
  {  
    "value": "motor",  
    "key": "deviceType"  
  }  
]
```

Untuk informasi selengkapnya, lihat [ListTagsForResource](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-logging-options

Contoh kode berikut menunjukkan cara menggunakan `put-logging-options`.

### AWS CLI

Untuk mengatur opsi pencatatan

`put-logging-options` Contoh berikut menetapkan atau memperbarui opsi logging Peristiwa AWS IoT. Jika Anda memperbarui nilai `loggingOptions`` field, it can take up to one minute for the change to take effect. Also, if you change the policy attached to the role you specified in the ``roleArn` bidang apa pun (misalnya, untuk memperbaiki kebijakan yang tidak valid), diperlukan waktu hingga lima menit agar perubahan tersebut diterapkan.

```
aws iotevents put-logging-options \  
  --cli-input-json file://logging-options.json
```

Isi dari `logging-options.json`:

```
{  
  "loggingOptions": {  
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",  
    "level": "DEBUG",  
    "enabled": true,  
    "detectorDebugOptions": [  
      {  
        "detectorModelName": "motorDetectorModel",  
        "keyValue": "Fulton-A32"  
      }  
    ]  
  }  
}
```

```

    }
  ]
}

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [PutLoggingOptions](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [PutLoggingOptions](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber daya

`tag-resource` Contoh berikut menambahkan atau memodifikasi (jika kunci `deviceType` sudah ada) tag yang dilampirkan sumber daya yang ditentukan.

```

aws iotevents tag-resource \
  --cli-input-json file://pressureInput.tag.json

```

Isi dari `pressureInput.tag.json`:

```

{
  "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
  "tags": [
    {
      "key": "deviceType",
      "value": "motor"
    }
  ]
}

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [TagResource](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dengan nama kunci yang ditentukan dari sumber daya yang ditentukan.

```
aws iotevents untag-resource \  
  --resource-arn arn:aws:iotevents:us-west-2:123456789012:input/PressureInput \  
  --tagkeys deviceType
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UntagResource](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-detector-model

Contoh kode berikut menunjukkan cara menggunakan `update-detector-model`.

### AWS CLI

Untuk memperbarui model detektor

`update-detector-model` Contoh berikut memperbarui model detektor yang ditentukan. Detektor (instance) yang dihasilkan oleh versi sebelumnya dihapus dan kemudian dibuat ulang saat input baru tiba.

```
aws iotevents update-detector-model \  
  --cli-input-json file://motorDetectorModel.update.json
```

Isi dari `motorDetectorModel.update.json`:

```
{  
  "detectorModelName": "motorDetectorModel",  
  "detectorModelDefinition": {  
    "states": [  
      {
```



```

    "stateName": "Normal",
    "onEnter": {
      "events": [
        {
          "eventName": "init",
          "condition": "true",
          "actions": [
            {
              "setVariable": {
                "variableName": "pressureThresholdBreached",
                "value": "0"
              }
            }
          ]
        }
      ]
    },
    "onInput": {
      "transitionEvents": [
        {
          "eventName": "Overpressurized",
          "condition": "$input.PressureInput.sensorData.pressure >
70",
          "actions": [
            {
              "setVariable": {
                "variableName": "pressureThresholdBreached",
                "value":
"$variable.pressureThresholdBreached + 3"
              }
            }
          ],
          "nextState": "Dangerous"
        }
      ]
    }
  },
  {
    "stateName": "Dangerous",
    "onEnter": {
      "events": [
        {
          "eventName": "Pressure Threshold Breached",
          "condition": "$variable.pressureThresholdBreached > 1",

```

```

        "actions": [
            {
                "sns": {
                    "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                }
            }
        ]
    },
    "onInput": {
        "events": [
            {
                "eventName": "Overpressurized",
                "condition": "$input.PressureInput.sensorData.pressure >
70",
                "actions": [
                    {
                        "setVariable": {
                            "variableName": "pressureThresholdBreached",
                            "value": "3"
                        }
                    }
                ]
            },
            {
                "eventName": "Pressure Okay",
                "condition": "$input.PressureInput.sensorData.pressure
<= 70",
                "actions": [
                    {
                        "setVariable": {
                            "variableName": "pressureThresholdBreached",
                            "value":
"$variable.pressureThresholdBreached - 1"
                        }
                    }
                ]
            }
        ],
        "transitionEvents": [
            {
                "eventName": "BackToNormal",

```

```

        "condition": "$input.PressureInput.sensorData.pressure
<= 70 && $variable.pressureThresholdBreached <= 1",
        "nextState": "Normal"
    }
]
},
"onExit": {
    "events": [
        {
            "eventName": "Normal Pressure Restored",
            "condition": "true",
            "actions": [
                {
                    "sns": {
                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                    }
                }
            ]
        }
    ]
}
]
}
},
"initialStateName": "Normal"
},
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

### Output:

```

{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",
    "lastUpdateTime": 1560799387.719,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560799387.719,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "2"
  }
}

```

```
}
```

Untuk informasi selengkapnya, lihat [UpdateDetectorModel](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [UpdateDetectorModel](#) di Referensi AWS CLI Perintah.

## update-input

Contoh kode berikut menunjukkan cara menggunakan `update-input`.

### AWS CLI

Untuk memperbarui input

`update-input` Contoh berikut memperbarui input yang ditentukan dengan deskripsi dan definisi baru.

```
aws iotevents update-input \  
  --cli-input-json file://pressureInput.json
```

Isi dari `pressureInput.json`:

```
{  
  "inputName": "PressureInput",  
  "inputDescription": "Pressure readings from a motor",  
  "inputDefinition": {  
    "attributes": [  
      { "jsonPath": "sensorData.pressure" },  
      { "jsonPath": "motorid" }  
    ]  
  }  
}
```

Output:

```
{  
  "inputConfiguration": {  
    "status": "ACTIVE",  
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",  
    "lastUpdateTime": 1560795976.458,  
    "creationTime": 1560795312.542,  
    "inputName": "PressureInput",
```

```
    "inputDescription": "Pressure readings from a motor"  
  }  
}
```

Untuk informasi selengkapnya, lihat [UpdateInput](#) di Referensi API Acara AWS IoT.

- Untuk detail API, lihat [UpdateInput](#) di Referensi AWS CLI Perintah.

## AWS IoT Events-Data contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Events-Data.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **batch-put-message**

Contoh kode berikut menunjukkan cara menggunakan `batch-put-message`.

### AWS CLI

Untuk mengirim pesan (input) ke AWS IoT Events

`batch-put-message` Contoh berikut mengirimkan satu set pesan ke sistem AWS IoT Events. Setiap payload pesan diubah menjadi input yang Anda tentukan (`inputName`) dan dicerna ke detektor apa pun yang memantau input tersebut. Jika beberapa pesan dikirim, urutan pemrosesan pesan tidak dijamin. Untuk menjamin pemesanan, Anda harus mengirim pesan satu per satu dan menunggu respons yang berhasil.

```
aws iotevents-data batch-put-message \  
  --cli-binary-format raw-in-base64-out \  
  --cli-input-json file://highPressureMessage.json
```

Isi dari `highPressureMessage.json`:

```
{  
  "messages": [  
    {  
      "messageId": "00001",  
      "inputName": "PressureInput",  
      "payload": "{\"motorid\": \"Fulton-A32\", \"sensorData\": {\"pressure\":  
80, \"temperature\": 39} }"  
    }  
  ]  
}
```

Output:

```
{  
  "BatchPutMessageErrorEntries": []  
}
```

Untuk informasi selengkapnya, lihat [BatchPutMessage](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [BatchPutMessage](#) di Referensi AWS CLI Perintah.

## batch-update-detector

Contoh kode berikut menunjukkan cara menggunakan `batch-update-detector`.

### AWS CLI

Untuk memperbarui detektor (contoh)

`batch-update-detector` Contoh berikut memperbarui status, nilai variabel, dan pengaturan timer dari satu atau lebih detektor (instance) dari model detektor tertentu.

```
aws iotevents-data batch-update-detector \  
  --cli-input-json file://budFulton-A32.json
```

Isi dari budFulton-A32.json:

```
{
  "detectors": [
    {
      "messageId": "00001",
      "detectorModelName": "motorDetectorModel",
      "keyValue": "Fulton-A32",
      "state": {
        "stateName": "Normal",
        "variables": [
          {
            "name": "pressureThresholdBreached",
            "value": "0"
          }
        ],
        "timers": [
        ]
      }
    }
  ]
}
```

Output:

```
{
  "batchUpdateDetectorErrorEntries": []
}
```

Untuk informasi selengkapnya, lihat [BatchUpdateDetector](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [BatchUpdateDetector](#) di Referensi AWS CLI Perintah.

## **create-detector-model**

Contoh kode berikut menunjukkan cara menggunakan `create-detector-model`.

### AWS CLI

Untuk membuat model detektor

`create-detector-model` Contoh berikut membuat model detektor.

```
aws iotevents create-detector-model \  
  --cli-input-json file://motorDetectorModel.json
```

Isi dari `motorDetectorModel.json`:

```
{  
  "detectorModelName": "motorDetectorModel",  
  "detectorModelDefinition": {  
    "states": [  
      {  
        "stateName": "Normal",  
        "onEnter": {  
          "events": [  
            {  
              "eventName": "init",  
              "condition": "true",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreach",  
                    "value": "0"  
                  }  
                }  
              ]  
            }  
          ]  
        },  
        "onInput": {  
          "transitionEvents": [  
            {  
              "eventName": "Overpressurized",  
              "condition": "$input.PressureInput.sensorData.pressure  
> 70",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreach",  
                    "value":  
"$variable.pressureThresholdBreach + 3"  
                  }  
                }  
              ],  
              "nextState": "Dangerous"  
            }  
          ]  
        }  
      ]  
    }  
  }  
}
```



```

    }
  ]
},
{
  "stateName": "Dangerous",
  "onEnter": {
    "events": [
      {
        "eventName": "Pressure Threshold Breached",
        "condition": "$variable.pressureThresholdBreached >
1",
        "actions": [
          {
            "sns": {
              "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
            }
          }
        ]
      }
    ]
  },
  "onInput": {
    "events": [
      {
        "eventName": "Overpressurized",
        "condition": "$input.PressureInput.sensorData.pressure
> 70",
        "actions": [
          {
            "setVariable": {
              "variableName": "pressureThresholdBreached",
              "value": "3"
            }
          }
        ]
      }
    ]
  },
  {
    "eventName": "Pressure Okay",
    "condition": "$input.PressureInput.sensorData.pressure
<= 70",
    "actions": [
      {

```

```

        "setVariable": {
            "variableName": "pressureThresholdBreached",
            "value":
"$variable.pressureThresholdBreached - 1"
        }
    ],
    "transitionEvents": [
        {
            "eventName": "BackToNormal",
            "condition": "$input.PressureInput.sensorData.pressure
<= 70 && $variable.pressureThresholdBreached <= 1",
            "nextState": "Normal"
        }
    ],
    "onExit": {
        "events": [
            {
                "eventName": "Normal Pressure Restored",
                "condition": "true",
                "actions": [
                    {
                        "sns": {
                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                        }
                    }
                ]
            }
        ]
    }
},
    "initialStateName": "Normal"
},
    "key": "motorid",
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

Output:

```
{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",
    "lastUpdateTime": 1560796816.077,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560796816.077,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
}
```

Untuk informasi selengkapnya, lihat [CreateDetectorModel](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [CreateDetectorModel](#) di Referensi AWS CLI Perintah.

## create-input

Contoh kode berikut menunjukkan cara menggunakan `create-input`.

### AWS CLI

Untuk membuat masukan

`create-input` Contoh berikut menciptakan masukan.

```
aws iotevents create-input \
  --cli-input-json file://pressureInput.json
```

Isi dari `pressureInput.json`:

```
{
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor",
  "inputDefinition": {
    "attributes": [
      { "jsonPath": "sensorData.pressure" },
      { "jsonPath": "motorid" }
    ]
  }
}
```

```
    ]  
  }  
}
```

Output:

```
{  
  "inputConfiguration": {  
    "status": "ACTIVE",  
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",  
    "lastUpdateTime": 1560795312.542,  
    "creationTime": 1560795312.542,  
    "inputName": "PressureInput",  
    "inputDescription": "Pressure readings from a motor"  
  }  
}
```

Untuk informasi selengkapnya, lihat [CreateInput](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [CreateInput](#) di Referensi AWS CLI Perintah.

## **delete-detector-model**

Contoh kode berikut menunjukkan cara menggunakan `delete-detector-model`.

### AWS CLI

Untuk menghapus model detektor

`delete-detector-model` Contoh berikut menghapus model detektor. Setiap contoh aktif dari model detektor juga dihapus.

```
aws iotevents delete-detector-model \  
  --detector-model-name motorDetectorModel*
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteDetectorModel](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [DeleteDetectorModel](#) di Referensi AWS CLI Perintah.

## delete-input

Contoh kode berikut menunjukkan cara menggunakan `delete-input`.

### AWS CLI

Untuk menghapus masukan

`delete-input` Contoh berikut menghapus input.

```
aws iotevents delete-input \  
  --input-name PressureInput
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeletelInput](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [DeletelInput](#) di Referensi AWS CLI Perintah.

## describe-detector-model

Contoh kode berikut menunjukkan cara menggunakan `describe-detector-model`.

### AWS CLI

Untuk mendapatkan informasi tentang model detektor

`describe-detector-model` Contoh berikut menjelaskan model detektor. Jika `version` parameter tidak ditentukan, perintah mengembalikan informasi tentang versi terbaru.

```
aws iotevents describe-detector-model \  
  --detector-model-name motorDetectorModel
```

Output:

```
{  
  "detectorModel": {  
    "detectorModelConfiguration": {  
      "status": "ACTIVE",  
      "lastUpdateTime": 1560796816.077,  
      "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",  
      "creationTime": 1560796816.077,  
    }  
  }  
}
```

```

    "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  },
  "detectorModelDefinition": {
    "states": [
      {
        "onInput": {
          "transitionEvents": [
            {
              "eventName": "Overpressurized",
              "actions": [
                {
                  "setVariable": {
                    "variableName":
"pressureThresholdBreached",
                    "value":
"$variable.pressureThresholdBreached + 3"
                  }
                ],
              "condition":
"$input.PressureInput.sensorData.pressure > 70",
              "nextState": "Dangerous"
            }
          ],
          "events": []
        },
        "stateName": "Normal",
        "onEnter": {
          "events": [
            {
              "eventName": "init",
              "actions": [
                {
                  "setVariable": {
                    "variableName":
"pressureThresholdBreached",
                    "value": "0"
                  }
                ]
            }
          ],

```

```

        "condition": "true"
      }
    ]
  },
  "onExit": {
    "events": []
  }
},
{
  "onInput": {
    "transitionEvents": [
      {
        "eventName": "BackToNormal",
        "actions": [],
        "condition":
"$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreach <= 1",
        "nextState": "Normal"
      }
    ],
    "events": [
      {
        "eventName": "Overpressurized",
        "actions": [
          {
            "setVariable": {
              "variableName":
"pressureThresholdBreach",
              "value": "3"
            }
          }
        ],
        "condition":
"$input.PressureInput.sensorData.pressure > 70"
      },
      {
        "eventName": "Pressure Okay",
        "actions": [
          {
            "setVariable": {
              "variableName":
"pressureThresholdBreach",
              "value":
"$variable.pressureThresholdBreach - 1"

```

```

        }
      ],
      "condition":
"$input.PressureInput.sensorData.pressure <= 70"
    }
  ]
},
"stateName": "Dangerous",
"onEnter": {
  "events": [
    {
      "eventName": "Pressure Threshold Breached",
      "actions": [
        {
          "sns": {
            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
          }
        }
      ],
      "condition": "$variable.pressureThresholdBreached >
1"
    }
  ],
  "onExit": {
    "events": [
      {
        "eventName": "Normal Pressure Restored",
        "actions": [
          {
            "sns": {
              "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
            }
          }
        ],
        "condition": "true"
      }
    ]
  }
}
],

```



```
        "initialStateName": "Normal"
      }
    }
  }
```

Untuk informasi selengkapnya, lihat [DescribeDetectorModel](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [DescribeDetectorModel](#) di Referensi AWS CLI Perintah.

## describe-detector

Contoh kode berikut menunjukkan cara menggunakan `describe-detector`.

### AWS CLI

Untuk mendapatkan informasi tentang detektor (instance)

`describe-detector` Contoh berikut mengembalikan informasi tentang detektor tertentu (contoh).

```
aws iotevents-data describe-detector \
  --detector-model-name motorDetectorModel \
  --key-value "Fulton-A32"
```

Output:

```
{
  "detector": {
    "lastUpdateTime": 1560797852.776,
    "creationTime": 1560797852.775,
    "state": {
      "variables": [
        {
          "name": "pressureThresholdBreached",
          "value": "3"
        }
      ],
      "stateName": "Dangerous",
      "timers": []
    },
    "keyValue": "Fulton-A32",
```

```
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
}
```

Untuk informasi selengkapnya, lihat [DescribeDetector](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [DescribeDetector](#) di Referensi AWS CLI Perintah.

## describe-input

Contoh kode berikut menunjukkan cara menggunakan `describe-input`.

### AWS CLI

Untuk mendapatkan informasi tentang masukan

`describe-input` Contoh berikut mengambil rincian input.

```
aws iotevents describe-input \
  --input-name PressureInput
```

Output:

```
{
  "input": {
    "inputConfiguration": {
      "status": "ACTIVE",
      "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/
PressureInput",
      "lastUpdateTime": 1560795312.542,
      "creationTime": 1560795312.542,
      "inputName": "PressureInput",
      "inputDescription": "Pressure readings from a motor"
    },
    "inputDefinition": {
      "attributes": [
        {
          "jsonPath": "sensorData.pressure"
        },
        {
          "jsonPath": "motorid"
        }
      ]
    }
  }
}
```

```
}
  ]
}
}
```

Untuk informasi selengkapnya, lihat [DescribeInput](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [DescribeInput](#) di Referensi AWS CLI Perintah.

## describe-logging-options

Contoh kode berikut menunjukkan cara menggunakan `describe-logging-options`.

### AWS CLI

Untuk mendapatkan informasi tentang pengaturan logging

`describe-logging-options` Contoh berikut mengambil opsi logging Peristiwa AWS IoT saat ini.

```
aws iotevents describe-logging-options
```

Output:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "enabled": false,
    "level": "ERROR"
  }
}
```

Untuk informasi selengkapnya, lihat [DescribeLoggingOptions](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [DescribeLoggingOptions](#) di Referensi AWS CLI Perintah.

## list-detector-model-versions

Contoh kode berikut menunjukkan cara menggunakan `list-detector-model-versions`.

## AWS CLI

Untuk mendapatkan informasi tentang versi model detektor

`list-detector-model-versions` Contoh berikut mencantumkan semua versi model detektor. Hanya metadata yang terkait dengan setiap versi model detektor yang dikembalikan.

```
aws iotevents list-detector-model-versions \  
--detector-model-name motorDetectorModel
```

Output:

```
{  
  "detectorModelVersionSummaries": [  
    {  
      "status": "ACTIVE",  
      "lastUpdateTime": 1560796816.077,  
      "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",  
      "creationTime": 1560796816.077,  
      "detectorModelArn": "arn:aws:iotevents:us-  
west-2:123456789012:detectorModel/motorDetectorModel",  
      "detectorModelName": "motorDetectorModel",  
      "detectorModelVersion": "1"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListDetectorModelVersions](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [ListDetectorModelVersions](#) di Referensi AWS CLI Perintah.

## **list-detector-models**

Contoh kode berikut menunjukkan cara menggunakan `list-detector-models`.

### AWS CLI

Untuk mendapatkan daftar model detektor Anda

`list-detector-models` Contoh berikut mencantumkan model detektor yang telah Anda buat. Hanya metadata yang terkait dengan setiap model detektor yang dikembalikan.

```
aws iotevents list-detector-models
```

Output:

```
{
  "detectorModelSummaries": [
    {
      "detectorModelName": "motorDetectorModel",
      "creationTime": 1552072424.212
      "detectorModelDescription": "Detect overpressure in a motor."
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ListDetectorModels](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [ListDetectorModels](#) di Referensi AWS CLI Perintah.

## list-detectors

Contoh kode berikut menunjukkan cara menggunakan `list-detectors`.

### AWS CLI

Untuk mendapatkan daftar detektor untuk model detektor

`list-detectors` Contoh berikut mencantumkan detektor (contoh model detektor).

```
aws iotevents-data list-detectors \
  --detector-model-name motorDetectorModel
```

Output:

```
{
  "detectorSummaries": [
    {
      "lastUpdateTime": 1558129925.2,
      "creationTime": 1552073155.527,
      "state": {
        "stateName": "Normal"
      }
    }
  ]
}
```

```
    },
    "keyValue": "Fulton-A32",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
]
```

Untuk informasi selengkapnya, lihat [ListDetectors](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [ListDetectors](#) di Referensi AWS CLI Perintah.

## list-inputs

Contoh kode berikut menunjukkan cara menggunakan `list-inputs`.

### AWS CLI

Untuk membuat daftar input

`list-inputs` Contoh berikut mencantumkan input yang telah Anda buat.

```
aws iotevents list-inputs
```

Output:

```
{
  "status": "ACTIVE",
  "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
  "lastUpdateTime": 1551742986.768,
  "creationTime": 1551742986.768,
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor"
}
```

Untuk informasi selengkapnya, lihat [ListInputs](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [ListInputs](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk mencantumkan tag yang ditetapkan ke sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag (metadata) yang telah Anda tetapkan ke sumber daya.

```
aws iotevents list-tags-for-resource \  
  --resource-arn "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput"
```

Output:

```
{  
  "tags": [  
    {  
      "value": "motor",  
      "key": "deviceType"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListTagsForResource](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-logging-options

Contoh kode berikut menunjukkan cara menggunakan `put-logging-options`.

## AWS CLI

Untuk mengatur opsi pencatatan

`list-tags-for-resource` Contoh berikut menetapkan atau memperbarui opsi logging Peristiwa AWS IoT. Jika Anda memperbarui nilai `loggingOptions` bidang apa pun, dibutuhkan waktu hingga satu menit agar perubahan diterapkan. Selain itu, jika Anda mengubah kebijakan yang dilampirkan pada peran yang Anda tentukan di `roleArn` bidang (misalnya, untuk memperbaiki kebijakan yang tidak valid), perubahan tersebut akan memakan waktu hingga lima menit agar perubahan tersebut diterapkan.

```
aws iotevents put-logging-options \  
  --cli-input-json file://logging-options.json
```

Isi dari `logging-options.json`:

```
{  
  "loggingOptions": {  
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",  
    "level": "DEBUG",  
    "enabled": true,  
    "detectorDebugOptions": [  
      {  
        "detectorModelName": "motorDetectorModel",  
        "keyValue": "Fulton-A32"  
      }  
    ]  
  }  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [PutLoggingOptions](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [PutLoggingOptions](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber daya

`tag-resource` Contoh berikut menambah atau memodifikasi tag dari sumber daya yang diberikan. Tag adalah metadata yang dapat digunakan untuk mengelola sumber daya.

```
aws iotevents tag-resource \  
  --cli-input-json file://pressureInput.tag.json
```

Isi dari `pressureInput.tag.json`:



```
{
  "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
  "tags": [
    {
      "key": "deviceType",
      "value": "motor"
    }
  ]
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [TagResource](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag yang ditentukan dari sumber daya.

```
aws iotevents untag-resource \
  --cli-input-json file://pressureInput.untag.json
```

Isi dari `pressureInput.untag.json`:

```
{
  "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
  "tagKeys": [
    "deviceType"
  ]
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [UntagResource](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-detector-model

Contoh kode berikut menunjukkan cara menggunakan `update-detector-model`.

### AWS CLI

Untuk memperbarui model detektor

`update-detector-model` Contoh berikut memperbarui model detektor. Detektor (instance) yang dihasilkan oleh versi sebelumnya dihapus dan kemudian dibuat ulang saat input baru tiba.

```
aws iotevents update-detector-model \  
  --cli-input-json file://motorDetectorModel.update.json
```

Isi dari `motorDetectorModel.update.json`:

```
{  
  "detectorModelName": "motorDetectorModel",  
  "detectorModelDefinition": {  
    "states": [  
      {  
        "stateName": "Normal",  
        "onEnter": {  
          "events": [  
            {  
              "eventName": "init",  
              "condition": "true",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreach",  
                    "value": "0"  
                  }  
                }  
              ]  
            }  
          ]  
        },  
        "onInput": {  
          "transitionEvents": [  
            {  
              "eventName": "Overpressurized",  
              "condition": "$input.PressureInput.sensorData.pressure > 70",  
            }  
          ]  
        }  
      ]  
    }  
  }  
}
```

```
        "actions": [
            {
                "setVariable": {
                    "variableName": "pressureThresholdBreach",
                    "value": "$variable.pressureThresholdBreach + 3"
                }
            }
        ],
        "nextState": "Dangerous"
    }
]
}
},
{
    "stateName": "Dangerous",
    "onEnter": {
        "events": [
            {
                "eventName": "Pressure Threshold Breach",
                "condition": "$variable.pressureThresholdBreach > 1",
                "actions": [
                    {
                        "sns": {
                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                        }
                    }
                ]
            }
        ]
    },
    "onInput": {
        "events": [
            {
                "eventName": "Overpressurized",
                "condition": "$input.PressureInput.sensorData.pressure > 70",
                "actions": [
                    {
                        "setVariable": {
                            "variableName": "pressureThresholdBreach",
                            "value": "3"
                        }
                    }
                ]
            }
        ]
    }
}
```

```
    },
    {
      "eventName": "Pressure Okay",
      "condition": "$input.PressureInput.sensorData.pressure <= 70",
      "actions": [
        {
          "setVariable": {
            "variableName": "pressureThresholdBreached",
            "value": "$variable.pressureThresholdBreached - 1"
          }
        }
      ]
    }
  ],
  "transitionEvents": [
    {
      "eventName": "BackToNormal",
      "condition": "$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreached <= 1",
      "nextState": "Normal"
    }
  ]
},
"onExit": {
  "events": [
    {
      "eventName": "Normal Pressure Restored",
      "condition": "true",
      "actions": [
        {
          "sns": {
            "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
          }
        }
      ]
    }
  ]
}
},
"initialStateName": "Normal"
},
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
```

```
}
```

Output:

```
{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",
    "lastUpdateTime": 1560799387.719,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560799387.719,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "2"
  }
}
```

Untuk informasi selengkapnya, lihat [UpdateDetectorModel](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [UpdateDetectorModel](#) di Referensi AWS CLI Perintah.

## update-input

Contoh kode berikut menunjukkan cara menggunakan `update-input`.

### AWS CLI

Untuk memperbarui masukan

`update-input` Contoh berikut memperbarui input.

```
aws iotevents update-input \
  --cli-input-json file://pressureInput.json
```

Isi dari `pressureInput.json`:

```
{
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor",
  "inputDefinition": {
```

```
    "attributes": [
      { "jsonPath": "sensorData.pressure" },
      { "jsonPath": "motorid" }
    ]
  }
}
```

Output:

```
{
  "inputConfiguration": {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1560795976.458,
    "creationTime": 1560795312.542,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Untuk informasi selengkapnya, lihat [UpdateInput](#) di Panduan Pengembang Acara AWS IoT\*.

- Untuk detail API, lihat [UpdateInput](#) di Referensi AWS CLI Perintah.

## AWS IoT Greengrass contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Greengrass.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

## Tindakan

### **associate-role-to-group**

Contoh kode berikut menunjukkan cara menggunakan `associate-role-to-group`.

#### AWS CLI

Untuk mengaitkan peran dengan kelompok Greengrass

`associate-role-to-group` Contoh berikut mengaitkan peran IAM yang ditentukan dengan grup Greengrass. Peran grup digunakan oleh fungsi dan konektor Lambda lokal untuk mengakses AWS layanan. Misalnya, peran grup Anda mungkin memberikan izin yang diperlukan untuk integrasi CloudWatch Log.

```
aws greengrass associate-role-to-group \  
  --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b \  
  --role-arn arn:aws:iam::123456789012:role/GG-Group-Role
```

Output:

```
{  
  "AssociatedAt": "2019-09-10T20:03:30Z"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Peran Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [AssociateRoleToGroup](#) di Referensi AWS CLI Perintah.

### **associate-service-role-to-account**

Contoh kode berikut menunjukkan cara menggunakan `associate-service-role-to-account`.

#### AWS CLI

Untuk mengaitkan peran layanan dengan AWS akun Anda

`associate-service-role-to-account` Contoh berikut mengaitkan peran layanan IAM, yang ditentukan oleh ARN-nya, dengan Greengrass AWS IoT di akun Anda. AWS Anda harus sebelumnya membuat peran layanan di IAM, dan Anda harus mengaitkan dokumen kebijakan dengannya yang memungkinkan AWS IoT Greengrass untuk mengambil peran ini.

```
aws greengrass associate-service-role-to-account \
  --role-arn "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
```

Output:

```
{
  "AssociatedAt": "2019-06-25T18:12:45Z"
}
```

Untuk informasi selengkapnya, lihat Peran [Layanan Greengrass di Panduan Pengembang Greengrass IoT AWS](#) .

- Untuk detail API, lihat [AssociateServiceRoleToAccount](#) di Referensi AWS CLI Perintah.

## create-connector-definition-version

Contoh kode berikut menunjukkan cara menggunakan `create-connector-definition-version`.

### AWS CLI

Untuk membuat versi definisi konektor

`create-connector-definition-version` Contoh berikut membuat versi definisi konektor dan mengaitkannya dengan definisi konektor yang ditentukan. Semua konektor dalam versi menentukan nilai untuk parameternya.

```
aws greengrass create-connector-definition-version \
  --connector-definition-id "55d0052b-0d7d-44d6-b56f-21867215e118" \
  --connectors "[{\"Id\": \"MyTwilioNotificationsConnector\",
  \"ConnectorArn\": \"arn:aws:greengrass:us-west-2:/:connectors/
  TwilioNotifications/versions/2\", \"Parameters\": {\"TWILIO_ACCOUNT_SID
  \": \"AC1a8d4204890840d7fc482aab38090d57\", \"TwilioAuthTokenSecretArn\":
  \"arn:aws:secretsmanager:us-west-2:123456789012:secret:greengrass-TwilioAuthToken-
  ntSlp6\", \"TwilioAuthTokenSecretArn-ResourceId\": \"TwilioAuthToken\",
  \"DefaultFromPhoneNumber\": \"4254492999\"}}]"
```

Output:

```
{
```



```

    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/55d0052b-0d7d-44d6-b56f-21867215e118/versions/33f709a0-c825-49cb-9eea-
dc8964fbd635",
    "CreationTimestamp": "2019-06-24T20:46:30.134Z",
    "Id": "55d0052b-0d7d-44d6-b56f-21867215e118",
    "Version": "33f709a0-c825-49cb-9eea-dc8964fbd635"
}

```

- Untuk detail API, lihat [CreateConnectorDefinitionVersion](#) di Referensi AWS CLI Perintah.

## create-connector-definition

Contoh kode berikut menunjukkan cara menggunakan `create-connector-definition`.

### AWS CLI

Untuk membuat definisi konektor

`create-connector-definition` Contoh contoh berikut membuat definisi konektor dan versi definisi konektor awal. Versi awal berisi satu konektor. Semua konektor dalam versi menentukan nilai untuk parameternya.

```

aws greengrass create-connector-definition \
  --name MySNSConnector \
  --initial-version "{\"Connectors\": [{\"Id\": \"MySNSConnector\", \"ConnectorArn
\": \"arn:aws:greengrass:us-west-2:/connectors/SNS/versions/1\", \"Parameters\":
{ \"DefaultSNSArn\": \"arn:aws:sns:us-west-2:123456789012:GGConnectorTopic\" } } ]}"

```

Output:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
  "CreationTimestamp": "2019-06-19T19:30:01.300Z",
  "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
  "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
  "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-
c7c2-4a26-a7e2-7bf478ea2623",
  "Name": "MySNSConnector"
}

```

Untuk informasi lebih lanjut, lihat [Memulai dengan Konektor Greengrass \(CLI\) AWS di Panduan Pengembang IoT](#).

- Untuk detail API, lihat [CreateConnectorDefinition](#) di Referensi AWS CLI Perintah.

## **create-core-definition-version**

Contoh kode berikut menunjukkan cara menggunakan `create-core-definition-version`.

### AWS CLI

Untuk membuat versi definisi inti

`create-core-definition-version` Contoh berikut membuat versi definisi inti dan mengaitkannya dengan definisi inti yang ditentukan. Versi ini hanya dapat berisi satu inti. Sebelum Anda dapat membuat inti, Anda harus terlebih dahulu membuat dan menyediakan hal AWS IoT yang sesuai. Proses ini mencakup `iot` perintah berikut, yang mengembalikan `ThingArn` dan `CertificateArn` diperlukan untuk `create-core-definition-version` perintah.

Buat hal AWS IoT yang sesuai dengan perangkat inti:

```
aws iot create-thing \  
  --thing-name "MyCoreDevice"
```

Output:

```
{  
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice",  
  "thingName": "MyCoreDevice",  
  "thingId": "cb419a19-9099-4515-9cec-e9b0e760608a"  
}
```

Buat kunci publik dan pribadi dan sertifikat perangkat inti untuk hal itu. Contoh ini menggunakan `create-keys-and-certificate` perintah dan memerlukan izin menulis ke direktori saat ini. Atau, Anda dapat menggunakan `create-certificate-from-csr` perintah.

```
aws iot create-keys-and-certificate \  
  --set-as-active \  
  --certificate-pem-outfile "myCore.cert.pem" \  
  --public-key-outfile "myCore.public.key" \  
  --private-key-outfile "myCore.private.key"
```

```
--private-key-outfile "myCore.private.key"
```

### Output:

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz",
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAkGgAwIBATgIUCGq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAAqKpRgnn6yq26U3yt5YFZquyukfRjBMXDcNOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
  "123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
}
```

Buat kebijakan AWS IoT yang memungkinkan iot dan greengrass bertindak. Untuk mempermudah, kebijakan berikut memungkinkan tindakan pada semua sumber daya, tetapi kebijakan Anda harus lebih ketat.

```
aws iot create-policy \
  --policy-name "Core_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\": \"Allow\", \"Action\":[\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect
\", \"iot:Receive\"], \"Resource\":[\"*\"]}, {\"Effect\": \"Allow\", \"Action\":
[\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot>DeleteThingShadow\"],
\"Resource\":[\"*\"]}, {\"Effect\": \"Allow\", \"Action\":[\"greengrass:*\"], \"Resource
\": [\"*\"]}]}"
```

### Output:

```
{
  "policyName": "Core_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/Core_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\": \"Allow\", \"Action\":[\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect
```

```

\", \"iot:Receive\"], \"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\":
[\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot>DeleteThingShadow\"],
\"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\": [\"greengrass:*\"], \"Resource
\": [\"*\"]}]}",
  "policyVersionId": "1"
}

```

Lampirkan kebijakan ke sertifikat:

```

aws iot attach-policy \
  --policy-name "Core_Devices" \
  --target "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"

```

Perintah ini tidak menghasilkan output.

Lampirkan benda itu ke sertifikat:

```

aws iot attach-thing-principal \
  --thing-name "MyCoreDevice" \
  --principal "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"

```

Perintah ini tidak menghasilkan output.

Buat versi definisi inti:

```

aws greengrass create-core-definition-version \
  --core-definition-id "582efe12-b05a-409e-9a24-a2ba1bcc4a12" \
  --cores "[{\"Id\": \"MyCoreDevice\", \"ThingArn\": \"arn:aws:iot:us-
west-2:123456789012:thing/MyCoreDevice\", \"CertificateArn\": \"arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz
\", \"SyncShadow\": true}]"

```

Output:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12/versions/3fdc1190-2ce5-44de-b98b-
eec8f9571014",
  "Version": "3fdc1190-2ce5-44de-b98b-eec8f9571014",
  "CreationTimestamp": "2019-09-18T00:15:09.838Z",

```

```
"Id": "582efe12-b05a-409e-9a24-a2ba1bcc4a12"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Inti AWS Greengrass IoT di Panduan Pengembang IoT Greengrass.AWS](#)

- Untuk detail API, lihat [CreateCoreDefinitionVersion](#) di Referensi AWS CLI Perintah.

## create-core-definition

Contoh kode berikut menunjukkan cara menggunakan `create-core-definition`.

### AWS CLI

Contoh 1: Untuk membuat definisi inti kosong

`create-core-definition` Contoh berikut membuat definisi inti Greengrass kosong (tidak ada versi awal). Sebelum inti dapat digunakan, Anda harus menggunakan `create-core-definition-version` perintah untuk menyediakan parameter lain untuk inti.

```
aws greengrass create-core-definition \
  --name cliGroup_Core
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/
b5c08008-54cb-44bd-9eec-c121b04283b5",
  "CreationTimestamp": "2019-06-25T18:23:22.106Z",
  "Id": "b5c08008-54cb-44bd-9eec-c121b04283b5",
  "LastUpdatedTimestamp": "2019-06-25T18:23:22.106Z",
  "Name": "cliGroup_Core"
}
```

Contoh 2: Untuk membuat definisi inti dengan versi awal

`create-core-definition` Contoh berikut membuat definisi inti yang berisi versi definisi inti awal. Versi ini hanya dapat berisi satu inti. Sebelum Anda dapat membuat inti, Anda harus terlebih dahulu membuat dan menyediakan hal AWS IoT yang sesuai. Proses ini mencakup `iot` perintah berikut, yang mengembalikan `ThingArn` dan `CertificateArn` diperlukan untuk `create-core-definition` perintah.

Buat hal AWS IoT yang sesuai dengan perangkat inti:

```
aws iot create-thing \  
  --thing-name "MyCoreDevice"
```

Output:

```
{  
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice",  
  "thingName": "MyCoreDevice",  
  "thingId": "cb419a19-9099-4515-9cec-e9b0e760608a"  
}
```

Buat kunci publik dan pribadi dan sertifikat perangkat inti untuk hal itu. Contoh ini menggunakan `create-keys-and-certificate` perintah dan memerlukan izin menulis ke direktori saat ini. Atau, Anda dapat menggunakan `create-certificate-from-csr` perintah.

```
aws iot create-keys-and-certificate \  
  --set-as-active \  
  --certificate-pem-outfile "myCore.cert.pem" \  
  --public-key-outfile "myCore.public.key" \  
  --private-key-outfile "myCore.private.key"
```

Output:

```
{  
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz",  
  "certificatePem": "-----BEGIN CERTIFICATE-----  
\nMIIDWTCAkGgAwIBATgIUCGq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END  
CERTIFICATE-----\n",  
  "keyPair": {  
    "PublicKey": "-----BEGIN PUBLIC KEY-----  
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEaKpRgnn6yq26U3y...wIDAQAB\n-----END  
PUBLIC KEY-----\n",  
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----  
\nMIIEowIABAKCAQEaKpRgnn6yq26U3yt5YFZquyukfRjBMXDcNOK4rMCxDR...fvY4+te\n-----END  
RSA PRIVATE KEY-----\n"  
  },  
  "certificateId":  
    "123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

```
}

```

Buat kebijakan AWS IoT yang memungkinkan iot dan greengrass bertindak. Untuk mempermudah, kebijakan berikut memungkinkan tindakan pada semua sumber daya, tetapi kebijakan Anda harus lebih ketat.

```
aws iot create-policy \
  --policy-name "Core_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\": [{\"Effect\": \"Allow\", \"Action\": [\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect\", \"iot:Receive\"], \"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\": [\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot:DeleteThingShadow\"], \"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\": [\"greengrass:*\"], \"Resource\": [\"*\"]} ]}"
```

Output:

```
{
  "policyName": "Core_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/Core_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\": [{\"Effect\": \"Allow\", \"Action\": [\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect\", \"iot:Receive\"], \"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\": [\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot:DeleteThingShadow\"], \"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\": [\"greengrass:*\"], \"Resource\": [\"*\"]} ]}",
  "policyVersionId": "1"
}
```

Lampirkan kebijakan ke sertifikat:

```
aws iot attach-policy \
  --policy-name "Core_Devices" \
  --target "arn:aws:iot:us-west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Perintah ini tidak menghasilkan output.

Lampirkan benda itu ke sertifikat:

```
aws iot attach-thing-principal \
  --thing-name "MyCoreDevice" \
```

```
--principal "arn:aws:iot:us-west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Perintah ini tidak menghasilkan output.

Buat definisi inti:

```
aws greengrass create-core-definition \
  --name "MyCores" \
  --initial-version "{ \"Cores\": [{ \"Id\": \"MyCoreDevice\", \"ThingArn\": \"arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice\", \"CertificateArn\": \"arn:aws:iot:us-west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz\", \"SyncShadow\": true } ] }"
```

Output:

```
{
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12/versions/cc87b5b3-8f4b-465d-944c-1d6de5dbfcdb",
  "Name": "MyCores",
  "LastUpdatedTimestamp": "2019-09-18T00:11:06.197Z",
  "LatestVersion": "cc87b5b3-8f4b-465d-944c-1d6de5dbfcdb",
  "CreationTimestamp": "2019-09-18T00:11:06.197Z",
  "Id": "582efe12-b05a-409e-9a24-a2ba1bcc4a12",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Inti AWS Greengrass IoT di Panduan Pengembang IoT Greengrass.AWS](#)

- Untuk detail API, lihat [CreateCoreDefinition](#) di Referensi AWS CLI Perintah.

## create-deployment

Contoh kode berikut menunjukkan cara menggunakan `create-deployment`.

### AWS CLI

Untuk membuat penerapan untuk versi grup Greengrass



`create-deployment` Contoh berikut menyebarkan versi tertentu dari grup Greengrass.

```
aws greengrass create-deployment \  
  --deployment-type NewDeployment \  
  --group-id "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca" \  
  --group-version-id "dc40c1e9-e8c8-4d28-a84d-a9cad5f599c9"
```

Output:

```
{  
  "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca/deployments/bfceb608-4e97-45bc-  
af5c-460144270308",  
  "DeploymentId": "bfceb608-4e97-45bc-af5c-460144270308"  
}
```

Untuk informasi selengkapnya, lihat [Memulai Konektor \(CLI\)](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [CreateDeployment](#) di Referensi AWS CLI Perintah.

## **create-device-definition-version**

Contoh kode berikut menunjukkan cara menggunakan `create-device-definition-version`.

### AWS CLI

Untuk membuat versi definisi perangkat

`create-device-definition-version` Contoh berikut membuat versi definisi perangkat dan mengaitkannya dengan definisi perangkat yang ditentukan. Versi ini mendefinisikan dua perangkat. Sebelum Anda dapat membuat perangkat Greengrass, Anda harus terlebih dahulu membuat dan menyediakan hal IoT yang sesuai. AWS Proses ini mencakup `iot` perintah berikut yang harus Anda jalankan untuk mendapatkan informasi yang diperlukan untuk perintah Greengrass:

Buat AWS IoT yang sesuai dengan perangkat:

```
aws iot create-thing \  
  --thing-name "InteriorTherm"
```

**Output:**

```
{
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",
  "thingName": "InteriorTherm",
  "thingId": "01d4763c-78a6-46c6-92be-7add080394bf"
}
```

Buat kunci publik dan pribadi dan sertifikat perangkat untuk benda itu. Contoh ini menggunakan `create-keys-and-certificate` perintah dan memerlukan izin menulis ke direktori saat ini. Atau, Anda dapat menggunakan `create-certificate-from-csr` perintah:

```
aws iot create-keys-and-certificate \
  --set-as-active \
  --certificate-pem-outfile "myDevice.cert.pem" \
  --public-key-outfile "myDevice.public.key" \
  --private-key-outfile "myDevice.private.key"
```

**Output:**

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCakGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjbmXDCnOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
  "66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
}
```

Buat kebijakan AWS IoT yang memungkinkan `iot` dan `greengrass` bertindak. Untuk mempermudah, kebijakan berikut memungkinkan tindakan pada semua sumber daya, tetapi kebijakan Anda bisa lebih ketat:

```
aws iot create-policy \
  --policy-name "GG_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"
  \"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
  \",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":"
  [\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
  \"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
  \":[\"*\"]}]}"
```

### Output:

```
{
  "policyName": "GG_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/GG_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"
  \"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
  \",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":"
  [\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
  \"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
  \":[\"*\"]}]}",
  "policyVersionId": "1"
}
```

### Lampirkan kebijakan ke sertifikat:

```
aws iot attach-policy \
  --policy-name "GG_Devices" \
  --target "arn:aws:iot:us-
  west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

### Lampirkan benda itu ke sertifikat

```
aws iot attach-thing-principal \
  --thing-name "InteriorTherm" \
  --principal "arn:aws:iot:us-
  west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Setelah Anda membuat dan mengkonfigurasi hal IoT seperti yang ditunjukkan di atas, gunakan ThingArn dan CertificateArn dari dua perintah pertama dalam contoh berikut.

```
aws greengrass create-device-definition-version \
```

```
--device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
--devices "[{"Id":"InteriorTherm","ThingArn":"arn:aws:iot:us-
west-2:123456789012:thing/InteriorTherm","CertificateArn":"arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92","
SyncShadow":true}, {"Id":"ExteriorTherm","ThingArn":"arn:aws:iot:us-
west-2:123456789012:thing/ExteriorTherm","CertificateArn":"arn:aws:iot:us-
west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02","
SyncShadow":true}]"]
```

### Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
  "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
  "CreationTimestamp": "2019-09-11T00:15:09.838Z",
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Untuk detail API, lihat [CreateDeviceDefinitionVersion](#) di Referensi AWS CLI Perintah.

## create-device-definition

Contoh kode berikut menunjukkan cara menggunakan `create-device-definition`.

### AWS CLI

Untuk membuat definisi perangkat

`create-device-definition` Contoh berikut membuat definisi perangkat yang berisi versi definisi perangkat awal. Versi awal mendefinisikan dua perangkat. Sebelum Anda dapat membuat perangkat Greengrass, Anda harus terlebih dahulu membuat dan menyediakan hal IoT yang sesuai. AWS Proses ini mencakup `iot` perintah berikut yang harus Anda jalankan untuk mendapatkan informasi yang diperlukan untuk perintah Greengrass:

Buat AWS IoT yang sesuai dengan perangkat:

```
aws iot create-thing \
  --thing-name "InteriorTherm"
```

**Output:**

```
{
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",
  "thingName": "InteriorTherm",
  "thingId": "01d4763c-78a6-46c6-92be-7add080394bf"
}
```

Buat kunci publik dan pribadi dan sertifikat perangkat untuk benda itu. Contoh ini menggunakan `create-keys-and-certificate` perintah dan memerlukan izin menulis ke direktori saat ini. Atau, Anda dapat menggunakan `create-certificate-from-csr` perintah:

```
aws iot create-keys-and-certificate \
  --set-as-active \
  --certificate-pem-outfile "myDevice.cert.pem" \
  --public-key-outfile "myDevice.public.key" \
  --private-key-outfile "myDevice.private.key"
```

**Output:**

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCakGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjbmXDcNOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
  "66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
}
```

Buat kebijakan AWS IoT yang memungkinkan `iot` dan `greengrass` bertindak. Untuk mempermudah, kebijakan berikut memungkinkan tindakan pada semua sumber daya, tetapi kebijakan Anda bisa lebih ketat:

```
aws iot create-policy \
  --policy-name "GG_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource\":[\"*\"]}]}"
```

### Output:

```
{
  "policyName": "GG_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/GG_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource\":[\"*\"]}]}",
  "policyVersionId": "1"
}
```

### Lampirkan kebijakan ke sertifikat:

```
aws iot attach-policy \
  --policy-name "GG_Devices" \
  --target "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

### Lampirkan benda itu ke sertifikat

```
aws iot attach-thing-principal \
  --thing-name "InteriorTherm" \
  --principal "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Setelah Anda membuat dan mengkonfigurasi hal IoT seperti yang ditunjukkan di atas, gunakan `ThingArn` dan `CertificateArn` dari dua perintah pertama dalam contoh berikut.

```
aws greengrass create-device-definition \
```

```
--name "Sensors" \
--initial-version "{\"Devices\":{\"Id\":\"InteriorTherm
\", \"ThingArn\":\"arn:aws:iot:us-west-2:123456789012:thing/
InteriorTherm\", \"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92\",
\"SyncShadow\":true}, {\"Id\":\"ExteriorTherm\", \"ThingArn\":\"arn:aws:iot:us-
west-2:123456789012:thing/ExteriorTherm\", \"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02\",
\"SyncShadow\":true}}}"
```

### Output:

```
{
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/3b5cc510-58c1-44b5-9d98-4ad858ffa795",
  "Name": "Sensors",
  "LastUpdatedTimestamp": "2019-09-11T00:11:06.197Z",
  "LatestVersion": "3b5cc510-58c1-44b5-9d98-4ad858ffa795",
  "CreationTimestamp": "2019-09-11T00:11:06.197Z",
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Untuk detail API, lihat [CreateDeviceDefinition](#) di Referensi AWS CLI Perintah.

## create-function-definition-version

Contoh kode berikut menunjukkan cara menggunakan `create-function-definition-version`.

### AWS CLI

Untuk membuat versi definisi fungsi

`create-function-definition-version` Contoh berikut membuat versi baru dari definisi fungsi yang ditentukan. Versi ini menentukan fungsi tunggal yang IDnya `Hello-World-function`, memungkinkan akses ke sistem file, dan menentukan ukuran memori maksimum dan periode batas waktu.

```
aws greengrass create-function-definition-version \
```

```
--cli-input-json "{\"FunctionDefinitionId\": \"e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3\", \"Functions\": [{\"Id\": \"Hello-World-function\", \"FunctionArn\": \"arn:aws:lambda:us-west-2:123456789012:function:Greengrass_HelloWorld_Counter:gghw-alias\", \"FunctionConfiguration\": {\"Environment\": {\"AccessSysfs\": true}, \"Executable\": \"greengrassHelloWorldCounter.function_handler\", \"MemorySize\": 16000, \"Pinned\": false, \"Timeout\": 25}}]}"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3/versions/74abd1cc-637e-4abe-8684-9a67890f4043",
  "CreationTimestamp": "2019-06-25T22:03:43.376Z",
  "Id": "e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3",
  "Version": "74abd1cc-637e-4abe-8684-9a67890f4043"
}
```

- Untuk detail API, lihat [CreateFunctionDefinitionVersion](#) di Referensi AWS CLI Perintah.

## create-function-definition

Contoh kode berikut menunjukkan cara menggunakan `create-function-definition`.

### AWS CLI

Untuk membuat definisi fungsi Lambda

`create-function-definition` Contoh berikut membuat definisi fungsi Lambda dan versi awal dengan menyediakan daftar fungsi Lambda (dalam hal ini, daftar hanya satu fungsi bernama `TempMonitorFunction`) dan konfigurasinya. Sebelum Anda dapat membuat definisi fungsi, Anda memerlukan fungsi Lambda ARN. Untuk membuat fungsi dan aliasnya, gunakan `create-function` dan `publish-version` perintah. `create-function` Perintah Lambda memerlukan ARN dari peran eksekusi, meskipun Greengrass AWS IoT tidak menggunakan peran itu karena izin ditentukan dalam peran grup Greengrass. Anda dapat menggunakan `create-role` perintah IAM untuk membuat peran kosong agar ARN dapat digunakan dengan Lambda atau Anda dapat menggunakan peran eksekusi yang `create-function` ada.

```
aws greengrass create-function-definition \
```



```
--name MyGreengrassFunctions \
--initial-version "{\"Functions\": [{\"Id\": \"TempMonitorFunction\",
\"FunctionArn\": \"arn:aws:lambda:us-
west-2:123456789012:function:TempMonitor:GG_TempMonitor\", \"FunctionConfiguration
\": {\"Executable\": \"temp_monitor.function_handler\", \"MemorySize\": 16000,
\"Timeout\": 5}}]}}"
```

**Output:**

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/3b0d0080-87e7-48c6-b182-503ec743a08b",
  "CreationTimestamp": "2019-06-19T22:24:44.585Z",
  "Id": "3b0d0080-87e7-48c6-b182-503ec743a08b",
  "LastUpdatedTimestamp": "2019-06-19T22:24:44.585Z",
  "LatestVersion": "67f918b9-efb4-40b0-b87c-de8c9faf085b",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/3b0d0080-87e7-48c6-b182-503ec743a08b/versions/67f918b9-
efb4-40b0-b87c-de8c9faf085b",
  "Name": "MyGreengrassFunctions"
}
```

Untuk informasi selengkapnya, lihat [Cara Mengkonfigurasi Akses Sumber Daya Lokal Menggunakan Antarmuka Baris AWS Perintah](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [CreateFunctionDefinition](#) di Referensi AWS CLI Perintah.

**create-group-certificate-authority**

Contoh kode berikut menunjukkan cara menggunakan `create-group-certificate-authority`.

**AWS CLI**

Untuk membuat otoritas sertifikat (CA) untuk grup

`create-group-certificate-authority` Contoh berikut membuat atau memutar CA untuk grup tertentu.

```
aws greengrass create-group-certificate-authority \
--group-id "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1"
```

**Output:**

```
{
  "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/8eaadd72-ce4b-4f15-892a-0cc4f3a343f1/certificateauthorities/
d31630d674c4437f6c5dbc0dca56312a902171ce2d086c38e509c8EXAMPLEecc5"
}
```

Untuk informasi selengkapnya, lihat [Keamanan AWS Greengrass IoT di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [CreateGroupCertificateAuthority](#) di Referensi AWS CLI Perintah.

## create-group-version

Contoh kode berikut menunjukkan cara menggunakan `create-group-version`.

### AWS CLI

Untuk membuat versi grup Greengrass

`create-group-version` Contoh berikut membuat versi grup dan mengaitkannya dengan grup yang ditentukan. Versi ini mereferensikan versi inti, sumber daya, konektor, fungsi, dan langganan yang berisi entitas untuk disertakan dalam versi grup ini. Anda harus membuat entitas ini sebelum Anda dapat membuat versi grup.

Untuk membuat definisi sumber daya dengan versi awal, gunakan `create-resource-definition` perintah. Untuk membuat definisi konektor dengan versi awal, gunakan `create-connector-definition` perintah. Untuk membuat definisi fungsi dengan versi awal, gunakan `create-function-definition` perintah. Untuk membuat definisi langganan dengan versi awal, gunakan perintah. Untuk `create-subscription-definition` mengambil ARN dari versi definisi inti terbaru, gunakan perintah dan tentukan ID versi grup terbaru. `get-group-version`

```
aws greengrass create-group-version \
  --group-id "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca" \
  --core-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/6a630442-8708-4838-ad36-eb98849d975e/
versions/6c87151b-1fb4-4cb2-8b31-6ee715d8f8ba" \
  --resource-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/
a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1" \
```

```
--connector-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/connectors/55d0052b-0d7d-44d6-b56f-21867215e118/versions/78a3331b-895d-489b-8823-17b4f9f418a0" \
--function-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/3b0d0080-87e7-48c6-b182-503ec743a08b/versions/67f918b9-efb4-40b0-b87c-de8c9faf085b" \
--subscription-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/subscriptions/9d611d57-5d5d-44bd-a3b4-fecbbdd69112/versions/aa645c47-ac90-420d-9091-8c7ffa4f103f"
```

### Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca/versions/e10b0459-4345-4a09-88a4-1af1f5d34638",
  "CreationTimestamp": "2019-06-20T18:42:47.020Z",
  "Id": "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca",
  "Version": "e10b0459-4345-4a09-88a4-1af1f5d34638"
}
```

Untuk informasi selengkapnya, lihat [AWS Ikhtisar Model Objek Grup Greengrass di Panduan Pengembang IoT Greengrass.AWS](#)

- Untuk detail API, lihat [CreateGroupVersion](#) di Referensi AWS CLI Perintah.

## create-group

Contoh kode berikut menunjukkan cara menggunakan `create-group`.

### AWS CLI

Untuk membuat grup Greengrass

`create-group` Contoh berikut membuat grup bernama `cli-created-group`.

```
aws greengrass create-group \
  --name cli-created-group
```

### Output:

```
{
```

```

    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/4e22bd92-898c-436b-ade5-434d883ff749",
    "CreationTimestamp": "2019-06-25T18:07:17.688Z",
    "Id": "4e22bd92-898c-436b-ade5-434d883ff749",
    "LastUpdatedTimestamp": "2019-06-25T18:07:17.688Z",
    "Name": "cli-created-group"
}

```

Untuk informasi selengkapnya, lihat [AWS Ikhtisar Model Objek Grup Greengrass di Panduan Pengembang IoT Greengrass.AWS](#)

- Untuk detail API, lihat [CreateGroup](#) di Referensi AWS CLI Perintah.

## create-logger-definition-version

Contoh kode berikut menunjukkan cara menggunakan `create-logger-definition-version`.

### AWS CLI

Untuk membuat versi definisi logger

`create-logger-definition-version` Contoh berikut membuat versi definisi logger dan mengaitkannya dengan definisi logger. Versi ini mendefinisikan empat konfigurasi logging: 1) log komponen sistem pada sistem file perangkat inti, 2) log fungsi Lambda yang ditentukan pengguna pada sistem file perangkat inti, 3) log komponen sistem di Amazon Logs, dan 4) log fungsi Lambda yang ditentukan pengguna di CloudWatch Amazon Logs. CloudWatch Catatan: Untuk integrasi CloudWatch Log, peran grup Anda harus memberikan izin yang sesuai.

```

aws greengrass create-logger-definition-version \
  --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0" \
  --loggers "[{"Id":"1","Component":"GreengrassSystem","Level":"ERROR",
  "Space":10240,"Type":"FileSystem"}, {"Id":"2","Component":"Lambda",
  "Level":"INFO","Space":10240,"Type":"FileSystem"}, {"Id":"3",
  "Component":"GreengrassSystem","Level":"WARN","Type":"AWSCloudWatch"},
  {"Id":"4","Component":"Lambda","Level":"INFO","Type":"AWSCloudWatch"}]"

```

Output:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/loggers/
a454b62a-5d56-4ca9-bdc4-8254e1662cb0/versions/49aedb1e-01a3-4d39-9871-3a052573f1ea",

```

```

"Version": "49aedb1e-01a3-4d39-9871-3a052573f1ea",
"CreationTimestamp": "2019-07-24T00:04:48.523Z",
"Id": "a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
}

```

Untuk informasi selengkapnya, lihat [Pemantauan dengan Log AWS Greengrass IoT di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [CreateLoggerDefinitionVersion](#) di Referensi AWS CLI Perintah.

## create-logger-definition

Contoh kode berikut menunjukkan cara menggunakan `create-logger-definition`.

### AWS CLI

Untuk membuat definisi logger

`create-logger-definition` Contoh berikut membuat definisi logger yang berisi versi definisi logger awal. Versi awal mendefinisikan tiga konfigurasi logging: 1) log komponen sistem pada sistem file perangkat inti, 2) log fungsi Lambda yang ditentukan pengguna pada sistem file perangkat inti, dan 3) log fungsi Lambda yang ditentukan pengguna di Amazon Logs. CloudWatch Catatan: Untuk integrasi CloudWatch Log, peran grup Anda harus memberikan izin yang sesuai.

```

aws greengrass create-logger-definition \
  --name "LoggingConfigs" \
  --initial-version "{ \"Loggers\": [ { \"Id\": \"1\", \"Component\": \"GreengrassSystem\", \"Level\": \"ERROR\", \"Space\": 10240, \"Type\": \"FileSystem\" }, { \"Id\": \"2\", \"Component\": \"Lambda\", \"Level\": \"INFO\", \"Space\": 10240, \"Type\": \"FileSystem\" }, { \"Id\": \"3\", \"Component\": \"Lambda\", \"Level\": \"INFO\", \"Type\": \"AWSCloudWatch\" } ] }"

```

Output:

```

{
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0/versions/de1d9854-1588-4525-b25e-b378f60f2322",
  "Name": "LoggingConfigs",
  "LastUpdatedTimestamp": "2019-07-23T23:52:17.165Z",
  "LatestVersion": "de1d9854-1588-4525-b25e-b378f60f2322",
}

```

```

    "CreationTimestamp": "2019-07-23T23:52:17.165Z",
    "Id": "a454b62a-5d56-4ca9-bdc4-8254e1662cb0",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
}

```

Untuk informasi selengkapnya, lihat [Pemantauan dengan Log AWS Greengrass IoT di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [CreateLoggerDefinition](#) di Referensi AWS CLI Perintah.

## create-resource-definition-version

Contoh kode berikut menunjukkan cara menggunakan `create-resource-definition-version`.

### AWS CLI

Untuk membuat versi definisi sumber daya

`create-resource-definition-version` Contoh berikut membuat versi baru dari file `TwilioAuthToken`.

```

aws greengrass create-resource-definition-version \
  --resource-definition-id "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38" \
  --resources "[{"Id": "TwilioAuthToken"}, {"Name": "MyTwilioAuthToken"}, {"ResourceDataContainer": {"SecretsManagerSecretResourceData": {"ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:greengrass-TwilioAuthToken-ntS1p6"}}}]"

```

Output:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/b3bcada0-5fb6-42df-bf0b-1ee4f15e769e",
  "CreationTimestamp": "2019-06-24T21:17:25.623Z",
  "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
  "Version": "b3bcada0-5fb6-42df-bf0b-1ee4f15e769e"
}

```

- Untuk detail API, lihat [CreateResourceDefinitionVersion](#) di Referensi AWS CLI Perintah.

## create-resource-definition

Contoh kode berikut menunjukkan cara menggunakan `create-resource-definition`.

### AWS CLI

Untuk membuat definisi sumber daya

`create-resource-definition` Contoh berikut membuat definisi sumber daya yang berisi daftar sumber daya yang akan digunakan dalam grup Greengrass. Dalam contoh ini, versi awal definisi sumber daya disertakan dengan menyediakan daftar sumber daya. Daftar ini mencakup satu sumber daya untuk token otorisasi Twilio dan ARN untuk rahasia yang disimpan di Secrets Manager. AWS Anda harus membuat rahasia sebelum Anda dapat membuat definisi sumber daya.

```
aws greengrass create-resource-definition \  
  --name MyGreengrassResources \  
  --initial-version "{\"Resources\": [{\"Id\": \"TwilioAuthToken\  
\", \"Name\": \"MyTwilioAuthToken\", \"ResourceDataContainer\":  
  {\"SecretsManagerSecretResourceData\": {\"ARN\": \"arn:aws:secretsmanager:us-  
west-2:123456789012:secret:greengrass-TwilioAuthToken-ntSlp6\"}}}]}"
```

Output:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",  
  "CreationTimestamp": "2019-06-19T21:51:28.212Z",  
  "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",  
  "LastUpdatedTimestamp": "2019-06-19T21:51:28.212Z",  
  "LatestVersion": "a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/a5f94d0b-  
f6bc-40f4-bb78-7a1c5fe13ba1",  
  "Name": "MyGreengrassResources"  
}
```

Untuk informasi selengkapnya, lihat [Cara Mengkonfigurasi Akses Sumber Daya Lokal Menggunakan Antarmuka Baris AWS Perintah](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [CreateResourceDefinition](#) di Referensi AWS CLI Perintah.

## create-software-update-job

Contoh kode berikut menunjukkan cara menggunakan `create-software-update-job`.

### AWS CLI

Untuk membuat pekerjaan pembaruan perangkat lunak untuk inti

`create-software-update-job` Contoh berikut membuat pekerjaan pembaruan over-the-air (OTA) untuk memperbarui perangkat lunak AWS IoT Greengrass Core pada inti yang namanya. `MyFirstGroup_Core` Perintah ini memerlukan peran IAM yang memungkinkan akses ke paket pembaruan perangkat lunak di Amazon S3 dan `iot.amazonaws.com` termasuk sebagai entitas tepercaya.

```
aws greengrass create-software-update-job \  
  --update-targets-architecture armv7l \  
  --update-targets ["arn:aws:iot:us-west-2:123456789012:thing/MyFirstGroup_Core \  
  \"] \  
  --update-targets-operating-system raspbian \  
  --software-to-update core \  
  --s3-url-signer-role arn:aws:iam::123456789012:role/OTA_signer_role \  
  --update-agent-log-level WARN
```

### Output:

```
{  
  "IotJobId": "GreengrassUpdateJob_30b353e3-3af7-4786-be25-4c446663c09e",  
  "IotJobArn": "arn:aws:iot:us-west-2:123456789012:job/  
GreengrassUpdateJob_30b353e3-3af7-4786-be25-4c446663c09e",  
  "PlatformSoftwareVersion": "1.9.3"  
}
```

Untuk informasi lebih lanjut, lihat [Pembaruan OTA Perangkat Lunak Inti AWS Greengrass IoT di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [CreateSoftwareUpdateJob](#) di Referensi AWS CLI Perintah.

## create-subscription-definition-version

Contoh kode berikut menunjukkan cara menggunakan `create-subscription-definition-version`.



## AWS CLI

Untuk membuat versi baru dari definisi langganan

`create-subscription-definition-version` Contoh berikut membuat versi baru dari definisi langganan yang berisi tiga langganan: pemberitahuan pemicu, input suhu, dan status keluaran.

```
aws greengrass create-subscription-definition-version \
  --subscription-definition-id "9d611d57-5d5d-44bd-a3b4-feccbdd69112" \
  --subscriptions "[{\\"Id\\": \\"TriggerNotification\\", \\"Source\\":
  \\"arn:aws:lambda:us-west-2:123456789012:function:TempMonitor:GG_TempMonitor
  \\", \\"Subject\\": \\"twilio/txt\\", \\"Target\\": \\"arn:aws:greengrass:us-west-2:/:
  connectors/TwilioNotifications/versions/1\\"},{\\"Id\\": \\"TemperatureInput\\", \\"Source
  \\": \\"cloud\\", \\"Subject\\": \\"temperature/input\\", \\"Target\\": \\"arn:aws:lambda:us-
  west-2:123456789012:function:TempMonitor:GG_TempMonitor\\"},{\\"Id\\": \\"OutputStatus
  \\", \\"Source\\": \\"arn:aws:greengrass:us-west-2:/:connectors/TwilioNotifications/
  versions/1\\"}, \\"Subject\\": \\"twilio/message/status\\", \\"Target\\": \\"cloud\\"}]]"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
  subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112/versions/7b65dfae-50b6-4d0f-
  b3e0-27728bfb0620",
  "CreationTimestamp": "2019-06-24T21:21:33.837Z",
  "Id": "9d611d57-5d5d-44bd-a3b4-feccbdd69112",
  "Version": "7b65dfae-50b6-4d0f-b3e0-27728bfb0620"
}
```

- Untuk detail API, lihat [CreateSubscriptionDefinitionVersion](#) di Referensi AWS CLI Perintah.

## `create-subscription-definition`

Contoh kode berikut menunjukkan cara menggunakan `create-subscription-definition`.

## AWS CLI

Untuk membuat definisi langganan

`create-subscription-definition` Contoh berikut membuat definisi langganan dan menentukan versi awalnya. Versi awal berisi tiga langganan: satu untuk topik MQTT tempat

konektor berlangganan, satu untuk memungkinkan fungsi menerima pembacaan suhu dari IoT AWS , dan satu lagi untuk memungkinkan AWS IoT menerima informasi status dari konektor. Contoh ini menyediakan ARN untuk alias fungsi Lambda yang dibuat sebelumnya dengan menggunakan perintah `aws lambda create-alias`

```
aws greengrass create-subscription-definition \
  --initial-version "{\"Subscriptions\": [{\"Id\":
  \"TriggerNotification\", \"Source\": \"arn:aws:lambda:us-
  west-2:123456789012:function:TempMonitor:GG_TempMonitor\", \"Subject\":
  \"twilio/txt\", \"Target\": \"arn:aws:greengrass:us-west-2::/connectors/
  TwilioNotifications/versions/1\"},{\"Id\": \"TemperatureInput\", \"Source\":
  \"cloud\", \"Subject\": \"temperature/input\", \"Target\": \"arn:aws:lambda:us-
  west-2:123456789012:function:TempMonitor:GG_TempMonitor\"},{\"Id\": \"OutputStatus
  \", \"Source\": \"arn:aws:greengrass:us-west-2::/connectors/TwilioNotifications/
  versions/1\", \"Subject\": \"twilio/message/status\", \"Target\": \"cloud\"}]}"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
  subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112",
  "CreationTimestamp": "2019-06-19T22:34:26.677Z",
  "Id": "9d611d57-5d5d-44bd-a3b4-feccbdd69112",
  "LastUpdatedTimestamp": "2019-06-19T22:34:26.677Z",
  "LatestVersion": "aa645c47-ac90-420d-9091-8c7ffa4f103f",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
  definition/subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112/versions/aa645c47-
  ac90-420d-9091-8c7ffa4f103f"
}
```

Untuk informasi selengkapnya, lihat [Memulai Konektor \(CLI\)](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [CreateSubscriptionDefinition](#) di Referensi AWS CLI Perintah.

## **delete-connector-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-connector-definition`.

### AWS CLI

Untuk menghapus definisi konektor

`delete-connector-definition` Contoh berikut menghapus definisi konektor Greengrass yang ditentukan. Jika Anda menghapus definisi konektor yang digunakan oleh grup, grup tersebut tidak dapat berhasil diterapkan.

```
aws greengrass delete-connector-definition \  
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteConnectorDefinition](#) di Referensi AWS CLI Perintah.

## **delete-core-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-core-definition`.

### AWS CLI

Untuk menghapus definisi inti

`delete-core-definition` Contoh berikut menghapus definisi inti Greengrass yang ditentukan, termasuk semua versi. Jika Anda menghapus inti yang terkait dengan grup Greengrass, grup tersebut tidak dapat digunakan dengan sukses.

```
aws greengrass delete-core-definition \  
  --core-definition-id "ff36cc5f-9f98-4994-b468-9d9b6dc52abd"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteCoreDefinition](#) di Referensi AWS CLI Perintah.

## **delete-device-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-device-definition`.

### AWS CLI

Untuk menghapus definisi perangkat

`delete-device-definition` Contoh berikut menghapus definisi perangkat yang ditentukan, termasuk semua versinya. Jika Anda menghapus versi definisi perangkat yang digunakan oleh versi grup, versi grup tidak dapat berhasil digunakan.

```
aws greengrass delete-device-definition \  
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteDeviceDefinition](#) di Referensi AWS CLI Perintah.

## **delete-function-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-function-definition`.

### AWS CLI

Untuk menghapus definisi fungsi

`delete-function-definition` Contoh berikut menghapus definisi fungsi Greengrass yang ditentukan. Jika Anda menghapus definisi fungsi yang digunakan oleh grup, grup tersebut tidak dapat berhasil diterapkan.

```
aws greengrass delete-function-definition \  
  --function-definition-id "fd4b906a-dff3-4c1b-96eb-52ebfcfac06a"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteFunctionDefinition](#) di Referensi AWS CLI Perintah.

## **delete-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-group`.

### AWS CLI

Untuk menghapus grup

`delete-group` Contoh berikut menghapus grup Greengrass yang ditentukan.

```
aws greengrass delete-group \  
  --group-id "4e22bd92-898c-436b-ade5-434d883ff749"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteGroup](#) di Referensi AWS CLI Perintah.

## **delete-logger-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-logger-definition`.

### AWS CLI

Untuk menghapus definisi logger

`delete-logger-definition` Contoh berikut menghapus definisi logger yang ditentukan, termasuk semua versi definisi logger. Jika Anda menghapus versi definisi logger yang digunakan oleh versi grup, versi grup tidak dapat berhasil digunakan.

```
aws greengrass delete-logger-definition \  
  --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pemantauan dengan Log AWS Greengrass IoT di Panduan Pengembang Greengrass IoT](#).AWS

- Untuk detail API, lihat [DeleteLoggerDefinition](#) di Referensi AWS CLI Perintah.

## **delete-resource-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-definition`.

### AWS CLI

Untuk menghapus definisi sumber daya

`delete-resource-definition` Contoh berikut menghapus definisi sumber daya yang ditentukan, termasuk semua versi sumber daya. Jika Anda menghapus definisi sumber daya yang digunakan oleh grup, grup tersebut tidak dapat berhasil diterapkan.

```
aws greengrass delete-resource-definition \  
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteResourceDefinition](#) di Referensi AWS CLI Perintah.

## **delete-subscription-definition**

Contoh kode berikut menunjukkan cara menggunakan `delete-subscription-definition`.

### AWS CLI

Untuk menghapus definisi langganan

`delete-subscription-definition` Contoh berikut menghapus definisi langganan Greengrass yang ditentukan. Jika Anda menghapus langganan yang sedang digunakan oleh grup, grup tersebut tidak dapat berhasil diterapkan.

```
aws greengrass delete-subscription-definition \  
  --subscription-definition-id "cd6f1c37-d9a4-4e90-be94-01a7404f5967"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteSubscriptionDefinition](#) di Referensi AWS CLI Perintah.

## **disassociate-role-from-group**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-role-from-group`.

### AWS CLI

Untuk memisahkan peran dari kelompok Greengrass

`disassociate-role-from-group` Contoh berikut memisahkan peran IAM dari grup Greengrass yang ditentukan.

```
aws greengrass disassociate-role-from-group \  
  --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b
```

Output:

```
{  
  "DisassociatedAt": "2019-09-10T20:05:49Z"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Peran Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [DisassociateRoleFromGroup](#) di Referensi AWS CLI Perintah.

## **disassociate-service-role-from-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-service-role-from-account`.

### AWS CLI

Untuk memisahkan peran layanan dari akun Anda AWS

`disassociate-service-role-from-account` Contoh berikut menghapus peran layanan yang terkait dengan AWS akun Anda. Jika Anda tidak menggunakan peran layanan di AWS Wilayah mana pun, gunakan `delete-role-policy` perintah untuk melepaskan kebijakan `AWSGreengrassResourceAccessRolePolicy` terkelola dari peran, lalu gunakan `delete-role` perintah untuk menghapus peran tersebut.

```
aws greengrass disassociate-service-role-from-account
```

Output:

```
{
  "DisassociatedAt": "2019-06-25T22:12:55Z"
}
```

Untuk informasi selengkapnya, lihat Peran [Layanan Greengrass di Panduan Pengembang Greengrass](#) IoT AWS .

- Untuk detail API, lihat [DisassociateServiceRoleFromAccount](#) di Referensi AWS CLI Perintah.

## **get-associated-role**

Contoh kode berikut menunjukkan cara menggunakan `get-associated-role`.

### AWS CLI

Untuk mendapatkan peran yang terkait dengan kelompok Greengrass

`get-associated-role` Contoh berikut mendapatkan peran IAM yang terkait dengan grup Greengrass yang ditentukan. Peran grup digunakan oleh fungsi dan konektor Lambda lokal untuk mengakses AWS layanan.

```
aws greengrass get-associated-role \  
  --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b
```

Output:

```
{  
  "RoleArn": "arn:aws:iam::123456789012:role/GG-Group-Role",  
  "AssociatedAt": "2019-09-10T20:03:30Z"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Peran Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [GetAssociatedRole](#) di Referensi AWS CLI Perintah.

## `get-bulk-deployment-status`

Contoh kode berikut menunjukkan cara menggunakan `get-bulk-deployment-status`.

### AWS CLI

Untuk memeriksa status penyebaran massal Anda

`get-bulk-deployment-status` Contoh berikut mengambil informasi status untuk operasi penyebaran massal yang ditentukan. Dalam contoh ini, file yang menentukan grup yang akan digunakan memiliki catatan input yang tidak valid.

```
aws greengrass get-bulk-deployment-status \  
  --bulk-deployment-id "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
```

Output:

```
{  
  "BulkDeploymentMetrics": {  
    "InvalidInputRecords": 1,  
    "RecordsProcessed": 1,  
  }  
}
```



```
    "RetryAttempts": 0
  },
  "BulkDeploymentStatus": "Completed",
  "CreatedAt": "2019-06-25T16:11:33.265Z",
  "tags": {}
}
```

Untuk informasi selengkapnya, lihat [Membuat Penerapan Massal untuk Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [GetBulkDeploymentStatus](#) di Referensi AWS CLI Perintah.

## get-connectivity-info

Contoh kode berikut menunjukkan cara menggunakan `get-connectivity-info`.

### AWS CLI

Untuk mendapatkan informasi konektivitas untuk inti Greengrass

`get-connectivity-info` Contoh berikut menampilkan titik akhir yang dapat digunakan perangkat untuk terhubung ke inti Greengrass yang ditentukan. Informasi konektivitas adalah daftar alamat IP atau nama domain, dengan nomor port yang sesuai dan metadata yang ditentukan pelanggan opsional.

```
aws greengrass get-connectivity-info \
  --thing-name "MyGroup_Core"
```

Output:

```
{
  "ConnectivityInfo": [
    {
      "Metadata": "",
      "PortNumber": 8883,
      "HostAddress": "127.0.0.1",
      "Id": "AUTOIP_127.0.0.1_0"
    },
    {
      "Metadata": "",
      "PortNumber": 8883,
```

```

    "HostAddress": "192.168.1.3",
    "Id": "AUTOIP_192.168.1.3_1"
  },
  {
    "Metadata": "",
    "PortNumber": 8883,
    "HostAddress": "::1",
    "Id": "AUTOIP_::1_2"
  },
  {
    "Metadata": "",
    "PortNumber": 8883,
    "HostAddress": "fe80::1e69:ed93:f5b:f6d",
    "Id": "AUTOIP_fe80::1e69:ed93:f5b:f6d_3"
  }
]
}

```

- Untuk detail API, lihat [GetConnectivityInfo](#) di Referensi AWS CLI Perintah.

## get-connector-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-connector-definition-version`.

### AWS CLI

Untuk mendapatkan kembali informasi tentang versi tertentu dari definisi konektor

`get-connector-definition-version` Contoh berikut mengambil informasi tentang versi tertentu dari definisi konektor yang ditentukan. Untuk mengambil ID dari semua versi definisi konektor, gunakan `list-connector-definition-versions` perintah. Untuk mengambil ID dari versi terakhir yang ditambahkan ke definisi konektor, gunakan `get-connector-definition` perintah dan periksa `LatestVersion` properti.

```

aws greengrass get-connector-definition-version \
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8" \
  --connector-definition-version-id "63c57963-c7c2-4a26-a7e2-7bf478ea2623"

```

Output:

```
{
```

```

    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-c7c2-4a26-
a7e2-7bf478ea2623",
    "CreationTimestamp": "2019-06-19T19:30:01.300Z",
    "Definition": {
      "Connectors": [
        {
          "ConnectorArn": "arn:aws:greengrass:us-west-2:./connectors/SNS/
versions/1",
          "Id": "MySNSConnector",
          "Parameters": {
            "DefaultSNSArn": "arn:aws:sns:us-
west-2:123456789012:GGConnectorTopic"
          }
        }
      ]
    },
    "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    "Version": "63c57963-c7c2-4a26-a7e2-7bf478ea2623"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengintegrasikan dengan Layanan dan Protokol Menggunakan Konektor Greengrass di Panduan Pengembang Greengrass IoT](#).AWS

- Untuk detail API, lihat [GetConnectorDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-connector-definition

Contoh kode berikut menunjukkan cara menggunakan `get-connector-definition`.

### AWS CLI

Untuk mengambil informasi tentang definisi konektor

`get-connector-definition` Contoh berikut mengambil informasi tentang definisi konektor yang ditentukan. Untuk mengambil ID definisi konektor Anda, gunakan `list-connector-definitions` perintah.

```
aws greengrass get-connector-definition \
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
  "CreationTimestamp": "2019-06-19T19:30:01.300Z",
  "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
  "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
  "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-
c7c2-4a26-a7e2-7bf478ea2623",
  "Name": "MySNSConnector",
  "tags": {}
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan dengan Layanan dan Protokol Menggunakan Konektor Greengrass di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [GetConnectorDefinition](#) di Referensi AWS CLI Perintah.

## get-core-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-core-definition-version`.

### AWS CLI

Untuk mengambil detail tentang versi tertentu dari definisi inti Greengrass

`get-core-definition-version` Contoh berikut mengambil informasi tentang versi tertentu dari definisi inti yang ditentukan. Untuk mengambil ID dari semua versi definisi inti, gunakan `list-core-definition-versions` perintah. Untuk mengambil ID dari versi terakhir yang ditambahkan ke definisi inti, gunakan `get-core-definition` perintah dan periksa `LatestVersion` properti.

```
aws greengrass get-core-definition-version \
  --core-definition-id "c906ed39-a1e3-4822-a981-7b9bd57b4b46" \
  --core-definition-version-id "42aeeac3-fd9d-4312-a8fd-ffa9404a20e0"
```

Output:

```
{
```

```

    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/
c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeac3-fd9d-4312-a8fd-ffa9404a20e0",
    "CreationTimestamp": "2019-06-18T16:21:21.351Z",
    "Definition": {
      "Cores": [
        {
          "CertificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/928dea7b82331b47c3ff77b0e763fc5e64e2f7c884e6ef391baed9b6b8e21b45",
          "Id": "1a39aac7-0885-4417-91f6-23e4cea6c511",
          "SyncShadow": false,
          "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/
GGGroup4Pi3_Core"
        }
      ]
    },
    "Id": "c906ed39-a1e3-4822-a981-7b9bd57b4b46",
    "Version": "42aeac3-fd9d-4312-a8fd-ffa9404a20e0"
  }
}

```

- Untuk detail API, lihat [GetCoreDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-core-definition

Contoh kode berikut menunjukkan cara menggunakan `get-core-definition`.

### AWS CLI

Untuk mengambil detail untuk definisi inti Greengrass

`get-core-definition` Contoh berikut mengambil informasi tentang definisi inti yang ditentukan. Untuk mengambil ID definisi inti Anda, gunakan `list-core-definitions` perintah.

```

aws greengrass get-core-definition \
  --core-definition-id "c906ed39-a1e3-4822-a981-7b9bd57b4b46"

```

Output:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/237d6916-27cf-457f-ba0c-e86cfb5d25cd",
  "CreationTimestamp": "2018-10-18T04:47:06.721Z",

```

```

    "Id": "237d6916-27cf-457f-ba0c-e86cfb5d25cd",
    "LastUpdatedTimestamp": "2018-10-18T04:47:06.721Z",
    "LatestVersion": "bd2cd6d4-2bc5-468a-8962-39e071e34b68",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/237d6916-27cf-457f-ba0c-e86cfb5d25cd/versions/
bd2cd6d4-2bc5-468a-8962-39e071e34b68",
    "tags": {}
}

```

- Untuk detail API, lihat [GetCoreDefinition](#) di Referensi AWS CLI Perintah.

## get-deployment-status

Contoh kode berikut menunjukkan cara menggunakan `get-deployment-status`.

### AWS CLI

Untuk mengambil status penerapan

`get-deployment-status` Contoh berikut mengambil status untuk penyebaran tertentu dari grup Greengrass yang ditentukan. Untuk mendapatkan ID penyebaran, gunakan `list-deployments` perintah dan tentukan ID grup.

```

aws greengrass get-deployment-status \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731" \
  --deployment-id "1065b8a0-812b-4f21-9d5d-e89b232a530f"

```

Output:

```

{
  "DeploymentStatus": "Success",
  "DeploymentType": "NewDeployment",
  "UpdatedAt": "2019-06-18T17:04:44.761Z"
}

```

- Untuk detail API, lihat [GetDeploymentStatus](#) di Referensi AWS CLI Perintah.

## get-device-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-device-definition-version`.

## AWS CLI

Untuk mendapatkan versi definisi perangkat

`get-device-definition-version` Contoh berikut mengambil informasi tentang versi tertentu dari definisi perangkat yang ditentukan. Untuk mengambil ID dari semua versi definisi perangkat, gunakan `list-device-definition-versions` perintah. Untuk mengambil ID versi terakhir yang ditambahkan ke definisi perangkat, gunakan `get-device-definition` perintah dan periksa `LatestVersion` properti.

```
aws greengrass get-device-definition-version \
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
  --device-definition-version-id "83c13984-6fed-447e-84d5-5b8aa45d5f71"
```

Output:

```
{
  "Definition": {
    "Devices": [
      {
        "CertificateArn": "arn:aws:iot:us-west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02",
        "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/ExteriorTherm",
        "SyncShadow": true,
        "Id": "ExteriorTherm"
      },
      {
        "CertificateArn": "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
        "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",
        "SyncShadow": true,
        "Id": "InteriorTherm"
      }
    ]
  },
  "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
  "CreationTimestamp": "2019-09-11T00:15:09.838Z",
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/versions/83c13984-6fed-447e-84d5-5b8aa45d5f71"
```

```
}
```

- Untuk detail API, lihat [GetDeviceDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-device-definition

Contoh kode berikut menunjukkan cara menggunakan `get-device-definition`.

### AWS CLI

Untuk mendapatkan definisi perangkat

`get-device-definition` Contoh berikut mengambil informasi tentang definisi perangkat yang ditentukan. Untuk mengambil ID definisi perangkat Anda, gunakan `list-device-definitions` perintah.

```
aws greengrass get-device-definition \  
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Output:

```
{  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/  
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",  
  "Name": "TemperatureSensors",  
  "tags": {},  
  "LastUpdatedTimestamp": "2019-09-11T00:19:03.698Z",  
  "LatestVersion": "83c13984-6fed-447e-84d5-5b8aa45d5f71",  
  "CreationTimestamp": "2019-09-11T00:11:06.197Z",  
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"  
}
```

- Untuk detail API, lihat [GetDeviceDefinition](#) di Referensi AWS CLI Perintah.

## get-function-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-function-definition-version`.



## AWS CLI

Untuk mengambil detail tentang versi tertentu dari fungsi Lambda

Berikut ini `get-function-definition-version` mengambil informasi tentang versi tertentu dari definisi fungsi yang ditentukan. Untuk mengambil ID dari semua versi definisi fungsi, gunakan `list-function-definition-versions` perintah. Untuk mengambil ID dari versi terakhir yang ditambahkan ke definisi fungsi, gunakan `get-function-definition` perintah dan periksa `LatestVersion` properti.

```
aws greengrass get-function-definition-version \
  --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85" \
  --function-definition-version-id "9748fda7-1589-4fcc-ac94-f5559e88678b"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/9748fda7-1589-4fcc-ac94-
f5559e88678b",
  "CreationTimestamp": "2019-06-18T17:04:30.776Z",
  "Definition": {
    "Functions": [
      {
        "FunctionArn": "arn:aws:lambda::function:GGIPDetector:1",
        "FunctionConfiguration": {
          "Environment": {},
          "MemorySize": 32768,
          "Pinned": true,
          "Timeout": 3
        },
        "Id": "26b69bdb-e547-46bc-9812-84ec04b6cc8c"
      },
      {
        "FunctionArn": "arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld:GG_HelloWorld",
        "FunctionConfiguration": {
          "EncodingType": "json",
          "Environment": {
            "Variables": {}
          },
          "MemorySize": 16384,
```

```

        "Pinned": true,
        "Timeout": 25
    },
    "Id": "384465a8-eedf-48c6-b793-4c35f7bfae9b"
}
]
},
"Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
"Version": "9748fda7-1589-4fcc-ac94-f5559e88678b"
}

```

- Untuk detail API, lihat [GetFunctionDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-function-definition

Contoh kode berikut menunjukkan cara menggunakan `get-function-definition`.

### AWS CLI

Untuk mengambil definisi fungsi

`get-function-definition` Contoh berikut menampilkan rincian untuk definisi fungsi yang ditentukan. Untuk mengambil ID dari definisi fungsi Anda, gunakan `list-function-definitions` perintah.

```
aws greengrass get-function-definition \
  --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
  "CreationTimestamp": "2019-06-18T16:21:21.431Z",
  "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
  "LastUpdatedTimestamp": "2019-06-18T16:21:21.431Z",
  "LatestVersion": "9748fda7-1589-4fcc-ac94-f5559e88678b",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
  "tags": {}
}
```

- Untuk detail API, lihat [GetFunctionDefinition](#) di Referensi AWS CLI Perintah.

## get-group-certificate-authority

Contoh kode berikut menunjukkan cara menggunakan `get-group-certificate-authority`.

### AWS CLI

Untuk mengambil CA yang terkait dengan kelompok Greengrass

`get-group-certificate-authority` Contoh berikut mengambil otoritas sertifikat (CA) yang terkait dengan grup Greengrass yang ditentukan. Untuk mendapatkan ID otoritas sertifikat, gunakan `list-group-certificate-authorities` perintah dan tentukan ID grup.

```
aws greengrass get-group-certificate-authority \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731" \
  --certificate-authority-id
  "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6"
```

### Output:

```
{
  "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/certificateauthorities/f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
  "GroupCertificateAuthorityId":
  "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
  "PemEncodedCertificate": "-----BEGIN CERTIFICATE-----
MIICiCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBWEXAMPLEGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDEXAMPLEEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAEEXAMPLESDBb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVWxHZAAd
BgkqhkiG9w0BCQEWEG5vb25lQGFTYXpvbi5jEXAMPLENMTIwNDI0MjA0NTIxWhcN
MTIwNDI0MjA0EXAMPLEBIDELMAKGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWEXAMPLEDASBgNVBAwTC01BTSDBb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVWEXAMPLEGkqhkiG9w0BCQEWEG5vb25lQGFT
YXpvbi5EXAMPLE8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvYsWtC2XADZ4nB+BLYgVIk60CEXAMPLE93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswYEXAMPLEEgpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKEXAMPLEAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
```

```
-----END CERTIFICATE-----\n"}
```

- Untuk detail API, lihat [GetGroupCertificateAuthority](#) di Referensi AWS CLI Perintah.

## get-group-certificate-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-group-certificate-configuration`.

### AWS CLI

Untuk mengambil konfigurasi untuk otoritas sertifikat yang digunakan oleh grup Greengrass

`get-group-certificate-configuration` Contoh berikut mengambil konfigurasi untuk otoritas sertifikat (CA) yang digunakan oleh kelompok Greengrass tertentu.

```
aws greengrass get-group-certificate-configuration \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
  "CertificateAuthorityExpiryInMilliseconds": 2524607999000,
  "CertificateExpiryInMilliseconds": 604800000,
  "GroupId": "1013db12-8b58-45ff-acc7-704248f66731"
}
```

- Untuk detail API, lihat [GetGroupCertificateConfiguration](#) di Referensi AWS CLI Perintah.

## get-group-version

Contoh kode berikut menunjukkan cara menggunakan `get-group-version`.

### AWS CLI

Untuk mengambil informasi tentang versi grup Greengrass

`get-group-version` Contoh berikut mengambil informasi tentang versi tertentu dari grup yang ditentukan. Untuk mengambil ID dari semua versi grup, gunakan `list-group-versions`

perintah. Untuk mengambil ID dari versi terakhir yang ditambahkan ke grup, gunakan `get-group` perintah dan periksa `LatestVersion` properti.

```
aws greengrass get-group-version \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731" \
  --group-version-id "115136b3-cfd7-4462-b77f-8741a4b00e5e"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
  "CreationTimestamp": "2019-06-18T17:04:30.915Z",
  "Definition": {
    "CoreDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeac3-
fd9d-4312-a8fd-ffa9404a20e0",
    "FunctionDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
    "SubscriptionDefinitionVersionArn": "arn:aws:greengrass:us-
west-2:123456789012:/greengrass/definition/subscriptions/70e49321-83d5-45d2-
bc09-81f4917ae152/versions/88ae8699-12ac-4663-ba3f-4d7f0519140b"
  },
  "Id": "1013db12-8b58-45ff-acc7-704248f66731",
  "Version": "115136b3-cfd7-4462-b77f-8741a4b00e5e"
}
```

- Untuk detail API, lihat [GetGroupVersion](#) di Referensi AWS CLI Perintah.

## get-group

Contoh kode berikut menunjukkan cara menggunakan `get-group`.

### AWS CLI

Untuk mengambil informasi tentang kelompok Greengrass

`get-group` Contoh berikut mengambil informasi tentang kelompok Greengrass tertentu. Untuk mengambil ID grup Anda, gunakan `list-groups` perintah.

```
aws greengrass get-group \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/1013db12-8b58-45ff-acc7-704248f66731",  
  "CreationTimestamp": "2019-06-18T16:21:21.457Z",  
  "Id": "1013db12-8b58-45ff-acc7-704248f66731",  
  "LastUpdatedTimestamp": "2019-06-18T16:21:21.457Z",  
  "LatestVersion": "115136b3-cfd7-4462-b77f-8741a4b00e5e",  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-  
b77f-8741a4b00e5e",  
  "Name": "GGGroup4Pi3",  
  "tags": {}  
}
```

- Untuk detail API, lihat [GetGroup](#) di Referensi AWS CLI Perintah.

## get-logger-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-logger-definition-version`.

### AWS CLI

Untuk mengambil informasi tentang versi definisi logger

`get-logger-definition-version` Contoh berikut mengambil informasi tentang versi tertentu dari definisi logger yang ditentukan. Untuk mengambil ID dari semua versi definisi logger, gunakan perintah `list-logger-definition-versions` Untuk mengambil ID versi terakhir yang ditambahkan ke definisi logger, gunakan `get-logger-definition` perintah dan periksa properti `LatestVersion`

```
aws greengrass get-logger-definition-version \  
  --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23" \  
  --logger-definition-version-id "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
  "CreationTimestamp": "2019-05-08T16:10:13.866Z",
  "Definition": {
    "Loggers": []
  },
  "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
  "Version": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
}
```

- Untuk detail API, lihat [GetLoggerDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-logger-definition

Contoh kode berikut menunjukkan cara menggunakan `get-logger-definition`.

### AWS CLI

Untuk mengambil informasi tentang definisi logger

`get-logger-definition` Contoh berikut mengambil informasi tentang definisi logger yang ditentukan. Untuk mengambil ID definisi logger Anda, gunakan perintah `list-logger-definitions`

```
aws greengrass get-logger-definition \
  --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23",
  "CreationTimestamp": "2019-05-08T16:10:13.809Z",
  "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
  "LastUpdatedTimestamp": "2019-05-08T16:10:13.809Z",
  "LatestVersion": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
}
```

```
"tags": {}  
}
```

- Untuk detail API, lihat [GetLoggerDefinition](#) di Referensi AWS CLI Perintah.

## get-resource-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-resource-definition-version`.

### AWS CLI

Untuk mengambil informasi tentang versi tertentu dari definisi sumber daya

`get-resource-definition-version` Contoh berikut mengambil informasi tentang versi tertentu dari definisi sumber daya yang ditentukan. Untuk mengambil ID dari semua versi definisi sumber daya, gunakan `list-resource-definition-versions` perintah. Untuk mengambil ID versi terakhir yang ditambahkan ke definisi sumber daya, gunakan `get-resource-definition` perintah dan periksa `LatestVersion` properti.

```
aws greengrass get-resource-definition-version \  
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658" \  
  --resource-definition-version-id "26e8829a-491a-464d-9c87-664bf6f6f2be"
```

### Output:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/  
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",  
  "CreationTimestamp": "2019-06-19T16:40:59.392Z",  
  "Definition": {  
    "Resources": [  
      {  
        "Id": "26ff3f7b-839a-4217-9fdc-a218308b3963",  
        "Name": "usb-port",  
        "ResourceDataContainer": {  
          "LocalDeviceResourceData": {  
            "GroupOwnerSetting": {  
              "AutoAddGroupOwner": false  
            },  
            "SourcePath": "/dev/bus/usb"  
          }  
        }  
      }  
    ]  
  }  
}
```



```

    }
  }
]
},
  "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
  "Version": "26e8829a-491a-464d-9c87-664bf6f6f2be"
}

```

- Untuk detail API, lihat [GetResourceDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-resource-definition

Contoh kode berikut menunjukkan cara menggunakan `get-resource-definition`.

### AWS CLI

Untuk mengambil informasi tentang definisi sumber daya

`get-resource-definition` Contoh berikut mengambil informasi tentang definisi sumber daya yang ditentukan. Untuk mengambil ID definisi sumber daya Anda, gunakan `list-resource-definitions` perintah.

```
aws greengrass get-resource-definition \
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658",
  "CreationTimestamp": "2019-06-19T16:40:59.261Z",
  "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
  "LastUpdatedTimestamp": "2019-06-19T16:40:59.261Z",
  "LatestVersion": "26e8829a-491a-464d-9c87-664bf6f6f2be",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
  "tags": {}
}
```

- Untuk detail API, lihat [GetResourceDefinition](#) di Referensi AWS CLI Perintah.

## get-service-role-for-account

Contoh kode berikut menunjukkan cara menggunakan `get-service-role-for-account`.

### AWS CLI

Untuk mengambil detail untuk peran layanan yang dilampirkan ke akun Anda

`get-service-role-for-account` Contoh berikut mengambil informasi tentang peran layanan yang dilampirkan ke AWS akun Anda.

```
aws greengrass get-service-role-for-account
```

Output:

```
{
  "AssociatedAt": "2018-10-18T15:59:20Z",
  "RoleArn": "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
}
```

Untuk informasi selengkapnya, lihat Peran [Layanan Greengrass di Panduan Pengembang Greengrass](#) IoT AWS .

- Untuk detail API, lihat [GetServiceRoleForAccount](#) di Referensi AWS CLI Perintah.

## get-subscription-definition-version

Contoh kode berikut menunjukkan cara menggunakan `get-subscription-definition-version`.

### AWS CLI

Untuk mengambil informasi tentang versi tertentu dari definisi langganan

`get-subscription-definition-version` Contoh berikut mengambil informasi tentang versi tertentu dari definisi langganan yang ditentukan. Untuk mengambil ID dari semua versi definisi langganan, gunakan `list-subscription-definition-versions` perintah. Untuk mengambil ID versi terakhir yang ditambahkan ke definisi langganan, gunakan `get-subscription-definition` perintah dan periksa `LatestVersion` properti.

```
aws greengrass get-subscription-definition-version \
  --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152" \
```

```
--subscription-definition-version-id "88ae8699-12ac-4663-ba3f-4d7f0519140b"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/88ae8699-12ac-4663-
ba3f-4d7f0519140b",
  "CreationTimestamp": "2019-06-18T17:03:52.499Z",
  "Definition": {
    "Subscriptions": [
      {
        "Id": "692c4484-d89f-4f64-8edd-1a041a65e5b6",
        "Source": "arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld:GG_HelloWorld",
        "Subject": "hello/world",
        "Target": "cloud"
      }
    ]
  },
  "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
  "Version": "88ae8699-12ac-4663-ba3f-4d7f0519140b"
}
```

- Untuk detail API, lihat [GetSubscriptionDefinitionVersion](#) di Referensi AWS CLI Perintah.

## get-subscription-definition

Contoh kode berikut menunjukkan cara menggunakan `get-subscription-definition`.

### AWS CLI

Untuk mengambil informasi tentang definisi langganan

`get-subscription-definition` Contoh berikut mengambil informasi tentang definisi langganan yang ditentukan. Untuk mengambil ID definisi langganan Anda, gunakan `list-subscription-definitions` perintah.

```
aws greengrass get-subscription-definition \
  --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152"
```

Output:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152",
  "CreationTimestamp": "2019-06-18T17:03:52.392Z",
  "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
  "LastUpdatedTimestamp": "2019-06-18T17:03:52.392Z",
  "LatestVersion": "88ae8699-12ac-4663-ba3f-4d7f0519140b",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b",
  "tags": {}
}
```

- Untuk detail API, lihat [GetSubscriptionDefinition](#) di Referensi AWS CLI Perintah.

## get-thing-runtime-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-thing-runtime-configuration`.

### AWS CLI

Untuk mengambil konfigurasi runtime dari inti Greengrass

`get-thing-runtime-configuration` Contoh berikut mengambil konfigurasi runtime dari inti Greengrass. Sebelum Anda dapat mengambil konfigurasi runtime, Anda harus menggunakan `update-thing-runtime-configuration` perintah untuk membuat konfigurasi runtime untuk inti.

```
aws greengrass get-thing-runtime-configuration \
  --thing-name SampleGreengrassCore
```

Output:

```
{
  "RuntimeConfiguration": {
    "TelemetryConfiguration": {
      "ConfigurationSyncStatus": "OutOfSync",
      "Telemetry": "On"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi setelan telemetri di Panduan Pengembang Greengrass AWS IoT](#).

- Untuk detail API, lihat [GetThingRuntimeConfiguration](#) di Referensi AWS CLI Perintah.

## list-bulk-deployment-detailed-reports

Contoh kode berikut menunjukkan cara menggunakan `list-bulk-deployment-detailed-reports`.

### AWS CLI

Untuk mencantumkan informasi tentang penerapan individual dalam penyebaran massal

`list-bulk-deployment-detailed-reports` Contoh berikut menampilkan informasi tentang penyebaran individu dalam operasi penyebaran massal, termasuk status.

```
aws greengrass list-bulk-deployment-detailed-reports \
  --bulk-deployment-id 42ce9c42-489b-4ed4-b905-8996aa50ef9d
```

### Output:

```
{
  "Deployments": [
    {
      "DeploymentType": "NewDeployment",
      "DeploymentStatus": "Success",
      "DeploymentId": "123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/
deployments/123456789012:123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/
versions/123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
      "CreatedAt": "2020-01-21T21:34:16.501Z"
    },
    {
      "DeploymentType": "NewDeployment",
      "DeploymentStatus": "InProgress",
      "DeploymentId": "123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE55555/
deployments/123456789012:123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
```

```

        "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE55555/versions/a1b2c3d4-5678-90ab-cdef-
EXAMPLE66666",
        "CreatedAt": "2020-01-21T21:34:16.486Z"
    },
    ...
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Penerapan Massal untuk Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [ListBulkDeploymentDetailedReports](#) di Referensi AWS CLI Perintah.

## list-bulk-deployments

Contoh kode berikut menunjukkan cara menggunakan `list-bulk-deployments`.

### AWS CLI

Untuk membuat daftar penyebaran massal

`list-bulk-deployments` Contoh berikut mencantumkan semua penyebaran massal.

```
aws greengrass list-bulk-deployments
```

Output:

```

{
  "BulkDeployments": [
    {
      "BulkDeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/bulk/deployments/870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
      "BulkDeploymentId": "870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
      "CreatedAt": "2019-06-25T16:11:33.265Z"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Membuat Penerapan Massal untuk Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [ListBulkDeployments](#) di Referensi AWS CLI Perintah.

## list-connector-definition-versions

Contoh kode berikut menunjukkan cara menggunakan `list-connector-definition-versions`.

### AWS CLI

Untuk membuat daftar versi yang tersedia untuk definisi konektor

`list-connector-definition-versions` Contoh berikut mencantumkan versi yang tersedia untuk definisi konektor yang ditentukan. Gunakan `list-connector-definitions` perintah untuk mendapatkan ID definisi konektor.

```
aws greengrass list-connector-definition-versions \  
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

### Output:

```
{  
  "Versions": [  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-  
c7c2-4a26-a7e2-7bf478ea2623",  
      "CreationTimestamp": "2019-06-19T19:30:01.300Z",  
      "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",  
      "Version": "63c57963-c7c2-4a26-a7e2-7bf478ea2623"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan dengan Layanan dan Protokol Menggunakan Konektor Greengrass di Panduan Pengembang Greengrass IoT](#).AWS

- Untuk detail API, lihat [ListConnectorDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-connector-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-connector-definitions`.

### AWS CLI

Untuk membuat daftar konektor Greengrass yang didefinisikan

`list-connector-definitions` Contoh berikut mencantumkan semua konektor Greengrass yang ditentukan untuk akun Anda. AWS

```
aws greengrass list-connector-definitions
```

Output:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
      "CreationTimestamp": "2019-06-19T19:30:01.300Z",
      "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
      "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
      "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/
versions/63c57963-c7c2-4a26-a7e2-7bf478ea2623",
      "Name": "MySNSConnector"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan dengan Layanan dan Protokol Menggunakan Konektor Greengrass di Panduan Pengembang Greengrass IoT](#).AWS

- Untuk detail API, lihat [ListConnectorDefinitions](#) di Referensi AWS CLI Perintah.

## `list-core-definition-versions`

Contoh kode berikut menunjukkan cara menggunakan `list-core-definition-versions`.

AWS CLI

Untuk membuat daftar versi definisi inti Greengrass

`list-core-definitions` Contoh berikut mencantumkan semua versi definisi inti Greengrass yang ditentukan. Anda dapat menggunakan `list-core-definitions` perintah untuk mendapatkan ID versi.

```
aws greengrass list-core-definition-versions \
```



```
--core-definition-id "eaf280cb-138c-4d15-af36-6f681a1348f7"
```

Output:

```
{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7/versions/467c36e4-c5da-440c-
a97b-084e62593b4c",
      "CreationTimestamp": "2019-06-18T16:14:17.709Z",
      "Id": "eaf280cb-138c-4d15-af36-6f681a1348f7",
      "Version": "467c36e4-c5da-440c-a97b-084e62593b4c"
    }
  ]
}
```

- Untuk detail API, lihat [ListCoreDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-core-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-core-definitions`.

### AWS CLI

Untuk membuat daftar definisi Greengrass core

`list-core-definitions` Contoh berikut mencantumkan semua definisi inti Greengrass untuk akun Anda. AWS

```
aws greengrass list-core-definitions
```

Output:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/0507843c-c1ef-4f06-b051-817030df7e7d",
      "CreationTimestamp": "2018-10-17T04:30:32.786Z",
```

```

    "Id": "0507843c-c1ef-4f06-b051-817030df7e7d",
    "LastUpdatedTimestamp": "2018-10-17T04:30:32.786Z",
    "LatestVersion": "bcd9e86-3793-491e-93af-3cdfbf4e22b7",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/0507843c-c1ef-4f06-b051-817030df7e7d/versions/bcd9e86-3793-491e-93af-3cdfbf4e22b7"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/31c22500-3509-4271-bafd-cf0655cda438",
    "CreationTimestamp": "2019-06-18T16:24:16.064Z",
    "Id": "31c22500-3509-4271-bafd-cf0655cda438",
    "LastUpdatedTimestamp": "2019-06-18T16:24:16.064Z",
    "LatestVersion": "2f350395-6d09-4c8a-8336-9ae5b57ace84",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/31c22500-3509-4271-bafd-cf0655cda438/versions/2f350395-6d09-4c8a-8336-9ae5b57ace84"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46",
    "CreationTimestamp": "2019-06-18T16:21:21.351Z",
    "Id": "c906ed39-a1e3-4822-a981-7b9bd57b4b46",
    "LastUpdatedTimestamp": "2019-06-18T16:21:21.351Z",
    "LatestVersion": "42aeec3-fd9d-4312-a8fd-ffa9404a20e0",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeec3-fd9d-4312-a8fd-ffa9404a20e0"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7",
    "CreationTimestamp": "2019-06-18T16:14:17.709Z",
    "Id": "eaf280cb-138c-4d15-af36-6f681a1348f7",
    "LastUpdatedTimestamp": "2019-06-18T16:14:17.709Z",
    "LatestVersion": "467c36e4-c5da-440c-a97b-084e62593b4c",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7/versions/467c36e4-c5da-440c-a97b-084e62593b4c"
  }
]
}

```

- Untuk detail API, lihat [ListCoreDefinitions](#) di Referensi AWS CLI Perintah.

## list-deployments

Contoh kode berikut menunjukkan cara menggunakan `list-deployments`.

### AWS CLI

Untuk membuat daftar penerapan untuk grup Greengrass

`list-deployments` Contoh berikut mencantumkan penerapan untuk grup Greengrass yang ditentukan. Anda dapat menggunakan `list-groups` perintah untuk mencari ID grup Anda.

```
aws greengrass list-deployments \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

### Output:

```
{
  "Deployments": [
    {
      "CreatedAt": "2019-06-18T17:04:32.702Z",
      "DeploymentId": "1065b8a0-812b-4f21-9d5d-e89b232a530f",
      "DeploymentType": "NewDeployment",
      "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e"
    }
  ]
}
```

- Untuk detail API, lihat [ListDeployments](#) di Referensi AWS CLI Perintah.

## list-device-definition-versions

Contoh kode berikut menunjukkan cara menggunakan `list-device-definition-versions`.

### AWS CLI

Untuk membuat daftar versi definisi perangkat

`list-device-definition-versions` Contoh berikut menampilkan versi definisi perangkat yang terkait dengan definisi perangkat yang ditentukan.

```
aws greengrass list-device-definition-versions \  
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Output:

```
{  
  "Versions": [  
    {  
      "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",  
      "CreationTimestamp": "2019-09-11T00:15:09.838Z",  
      "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/  
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71"  
    },  
    {  
      "Version": "3b5cc510-58c1-44b5-9d98-4ad858ffa795",  
      "CreationTimestamp": "2019-09-11T00:11:06.197Z",  
      "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/  
versions/3b5cc510-58c1-44b5-9d98-4ad858ffa795"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListDeviceDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-device-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-device-definitions`.

### AWS CLI

Untuk mencantumkan definisi perangkat Anda

`list-device-definitions` Contoh berikut menampilkan detail tentang definisi perangkat di AWS akun Anda di AWS Wilayah yang ditentukan.

```
aws greengrass list-device-definitions \  
  --region us-west-2
```

## Output:

```
{
  "Definitions": [
    {
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/50f3274c-3f0a-4f57-b114-6f46085281ab/versions/c777b0f5-1059-449b-beaa-f003ebc56c34",
      "LastUpdatedTimestamp": "2019-06-14T15:42:09.059Z",
      "LatestVersion": "c777b0f5-1059-449b-beaa-f003ebc56c34",
      "CreationTimestamp": "2019-06-14T15:42:09.059Z",
      "Id": "50f3274c-3f0a-4f57-b114-6f46085281ab",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/50f3274c-3f0a-4f57-b114-6f46085281ab"
    },
    {
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/e01951c9-6134-479a-969a-1a15cac11c40/versions/514d57aa-4ee6-401c-9fac-938a9f7a51e5",
      "Name": "TestDeviceDefinition",
      "LastUpdatedTimestamp": "2019-04-16T23:17:43.245Z",
      "LatestVersion": "514d57aa-4ee6-401c-9fac-938a9f7a51e5",
      "CreationTimestamp": "2019-04-16T23:17:43.245Z",
      "Id": "e01951c9-6134-479a-969a-1a15cac11c40",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/e01951c9-6134-479a-969a-1a15cac11c40"
    },
    {
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
      "Name": "TemperatureSensors",
      "LastUpdatedTimestamp": "2019-09-10T00:19:03.698Z",
      "LatestVersion": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
      "CreationTimestamp": "2019-09-11T00:11:06.197Z",
      "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
    }
  ]
}
```

- Untuk detail API, lihat [ListDeviceDefinitions](#) di Referensi AWS CLI Perintah.

## list-function-definition-versions

Contoh kode berikut menunjukkan cara menggunakan `list-function-definition-versions`.

### AWS CLI

Untuk membuat daftar versi fungsi Lambda

`list-function-definition-versions` Contoh berikut mencantumkan semua versi fungsi Lambda yang ditentukan. Anda dapat menggunakan `list-function-definitions` perintah untuk mendapatkan ID.

```
aws greengrass list-function-definition-versions \
  --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85"
```

### Output:

```
{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
      "CreationTimestamp": "2019-06-18T17:04:30.776Z",
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "Version": "9748fda7-1589-4fcc-ac94-f5559e88678b"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/9b08df77-26f2-4c29-93d2-769715edcfec",
      "CreationTimestamp": "2019-06-18T17:02:44.087Z",
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "Version": "9b08df77-26f2-4c29-93d2-769715edcfec"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/4236239f-94f7-4b90-a2f8-2a24c829d21e",
      "CreationTimestamp": "2019-06-18T17:01:42.284Z",
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "Version": "4236239f-94f7-4b90-a2f8-2a24c829d21e"
    }
  ]
}
```

```

    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/343408bb-549a-4fbe-b043-853643179a39",
      "CreationTimestamp": "2019-06-18T16:21:21.431Z",
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "Version": "343408bb-549a-4fbe-b043-853643179a39"
    }
  ]
}

```

- Untuk detail API, lihat [ListFunctionDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-function-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-function-definitions`.

### AWS CLI

Untuk daftar fungsi Lambda

`list-function-definitions` Contoh berikut mencantumkan semua fungsi Lambda yang ditentukan untuk akun Anda AWS .

```
aws greengrass list-function-definitions
```

Output:

```

{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/017970a5-8952-46dd-b1c1-020b3ae8e960",
      "CreationTimestamp": "2018-10-17T04:30:32.884Z",
      "Id": "017970a5-8952-46dd-b1c1-020b3ae8e960",
      "LastUpdatedTimestamp": "2018-10-17T04:30:32.884Z",
      "LatestVersion": "4380b302-790d-4ed8-92bf-02e88afecb15",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/017970a5-8952-46dd-b1c1-020b3ae8e960/versions/4380b302-790d-4ed8-92bf-02e88afecb15"
    }
  ]
}

```

```

    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "CreationTimestamp": "2019-06-18T16:21:21.431Z",
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "LastUpdatedTimestamp": "2019-06-18T16:21:21.431Z",
      "LatestVersion": "9748fda7-1589-4fcc-ac94-f5559e88678b",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/6598e653-a262-440c-9967-e2697f64da7b",
      "CreationTimestamp": "2019-06-18T16:24:16.123Z",
      "Id": "6598e653-a262-440c-9967-e2697f64da7b",
      "LastUpdatedTimestamp": "2019-06-18T16:24:16.123Z",
      "LatestVersion": "38bc6ccd-98a2-4ce7-997e-16c84748fae4",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/6598e653-a262-440c-9967-e2697f64da7b/
versions/38bc6ccd-98a2-4ce7-997e-16c84748fae4"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/c668df84-fad2-491b-95f4-655d2cad7885",
      "CreationTimestamp": "2019-06-18T16:14:17.784Z",
      "Id": "c668df84-fad2-491b-95f4-655d2cad7885",
      "LastUpdatedTimestamp": "2019-06-18T16:14:17.784Z",
      "LatestVersion": "37dd68c4-a64f-40ba-aa13-71fecc3ebded",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/c668df84-fad2-491b-95f4-655d2cad7885/
versions/37dd68c4-a64f-40ba-aa13-71fecc3ebded"
    }
  ]
}

```

- Untuk detail API, lihat [ListFunctionDefinitions](#) di Referensi AWS CLI Perintah.

## list-group-certificate-authorities

Contoh kode berikut menunjukkan cara menggunakan `list-group-certificate-authorities`.



## AWS CLI

Untuk membuat daftar CA saat ini untuk grup

`list-group-certificate-authorities` Contoh berikut mencantumkan otoritas sertifikat (CA) saat ini untuk grup Greengrass yang ditentukan.

```
aws greengrass list-group-certificate-authorities \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{  
  "GroupCertificateAuthorities": [  
    {  
      "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-  
west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/  
certificateauthorities/  
f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",  
      "GroupCertificateAuthorityId":  
        "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListGroupCertificateAuthorities](#) di Referensi AWS CLI Perintah.

## **list-group-versions**

Contoh kode berikut menunjukkan cara menggunakan `list-group-versions`.

### AWS CLI

Untuk membuat daftar versi grup Greengrass

`list-group-versions` Contoh berikut mencantumkan versi grup Greengrass yang ditentukan.

```
aws greengrass list-group-versions \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
      "CreationTimestamp": "2019-06-18T17:04:30.915Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "Version": "115136b3-cfd7-4462-b77f-8741a4b00e5e"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/4340669d-
d14d-44e3-920c-46c928750750",
      "CreationTimestamp": "2019-06-18T17:03:52.663Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "Version": "4340669d-d14d-44e3-920c-46c928750750"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/
versions/1b06e099-2d5b-4f10-91b9-78c4e060f5da",
      "CreationTimestamp": "2019-06-18T17:02:44.189Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "Version": "1b06e099-2d5b-4f10-91b9-78c4e060f5da"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/2d3f27f1-3b43-4554-
ab7a-73ec30477efe",
      "CreationTimestamp": "2019-06-18T17:01:42.401Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "Version": "2d3f27f1-3b43-4554-ab7a-73ec30477efe"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/d20f7ae9-3444-4c1c-b025-
e2ede23cdd31",
      "CreationTimestamp": "2019-06-18T16:21:21.457Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "Version": "d20f7ae9-3444-4c1c-b025-e2ede23cdd31"
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [ListGroupVersions](#) di Referensi AWS CLI Perintah.

## list-groups

Contoh kode berikut menunjukkan cara menggunakan `list-groups`.

### AWS CLI

Untuk membuat daftar kelompok Greengrass

`list-groups` Contoh berikut mencantumkan semua grup Greengrass yang didefinisikan di akun Anda. AWS

```
aws greengrass list-groups
```

Output:

```
{
  "Groups": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731",
      "CreationTimestamp": "2019-06-18T16:21:21.457Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "LastUpdatedTimestamp": "2019-06-18T16:21:21.457Z",
      "LatestVersion": "115136b3-cfd7-4462-b77f-8741a4b00e5e",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-b77f-8741a4b00e5e",
      "Name": "GGGroup4Pi3"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8",
      "CreationTimestamp": "2018-10-31T21:52:46.603Z",
      "Id": "1402daf9-71cf-4cfe-8be0-d5e80526d0d8",
      "LastUpdatedTimestamp": "2018-10-31T21:52:46.603Z",
      "LatestVersion": "749af901-60ab-456f-a096-91b12d983c29",
    }
  ]
}
```

```

    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8/versions/749af901-60ab-456f-
a096-91b12d983c29",
    "Name": "MyTestGroup"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/504b5c8d-bbed-4635-aff1-48ec5b586db5",
    "CreationTimestamp": "2018-12-31T21:39:36.771Z",
    "Id": "504b5c8d-bbed-4635-aff1-48ec5b586db5",
    "LastUpdatedTimestamp": "2018-12-31T21:39:36.771Z",
    "LatestVersion": "46911e8e-f9bc-4898-8b63-59c7653636ec",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/504b5c8d-bbed-4635-aff1-48ec5b586db5/versions/46911e8e-
f9bc-4898-8b63-59c7653636ec",
    "Name": "smp-ggrass-group"
  }
]
}

```

- Untuk detail API, lihat [ListGroups](#) di Referensi AWS CLI Perintah.

## list-logger-definition-versions

Contoh kode berikut menunjukkan cara menggunakan `list-logger-definition-versions`.

### AWS CLI

Untuk mendapatkan daftar versi definisi logger

`list-logger-definition-versions` Contoh berikut mendapatkan daftar semua versi definisi logger yang ditentukan.

```
aws greengrass list-logger-definition-versions \
  --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23"
```

Output:

```
{
  "Versions": [
    {
```

```

        "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
        "CreationTimestamp": "2019-05-08T16:10:13.866Z",
        "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
        "Version": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
    },
    {
        "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/3ec6d3af-eb85-48f9-
a16d-1c795fe696d7",
        "CreationTimestamp": "2019-05-08T16:10:13.809Z",
        "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
        "Version": "3ec6d3af-eb85-48f9-a16d-1c795fe696d7"
    }
]
}

```

- Untuk detail API, lihat [ListLoggerDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-logger-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-logger-definitions`.

### AWS CLI

Untuk mendapatkan daftar definisi logger

`list-logger-definitions` Contoh berikut mencantumkan semua definisi logger untuk AWS akun Anda.

```
aws greengrass list-logger-definitions
```

Output:

```

{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23",
      "CreationTimestamp": "2019-05-08T16:10:13.809Z",

```

```

        "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
        "LastUpdatedTimestamp": "2019-05-08T16:10:13.809Z",
        "LatestVersion": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073",
        "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/
versions/5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
    }
]
}

```

- Untuk detail API, lihat [ListLoggerDefinitions](#) di Referensi AWS CLI Perintah.

## list-resource-definition-versions

Contoh kode berikut menunjukkan cara menggunakan `list-resource-definition-versions`.

### AWS CLI

Untuk membuat daftar versi definisi sumber daya

`list-resource-definition-versions` Contoh berikut mencantumkan versi untuk sumber Greengrass yang ditentukan.

```

aws greengrass list-resource-definition-versions \
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"

```

Output:

```

{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
      "CreationTimestamp": "2019-06-19T16:40:59.392Z",
      "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
      "Version": "26e8829a-491a-464d-9c87-664bf6f6f2be"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/432d92f6-12de-4ec9-a704-619a942a62aa",

```

```

    "CreationTimestamp": "2019-06-19T16:40:59.261Z",
    "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
    "Version": "432d92f6-12de-4ec9-a704-619a942a62aa"
  }
]
}

```

- Untuk detail API, lihat [ListResourceDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-resource-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-resource-definitions`.

### AWS CLI

Untuk membuat daftar sumber daya yang didefinisikan

`list-resource-definitions` Contoh berikut mencantumkan sumber daya yang didefinisikan untuk AWS IoT Greengrass untuk digunakan.

```
aws greengrass list-resource-definitions
```

### Output:

```

{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658",
      "CreationTimestamp": "2019-06-19T16:40:59.261Z",
      "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
      "LastUpdatedTimestamp": "2019-06-19T16:40:59.261Z",
      "LatestVersion": "26e8829a-491a-464d-9c87-664bf6f6f2be",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/versions/26e8829a-491a-464d-9c87-664bf6f6f2be"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
      "CreationTimestamp": "2019-06-19T21:51:28.212Z",

```

```

        "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
        "LastUpdatedTimestamp": "2019-06-19T21:51:28.212Z",
        "LatestVersion": "a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
        "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/
a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
        "Name": "MyGreengrassResources"
    }
]
}

```

- Untuk detail API, lihat [ListResourceDefinitions](#) di Referensi AWS CLI Perintah.

## list-subscription-definition-versions

Contoh kode berikut menunjukkan cara menggunakan `list-subscription-definition-versions`.

### AWS CLI

Untuk membuat daftar versi definisi langganan

`list-subscription-definition-versions` Contoh berikut mencantumkan semua versi langganan yang ditentukan. Anda dapat menggunakan `list-subscription-definitions` perintah untuk mencari ID langganan.

```

aws greengrass list-subscription-definition-versions \
  --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152"

```

Output:

```

{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b",
      "CreationTimestamp": "2019-06-18T17:03:52.499Z",
      "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
      "Version": "88ae8699-12ac-4663-ba3f-4d7f0519140b"
    },

```



```

    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/7e320ba3-
c369-4069-a2f0-90acb7f219d6",
      "CreationTimestamp": "2019-06-18T17:03:52.392Z",
      "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
      "Version": "7e320ba3-c369-4069-a2f0-90acb7f219d6"
    }
  ]
}

```

- Untuk detail API, lihat [ListSubscriptionDefinitionVersions](#) di Referensi AWS CLI Perintah.

## list-subscription-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-subscription-definitions`.

### AWS CLI

Untuk mendapatkan daftar definisi langganan

`list-subscription-definitions` Contoh berikut mencantumkan semua langganan AWS Greengrass IoT yang ditentukan di akun Anda. AWS

```
aws greengrass list-subscription-definitions
```

Output:

```

{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152",
      "CreationTimestamp": "2019-06-18T17:03:52.392Z",
      "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
      "LastUpdatedTimestamp": "2019-06-18T17:03:52.392Z",
      "LatestVersion": "88ae8699-12ac-4663-ba3f-4d7f0519140b",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b"
    },
    {

```

```

    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/cd6f1c37-d9a4-4e90-be94-01a7404f5967",
    "CreationTimestamp": "2018-10-18T15:45:34.024Z",
    "Id": "cd6f1c37-d9a4-4e90-be94-01a7404f5967",
    "LastUpdatedTimestamp": "2018-10-18T15:45:34.024Z",
    "LatestVersion": "d1cf8fac-284f-4f6a-98fe-a2d36d089373",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/cd6f1c37-d9a4-4e90-be94-01a7404f5967/versions/
d1cf8fac-284f-4f6a-98fe-a2d36d089373"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/fa81bc84-3f59-4377-a84b-5d0134da359b",
    "CreationTimestamp": "2018-10-22T17:09:31.429Z",
    "Id": "fa81bc84-3f59-4377-a84b-5d0134da359b",
    "LastUpdatedTimestamp": "2018-10-22T17:09:31.429Z",
    "LatestVersion": "086d1b08-b25a-477c-a16f-6f9b3a9c295a",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/fa81bc84-3f59-4377-a84b-5d0134da359b/
versions/086d1b08-b25a-477c-a16f-6f9b3a9c295a"
  }
]
}

```

- Untuk detail API, lihat [ListSubscriptionDefinitions](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag dan nilainya yang dilampirkan ke sumber daya yang ditentukan.

```

aws greengrass list-tags-for-resource \
  --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658"

```

Output:

```
{
  "tags": {
    "ResourceSubType": "USB",
    "ResourceType": "Device"
  }
}
```

Untuk informasi lebih lanjut, lihat [Menandai Sumber Daya Greengrass Anda di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## reset-deployments

Contoh kode berikut menunjukkan cara menggunakan `reset-deployments`.

### AWS CLI

Untuk membersihkan informasi penyebaran untuk grup Greengrass

`reset-deployments` Contoh berikut membersihkan informasi penyebaran untuk grup Greengrass yang ditentukan. Saat Anda menambahkan `--force` option, informasi penyebaran diatur ulang tanpa menunggu perangkat inti merespons.

```
aws greengrass reset-deployments \
  --group-id "1402daf9-71cf-4cfe-8be0-d5e80526d0d8" \
  --force
```

Output:

```
{
  "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8/
deployments/7dd4e356-9882-46a3-9e28-6d21900c011a",
  "DeploymentId": "7dd4e356-9882-46a3-9e28-6d21900c011a"
}
```

Untuk informasi selengkapnya, lihat [Mengatur Ulang Penerapan di Panduan Pengembang AWS Greengrass IoT](#).

- Untuk detail API, lihat [ResetDeployments](#) di Referensi AWS CLI Perintah.

## start-bulk-deployment

Contoh kode berikut menunjukkan cara menggunakan `start-bulk-deployment`.

### AWS CLI

Untuk memulai operasi penyebaran massal

`start-bulk-deployment` Contoh berikut memulai operasi penyebaran massal, menggunakan file yang disimpan dalam bucket S3 untuk menentukan grup yang akan digunakan.

```
aws greengrass start-bulk-deployment \
  --cli-input-json "{\"InputFileUri\":\"https://gg-group-deployment1.s3-us-
west-2.amazonaws.com/MyBulkDeploymentInputFile.txt\", \"ExecutionRoleArn\":
\"arn:aws:iam::123456789012:role/ggCreateDeploymentRole\", \"AmznClientToken\":
\"yourAmazonClientToken\"}"
```

Output:

```
{
  "BulkDeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
bulk/deployments/870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
  "BulkDeploymentId": "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
}
```

Untuk informasi selengkapnya, lihat [Membuat Penerapan Massal untuk Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [StartBulkDeployment](#) di Referensi AWS CLI Perintah.

## stop-bulk-deployment

Contoh kode berikut menunjukkan cara menggunakan `stop-bulk-deployment`.

### AWS CLI

Untuk menghentikan penyebaran massal

`stop-bulk-deployment` Contoh berikut menghentikan penyebaran massal yang ditentukan. Jika Anda mencoba menghentikan penyebaran massal yang selesai, Anda menerima kesalahan: `InvalidInputException: Cannot change state of finished execution.`

```
aws greengrass stop-bulk-deployment \  
  --bulk-deployment-id "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Penerapan Massal untuk Grup](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [StopBulkDeployment](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menerapkan tag ke sumber daya

`tag-resource` Contoh berikut berlaku dua tag, `ResourceType` dan `ResourceSubType`, ke sumber Greengrass yang ditentukan. Operasi ini dapat menambahkan tag dan nilai baru atau memperbarui nilai untuk tag yang ada. Gunakan `untag-resource` perintah untuk menghapus tag.

```
aws greengrass tag-resource \  
  --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658" \  
  --tags "ResourceType=Device,ResourceSubType=USB"
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menandai Sumber Daya Greengrass Anda di Panduan Pengembang Greengrass](#) IoT.AWS

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dan nilainya dari sumber daya

`untag-resource` Contoh berikut menghapus tag yang kuncinya `Category` dari grup Greengrass yang ditentukan. Jika kunci `Category` tidak ada untuk sumber daya yang ditentukan, tidak ada kesalahan yang dikembalikan.

```
aws greengrass untag-resource \  
  --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/1013db12-8b58-45ff-acc7-704248f66731" \  
  --tag-keys "Category"
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menandai Sumber Daya Greengrass Anda di Panduan Pengembang Greengrass IoT.AWS](#)

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-connectivity-info

Contoh kode berikut menunjukkan cara menggunakan `update-connectivity-info`.

### AWS CLI

Untuk memperbarui informasi konektivitas untuk inti Greengrass

`update-connectivity-info` Contoh berikut mengubah titik akhir yang dapat digunakan perangkat untuk terhubung ke inti Greengrass yang ditentukan. Informasi konektivitas adalah daftar alamat IP atau nama domain, dengan nomor port yang sesuai dan metadata yang ditentukan pelanggan opsional. Anda mungkin perlu memperbarui informasi konektivitas saat jaringan lokal berubah.

```
aws greengrass update-connectivity-info \  
  --thing-name "MyGroup_Core" \  
  --connectivity-info "[{\\"Metadata\\":\\"\\",\\"PortNumber\\":8883,\\"HostAddress\\":  
\\"127.0.0.1\\",\\"Id\\":\\"localhost_127.0.0.1_0\\"},{\\"Metadata\\":\\"\\",\\"PortNumber  
\\":8883,\\"HostAddress\\":\\"192.168.1.3\\",\\"Id\\":\\"localIP_192.168.1.3\\"}]"
```

Output:

```
{  
  "Version": "312de337-59af-4cf9-a278-2a23bd39c300"
```

```
}
```

- Untuk detail API, lihat [UpdateConnectivityInfo](#) di Referensi AWS CLI Perintah.

## update-connector-definition

Contoh kode berikut menunjukkan cara menggunakan `update-connector-definition`.

### AWS CLI

Untuk memperbarui nama untuk definisi konektor

`update-connector-definition` Contoh berikut memperbarui nama untuk definisi konektor yang ditentukan. Jika Anda ingin memperbarui detail untuk konektor, gunakan `create-connector-definition-version` perintah untuk membuat versi baru.

```
aws greengrass update-connector-definition \  
  --connector-definition-id "55d0052b-0d7d-44d6-b56f-21867215e118" \  
  --name "GreengrassConnectors2019"
```

Untuk informasi selengkapnya, lihat [Mengintegrasikan dengan Layanan dan Protokol Menggunakan Konektor](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [UpdateConnectorDefinition](#) di Referensi AWS CLI Perintah.

## update-core-definition

Contoh kode berikut menunjukkan cara menggunakan `update-core-definition`.

### AWS CLI

Untuk memperbarui definisi inti

`update-core-definition` Contoh berikut mengubah nama definisi inti yang ditentukan. Anda hanya dapat memperbarui name properti definisi inti.

```
aws greengrass update-core-definition \  
  --core-definition-id "582efe12-b05a-409e-9a24-a2ba1bcc4a12" \  
  --name "MyCoreDevices"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi Inti AWS Greengrass IoT di Panduan Pengembang IoT Greengrass](#).AWS

- Untuk detail API, lihat [UpdateCoreDefinition](#) di Referensi AWS CLI Perintah.

## update-device-definition

Contoh kode berikut menunjukkan cara menggunakan `update-device-definition`.

### AWS CLI

Untuk memperbarui definisi perangkat

`update-device-definition` Contoh berikut mengubah nama definisi perangkat yang ditentukan. Anda hanya dapat memperbarui name properti definisi perangkat.

```
aws greengrass update-device-definition \  
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \  
  --name "TemperatureSensors"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateDeviceDefinition](#) di Referensi AWS CLI Perintah.

## update-function-definition

Contoh kode berikut menunjukkan cara menggunakan `update-function-definition`.

### AWS CLI

Untuk memperbarui nama untuk definisi fungsi

`update-function-definition` Contoh berikut memperbarui nama untuk definisi fungsi yang ditentukan. Jika Anda ingin memperbarui detail untuk fungsi tersebut, gunakan `create-function-definition-version` perintah untuk membuat versi baru.

```
aws greengrass update-function-definition \  
  --function-definition-id "e47952bd-dea9-4e2c-a7e1-37bbe8807f46" \  
  --name ObsoleteFunction
```

Perintah ini tidak menghasilkan output.



Untuk informasi selengkapnya, lihat [Menjalankan Fungsi Lambda Lokal](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [UpdateFunctionDefinition](#) di Referensi AWS CLI Perintah.

## update-group-certificate-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-group-certificate-configuration`.

### AWS CLI

Untuk memperbarui berakhirnya sertifikat grup

`update-group-certificate-configuration` Contoh berikut menetapkan kadaluwarsa 10 hari untuk sertifikat yang dihasilkan untuk grup yang ditentukan.

```
aws greengrass update-group-certificate-configuration \  
  --group-id "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1" \  
  --certificate-expiry-in-milliseconds 864000000
```

Output:

```
{  
  "CertificateExpiryInMilliseconds": 864000000,  
  "CertificateAuthorityExpiryInMilliseconds": 2524607999000,  
  "GroupId": "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1"  
}
```

Untuk informasi selengkapnya, lihat [Keamanan AWS Greengrass IoT di Panduan Pengembang Greengrass IoT](#).AWS

- Untuk detail API, lihat [UpdateGroupCertificateConfiguration](#) di Referensi AWS CLI Perintah.

## update-group

Contoh kode berikut menunjukkan cara menggunakan `update-group`.

### AWS CLI

Untuk memperbarui nama grup

`update-group` Contoh berikut memperbarui nama grup Greengrass yang ditentukan. Jika Anda ingin memperbarui detail untuk grup, gunakan `create-group-version` perintah untuk membuat versi baru.

```
aws greengrass update-group \  
  --group-id "1402daf9-71cf-4cfe-8be0-d5e80526d0d8" \  
  --name TestGroup4of6
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi AWS Greengrass IoT di IoT di Panduan Pengembang AWS IoT](#).AWS

- Untuk detail API, lihat [UpdateGroup](#) di Referensi AWS CLI Perintah.

## **update-logger-definition**

Contoh kode berikut menunjukkan cara menggunakan `update-logger-definition`.

### AWS CLI

Untuk memperbarui definisi logger

`update-logger-definition` Contoh berikut mengubah nama definisi logger yang ditentukan. Anda hanya dapat memperbarui name properti definisi logger.

```
aws greengrass update-logger-definition \  
  --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0" \  
  --name "LoggingConfigsForSensors"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pemantauan dengan Log AWS Greengrass IoT di Panduan Pengembang Greengrass IoT](#).AWS

- Untuk detail API, lihat [UpdateLoggerDefinition](#) di Referensi AWS CLI Perintah.

## **update-resource-definition**

Contoh kode berikut menunjukkan cara menggunakan `update-resource-definition`.

### AWS CLI

Untuk memperbarui nama untuk definisi sumber daya

`update-resource-definition` Contoh berikut memperbarui nama untuk definisi sumber daya yang ditentukan. Jika Anda ingin mengubah detail untuk sumber daya, gunakan `create-resource-definition-version` perintah untuk membuat versi baru.

```
aws greengrass update-resource-definition \  
  --resource-definition-id "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38" \  
  --name GreengrassConnectorResources
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengakses Sumber Daya Lokal dengan Fungsi dan Konektor Lambda](#) di Panduan Pengembang AWS Greengrass IoT.

- Untuk detail API, lihat [UpdateResourceDefinition](#) di Referensi AWS CLI Perintah.

## update-subscription-definition

Contoh kode berikut menunjukkan cara menggunakan `update-subscription-definition`.

### AWS CLI

Untuk memperbarui nama untuk definisi langganan

`update-subscription-definition` Contoh berikut memperbarui nama untuk definisi langganan yang ditentukan. Jika Anda ingin mengubah detail langganan, gunakan `create-subscription-definition-version` perintah untuk membuat versi baru.

```
aws greengrass update-subscription-definition \  
  --subscription-definition-id "fa81bc84-3f59-4377-a84b-5d0134da359b" \  
  --name "ObsoleteSubscription"
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat judul di panduan.

- Untuk detail API, lihat [UpdateSubscriptionDefinition](#) di Referensi AWS CLI Perintah.

## update-thing-runtime-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-thing-runtime-configuration`.

## AWS CLI

Untuk mengaktifkan telemetri dalam konfigurasi runtime inti Greengrass

`update-thing-runtime-configuration` Contoh berikut memperbarui konfigurasi runtime inti Greengrass untuk mengaktifkan telemetri.

```
aws greengrass update-thing-runtime-configuration \  
  --thing-name SampleGreengrassCore \  
  --telemetry-configuration {"Telemetry\":"0n\"}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi setelan telemetri di Panduan Pengembang Greengrass AWS IoT](#).

- Untuk detail API, lihat [UpdateThingRuntimeConfiguration](#) di Referensi AWS CLI Perintah.

## AWS IoT Greengrass V2 contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Greengrass V2.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-service-role-to-account**

Contoh kode berikut menunjukkan cara menggunakan `associate-service-role-to-account`.

## AWS CLI

Untuk mengaitkan peran layanan Greengrass ke akun Anda AWS

`associate-service-role-to-account` Contoh berikut mengaitkan peran layanan dengan AWS IoT Greengrass untuk akun Anda. AWS

```
aws greengrassv2 associate-service-role-to-account \  
  --role-arn arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole
```

Output:

```
{  
  "associatedAt": "2022-01-19T19:21:53Z"  
}
```

Untuk informasi selengkapnya, lihat Peran [layanan Greengrass di Panduan Pengembang Greengrass V2 IoT AWS](#) .

- Untuk detail API, lihat [AssociateServiceRoleToAccount](#) di Referensi AWS CLI Perintah.

## **batch-associate-client-device-with-core-device**

Contoh kode berikut menunjukkan cara menggunakan `batch-associate-client-device-with-core-device`.

## AWS CLI

Untuk mengaitkan perangkat klien dengan perangkat inti

`batch-associate-client-device-with-core-device` Contoh berikut mengaitkan dua perangkat klien dengan perangkat inti.

```
aws greengrassv2 batch-associate-client-device-with-core-device \  
  --core-device-thing-name MyGreengrassCore \  
  --entries thingName=MyClientDevice1 thingName=MyClientDevice2
```

Output:

```
{
```

```
"errorEntries": []  
}
```

Untuk informasi selengkapnya, lihat [Berinteraksi dengan perangkat IoT lokal di Panduan Pengembang AWS Greengrass V2](#).

- Untuk detail API, lihat [BatchAssociateClientDeviceWithCoreDevice](#) di Referensi AWS CLI Perintah.

## **batch-disassociate-client-device-from-core-device**

Contoh kode berikut menunjukkan cara menggunakan `batch-disassociate-client-device-from-core-device`.

### AWS CLI

Untuk memisahkan perangkat klien dari perangkat inti

`batch-disassociate-client-device-from-core-device` Contoh berikut memisahkan dua perangkat klien dari perangkat inti.

```
aws greengrassv2 batch-disassociate-client-device-from-core-device \  
  --core-device-thing-name MyGreengrassCore \  
  --entries thingName=MyClientDevice1 thingName=MyClientDevice2
```

Output:

```
{  
  "errorEntries": []  
}
```

Untuk informasi selengkapnya, lihat [Berinteraksi dengan perangkat IoT lokal di Panduan Pengembang AWS Greengrass V2](#).

- Untuk detail API, lihat [BatchDisassociateClientDeviceFromCoreDevice](#) di Referensi AWS CLI Perintah.

## **cancel-deployment**

Contoh kode berikut menunjukkan cara menggunakan `cancel-deployment`.

## AWS CLI

Untuk membatalkan penerapan

`cancel-deployment` Contoh berikut menghentikan penerapan berkelanjutan ke grup sesuatu.

```
aws greengrassv2 cancel-deployment \  
  --deployment-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "message": "SUCCESS"  
}
```

Untuk informasi selengkapnya, lihat [Batalkan penerapan](#) di Panduan Pengembang AWS Greengrass V2 IoT.

- Untuk detail API, lihat [CancelDeployment](#) di Referensi AWS CLI Perintah.

## **create-component-version**

Contoh kode berikut menunjukkan cara menggunakan `create-component-version`.

### AWS CLI

Contoh 1: Untuk membuat versi komponen dari resep

`create-component-version` Contoh berikut membuat versi komponen Hello World dari file resep.

```
aws greengrassv2 create-component-version \  
  --inline-recipe fileb://com.example>HelloWorld-1.0.0.json
```

Isi dari `com.example>HelloWorld-1.0.0.json`:

```
{  
  "RecipeFormatVersion": "2020-01-25",  
  "ComponentName": "com.example>HelloWorld",  
  "ComponentVersion": "1.0.0",  
  "ComponentDescription": "My first AWS IoT Greengrass component.",  
  "ComponentPublisher": "Amazon",
```

```

"ComponentConfiguration": {
  "DefaultConfiguration": {
    "Message": "world"
  }
},
"Manifests": [
  {
    "Platform": {
      "os": "linux"
    },
    "Lifecycle": {
      "Run": "echo 'Hello {configuration:/Message}'"
    }
  }
]
}

```

### Output:

```

{
  "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",
  "componentName": "com.example.HelloWorld",
  "componentVersion": "1.0.0",
  "creationTimestamp": "2021-01-07T16:24:33.650000-08:00",
  "status": {
    "componentState": "REQUESTED",
    "message": "NONE",
    "errors": {}
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat komponen khusus](#) dan [Mengunggah komponen yang akan diterapkan](#) di Panduan Pengembang Greengrass V2 AWS IoT.

Contoh 2: Untuk membuat versi komponen dari fungsi AWS Lambda

create-component-version Contoh berikut membuat versi komponen Hello World dari fungsi AWS Lambda.

```

aws greengrassv2 create-component-version \
  --cli-input-json file://lambda-function-component.json

```



Isi dari `lambda-function-component.json`:

```
{
  "lambdaFunction": {
    "lambdaArn": "arn:aws:lambda:us-
west-2:123456789012:function:HelloWorldPythonLambda:1",
    "componentName": "com.example.HelloWorld",
    "componentVersion": "1.0.0",
    "componentLambdaParameters": {
      "eventSources": [
        {
          "topic": "hello/world/+",
          "type": "IOT_CORE"
        }
      ]
    }
  }
}
```

Output:

```
{
  "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",
  "componentName": "com.example.HelloWorld",
  "componentVersion": "1.0.0",
  "creationTimestamp": "2021-01-07T17:05:27.347000-08:00",
  "status": {
    "componentState": "REQUESTED",
    "message": "NONE",
    "errors": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Menjalankan fungsi AWS Lambda](#) di Panduan Pengembang AWS Greengrass V2 IoT.

- Untuk detail API, lihat [CreateComponentVersion](#) di Referensi AWS CLI Perintah.

## create-deployment

Contoh kode berikut menunjukkan cara menggunakan `create-deployment`.

## AWS CLI

Contoh 1: Untuk membuat penerapan

`create-deployment` Contoh berikut menyebarkan AWS IoT Greengrass Command Line Interface ke perangkat inti.

```
aws greengrassv2 create-deployment \  
  --cli-input-json file://cli-deployment.json
```

Isi dari `cli-deployment.json`:

```
{  
  "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/MyGreengrassCore",  
  "deploymentName": "Deployment for MyGreengrassCore",  
  "components": {  
    "aws.greengrass.Cli": {  
      "componentVersion": "2.0.3"  
    }  
  },  
  "deploymentPolicies": {  
    "failureHandlingPolicy": "DO_NOTHING",  
    "componentUpdatePolicy": {  
      "timeoutInSeconds": 60,  
      "action": "NOTIFY_COMPONENTS"  
    },  
    "configurationValidationPolicy": {  
      "timeoutInSeconds": 60  
    }  
  },  
  "iotJobConfiguration": {}  
}
```

Output:

```
{  
  "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Untuk informasi selengkapnya, lihat [Membuat penerapan](#) di Panduan Pengembang IoT AWS Greengrass V2.

## Contoh 2: Untuk membuat penerapan yang memperbarui konfigurasi komponen

`create-deployment` Contoh berikut menyebarkan komponen inti Greengrass AWS IoT ke sekelompok perangkat inti. Penerapan ini menerapkan pembaruan konfigurasi berikut untuk komponen inti:

Setel ulang pengaturan proxy perangkat target ke defaultnya tanpa pengaturan proxy. Setel ulang pengaturan MQTT perangkat target ke defaultnya. Menetapkan opsi JVM untuk JVM. set level logging untuk nukleus.

```
aws greengrassv2 create-deployment \  
  --cli-input-json file://nucleus-deployment.json
```

Isi dari `nucleus-deployment.json`:

```
{  
  "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/  
MyGreengrassCoreGroup",  
  "deploymentName": "Deployment for MyGreengrassCoreGroup",  
  "components": {  
    "aws.greengrass.Nucleus": {  
      "componentVersion": "2.0.3",  
      "configurationUpdate": {  
        "reset": [  
          "/networkProxy",  
          "/mqtt"  
        ],  
        "merge": "{\"jvmOptions\":\"-Xmx64m\",\"logging\":{\"level\":\"WARN  
\"}}"  
      }  
    }  
  },  
  "deploymentPolicies": {  
    "failureHandlingPolicy": "ROLLBACK",  
    "componentUpdatePolicy": {  
      "timeoutInSeconds": 60,  
      "action": "NOTIFY_COMPONENTS"  
    },  
    "configurationValidationPolicy": {  
      "timeoutInSeconds": 60  
    }  
  },  
}
```

```
"iotJobConfiguration": {}  
}
```

Output:

```
{  
  "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE22222"  
}
```

Untuk informasi selengkapnya, lihat [Membuat penerapan](#) dan [Memperbarui konfigurasi komponen](#) di Panduan Pengembang Greengrass AWS V2 IoT.

- Untuk detail API, lihat [CreateDeployment](#) di Referensi AWS CLI Perintah.

## delete-component

Contoh kode berikut menunjukkan cara menggunakan `delete-component`.

### AWS CLI

Untuk menghapus versi komponen

`delete-component` Contoh berikut menghapus komponen Hello World.

```
aws greengrassv2 delete-component \  
  --arn arn:aws:greengrass:us-  
west-2:123456789012:components:com.example>HelloWorld:versions:1.0.0
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [DeleteComponent](#) di Referensi AWS CLI Perintah.

## delete-core-device

Contoh kode berikut menunjukkan cara menggunakan `delete-core-device`.

## AWS CLI

Untuk menghapus perangkat inti

`delete-core-device` Contoh berikut menghapus perangkat inti AWS Greengrass IoT.

```
aws greengrassv2 delete-core-device \  
  --core-device-thing-name MyGreengrassCore
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Menghapus instalasi perangkat lunak AWS Greengrass Core AWS IoT di Panduan Pengembang IoT Greengrass V2](#).

- Untuk detail API, lihat [DeleteCoreDevice](#) di Referensi AWS CLI Perintah.

## `describe-component`

Contoh kode berikut menunjukkan cara menggunakan `describe-component`.

## AWS CLI

Untuk mendeskripsikan versi komponen

`describe-component` Contoh berikut menjelaskan komponen Hello World.

```
aws greengrassv2 describe-component \  
  --arn arn:aws:greengrass:us-  
west-2:123456789012:components:com.example>HelloWorld:versions:1.0.0
```

Output:

```
{  
  "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example>HelloWorld:versions:1.0.0",  
  "componentName": "com.example>HelloWorld",  
  "componentVersion": "1.0.0",  
  "creationTimestamp": "2021-01-07T17:12:11.133000-08:00",  
  "publisher": "Amazon",  
  "description": "My first AWS IoT Greengrass component.",  
  "status": {
```

```
    "componentState": "DEPLOYABLE",
    "message": "NONE",
    "errors": {}
  },
  "platforms": [
    {
      "attributes": {
        "os": "linux"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [DescribeComponent](#) di Referensi AWS CLI Perintah.

## **disassociate-service-role-from-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-service-role-from-account`.

### AWS CLI

Untuk memisahkan peran layanan Greengrass dari akun Anda AWS

`disassociate-service-role-from-account` Contoh berikut memisahkan peran layanan Greengrass dari AWS IoT Greengrass untuk akun Anda. AWS

```
aws greengrassv2 disassociate-service-role-from-account
```

Output:

```
{
  "disassociatedAt": "2022-01-19T19:26:09Z"
}
```

Untuk informasi selengkapnya, lihat Peran [layanan Greengrass di Panduan Pengembang Greengrass V2 IoT AWS](#) .

- Untuk detail API, lihat [DisassociateServiceRoleFromAccount](#) di Referensi AWS CLI Perintah.

## get-component-version-artifact

Contoh kode berikut menunjukkan cara menggunakan `get-component-version-artifact`.

### AWS CLI

Untuk mendapatkan URL untuk mengunduh artefak komponen

`get-component-version-artifact` Contoh berikut mendapatkan URL untuk mengunduh file JAR komponen konsol debug lokal.

```
aws greengrassv2 get-component-version-artifact \
  --arn arn:aws:greengrass:us-
west-2:aws:components:aws.greengrass.LocalDebugConsole:versions:2.0.3 \
  --artifact-name "Uvt6ZEzQ9TKiAuLbfXBX_APdY0TWks3uc46tHFHTzBM=/
aws.greengrass.LocalDebugConsole.jar"
```

Output:

```
{
  "preSignedUrl": "https://evergreencomponentmanageme-
artifactbucket7410c9ef-g18n1iya8kwr.s3.us-west-2.amazonaws.com/public/
aws.greengrass.LocalDebugConsole/2.0.3/s3/ggv2-component-releases-prod-pdx/
EvergreenHttpDebugView/2ffc496ba41b39568968b22c582b4714a937193ee7687a45527238e696672521/
aws.greengrass.LocalDebugConsole/aws.greengrass.LocalDebugConsole.jar?X-Amz-
Security-Token=KwFLKSdEXAMPLE..."
}
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [GetComponentVersionArtifact](#) di Referensi AWS CLI Perintah.

## get-component

Contoh kode berikut menunjukkan cara menggunakan `get-component`.

### AWS CLI

Contoh 1: Untuk mengunduh resep komponen dalam format YAMM (Linux, macOS, atau Unix)

`get-component` Contoh berikut mengunduh resep komponen Hello World ke file dalam format YAMM. Perintah ini melakukan hal berikut:

Menggunakan `--query` parameter `--output` dan untuk mengontrol output perintah. Parameter ini mengekstrak gumpalan resep dari output perintah. Untuk informasi selengkapnya tentang mengontrol output, lihat [Mengontrol Output AWS Perintah](#) di Panduan Pengguna Antarmuka Baris Perintah. Menggunakan base64 utilitas. Utilitas ini menerjemahkan gumpalan yang diekstraksi ke teks asli. Gumpalan yang dikembalikan oleh `get-component` perintah yang berhasil adalah teks yang dikodekan base64. Anda harus memecahkan kode gumpalan ini untuk mendapatkan teks asli. Menyimpan teks yang diterjemahkan ke file. Bagian terakhir dari perintah (`> com.example.HelloWorld-1.0.0.json`) menyimpan teks yang diterjemahkan ke file.

```
aws greengrassv2 get-component \  
  --arn arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0 \  
  --recipe-output-format YAML \  
  --query recipe \  
  --output text | base64 --decode > com.example.HelloWorld-1.0.0.json
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

Contoh 2: Untuk mengunduh resep komponen dalam format YAMM (Windows CMD)

`get-component` Contoh berikut mengunduh resep komponen Hello World ke file dalam format YAMM. Perintah ini menggunakan `certutil` utilitas.

```
aws greengrassv2 get-component ^  
  --arn arn:aws:greengrass:us-  
west-2:675946970638:components:com.example.HelloWorld:versions:1.0.0 ^  
  --recipe-output-format YAML ^  
  --query recipe ^  
  --output text > com.example.HelloWorld-1.0.0.yaml.b64  
  
certutil -decode com.example.HelloWorld-1.0.0.yaml.b64  
com.example.HelloWorld-1.0.0.yaml
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

Contoh 3: Untuk mengunduh resep komponen dalam format YAMM (Windows PowerShell)



`get-component` Contoh berikut mengunduh resep komponen Hello World ke file dalam format YAMM. Perintah ini menggunakan `certutil` utilitas.

```
aws greengrassv2 get-component `
  --arn arn:aws:greengrass:us-
west-2:675946970638:components:com.example>HelloWorld:versions:1.0.0 `
  --recipe-output-format YAML `
  --query recipe `
  --output text > com.example>HelloWorld-1.0.0.yaml.b64

certutil -decode com.example>HelloWorld-1.0.0.yaml.b64
com.example>HelloWorld-1.0.0.yaml
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [GetComponent](#) di Referensi AWS CLI Perintah.

## get-connectivity-info

Contoh kode berikut menunjukkan cara menggunakan `get-connectivity-info`.

### AWS CLI

Untuk mendapatkan informasi konektivitas untuk perangkat inti Greengrass

`get-connectivity-info` Contoh berikut mendapatkan informasi konektivitas untuk perangkat inti Greengrass. Perangkat klien menggunakan informasi ini untuk terhubung ke broker MQTT yang berjalan pada perangkat inti ini.

```
aws greengrassv2 get-connectivity-info \
  --thing-name MyGreengrassCore
```

Output:

```
{
  "connectivityInfo": [
    {
      "id": "localIP_192.0.2.0",
      "hostAddress": "192.0.2.0",
      "portNumber": 8883
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Mengelola titik akhir perangkat inti di Panduan Pengembang Greengrass V2 AWS IoT](#).

- Untuk detail API, lihat [GetConnectivityInfo](#) di Referensi AWS CLI Perintah.

## get-core-device

Contoh kode berikut menunjukkan cara menggunakan `get-core-device`.

### AWS CLI

Untuk mendapatkan perangkat inti

`get-core-device` Contoh berikut mendapatkan informasi tentang perangkat inti AWS Greengrass IoT.

```
aws greengrassv2 get-core-device \
  --core-device-thing-name MyGreengrassCore
```

Output:

```
{
  "coreDeviceThingName": "MyGreengrassCore",
  "coreVersion": "2.0.3",
  "platform": "linux",
  "architecture": "amd64",
  "status": "HEALTHY",
  "lastStatusUpdateTimestamp": "2021-01-08T04:57:58.838000-08:00",
  "tags": {}
}
```

Untuk informasi selengkapnya, lihat [Periksa status perangkat inti di Panduan Pengembang AWS Greengrass V2 IoT](#).

- Untuk detail API, lihat [GetCoreDevice](#) di Referensi AWS CLI Perintah.

## get-deployment

Contoh kode berikut menunjukkan cara menggunakan `get-deployment`.

## AWS CLI

Untuk mendapatkan penerapan

`get-deployment` Contoh berikut mendapatkan informasi tentang penyebaran komponen inti Greengrass AWS IoT ke sekelompok perangkat inti.

```
aws greengrassv2 get-deployment \  
  --deployment-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/  
MyGreengrassCoreGroup",  
  "revisionId": "14",  
  "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "deploymentName": "Deployment for MyGreengrassCoreGroup",  
  "deploymentStatus": "ACTIVE",  
  "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE22222",  
  "components": {  
    "aws.greengrass.Nucleus": {  
      "componentVersion": "2.0.3",  
      "configurationUpdate": {  
        "merge": "{\"jvmOptions\":\"-Xmx64m\",\"logging\":{\"level\":\"WARN  
\"}}\",  
        "reset": [  
          "/networkProxy",  
          "/mqtt"  
        ]  
      }  
    }  
  },  
  "deploymentPolicies": {  
    "failureHandlingPolicy": "ROLLBACK",  
    "componentUpdatePolicy": {  
      "timeoutInSeconds": 60,  
      "action": "NOTIFY_COMPONENTS"  
    },  
    "configurationValidationPolicy": {  
      "timeoutInSeconds": 60
```

```
    }
  },
  "iotJobConfiguration": {},
  "creationTimestamp": "2021-01-07T17:21:20.691000-08:00",
  "isLatestForTarget": false,
  "tags": {}
}
```

Untuk informasi selengkapnya, lihat [Menerapkan komponen ke perangkat](#) di Panduan Pengembang AWS Greengrass V2.

- Untuk detail API, lihat [GetDeployment](#) di Referensi AWS CLI Perintah.

### **get-service-role-for-account**

Contoh kode berikut menunjukkan cara menggunakan `get-service-role-for-account`.

#### AWS CLI

Untuk mendapatkan peran layanan Greengrass untuk akun Anda AWS

`get-service-role-for-account` Contoh berikut mendapatkan peran layanan yang terkait dengan AWS IoT Greengrass untuk akun Anda. AWS

```
aws greengrassv2 get-service-role-for-account
```

Output:

```
{
  "associatedAt": "2022-01-19T19:21:53Z",
  "roleArn": "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
}
```

Untuk informasi selengkapnya, lihat Peran [layanan Greengrass di Panduan Pengembang Greengrass V2 IoT AWS](#) .

- Untuk detail API, lihat [GetServiceRoleForAccount](#) di Referensi AWS CLI Perintah.

### **list-client-devices-associated-with-core-device**

Contoh kode berikut menunjukkan cara menggunakan `list-client-devices-associated-with-core-device`.

## AWS CLI

Untuk membuat daftar perangkat klien yang terkait dengan perangkat inti

`list-client-devices-associated-with-core-device` Contoh berikut mencantumkan semua perangkat klien yang terkait dengan perangkat inti.

```
aws greengrassv2 list-client-devices-associated-with-core-device \
  --core-device-thing-name MyTestGreengrassCore
```

Output:

```
{
  "associatedClientDevices": [
    {
      "thingName": "MyClientDevice2",
      "associationTimestamp": "2021-07-12T16:33:55.843000-07:00"
    },
    {
      "thingName": "MyClientDevice1",
      "associationTimestamp": "2021-07-12T16:33:55.843000-07:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Berinteraksi dengan perangkat IoT lokal di Panduan Pengembang AWS Greengrass V2](#).

- Untuk detail API, lihat [ListClientDevicesAssociatedWithCoreDevice](#) di Referensi AWS CLI Perintah.

## **list-component-versions**

Contoh kode berikut menunjukkan cara menggunakan `list-component-versions`.

## AWS CLI

Untuk membuat daftar versi komponen

`list-component-versions` Contoh berikut mencantumkan semua versi komponen Hello World.

```
aws greengrassv2 list-component-versions \  
  --arn arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld
```

Output:

```
{  
  "componentVersions": [  
    {  
      "componentName": "com.example.HelloWorld",  
      "componentVersion": "1.0.1",  
      "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.1"  
    },  
    {  
      "componentName": "com.example.HelloWorld",  
      "componentVersion": "1.0.0",  
      "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [ListComponentVersions](#) di Referensi AWS CLI Perintah.

## list-components

Contoh kode berikut menunjukkan cara menggunakan `list-components`.

### AWS CLI

Untuk daftar komponen

`list-components` Contoh berikut mencantumkan setiap komponen dan versi terbarunya yang ditentukan di AWS akun Anda di Wilayah saat ini.

```
aws greengrassv2 list-components
```

**Output:**

```
{
  "components": [
    {
      "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld",
      "componentName": "com.example.HelloWorld",
      "latestVersion": {
        "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.1",
        "componentVersion": "1.0.1",
        "creationTimestamp": "2021-01-08T16:51:07.352000-08:00",
        "description": "My first AWS IoT Greengrass component.",
        "publisher": "Amazon",
        "platforms": [
          {
            "attributes": {
              "os": "linux"
            }
          }
        ]
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola komponen](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [ListComponents](#) di Referensi AWS CLI Perintah.

**list-core-devices**

Contoh kode berikut menunjukkan cara menggunakan `list-core-devices`.

**AWS CLI**

Untuk membuat daftar perangkat inti

`list-core-devices` Contoh berikut mencantumkan perangkat inti AWS Greengrass IoT di akun Anda di Wilayah saat ini. AWS

```
aws greengrassv2 list-core-devices
```

Output:

```
{
  "coreDevices": [
    {
      "coreDeviceThingName": "MyGreengrassCore",
      "status": "HEALTHY",
      "lastStatusUpdateTimestamp": "2021-01-08T04:57:58.838000-08:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Periksa status perangkat inti di Panduan](#) Pengembang AWS Greengrass V2 IoT.

- Untuk detail API, lihat [ListCoreDevices](#) di Referensi AWS CLI Perintah.

## list-deployments

Contoh kode berikut menunjukkan cara menggunakan `list-deployments`.

### AWS CLI

Untuk membuat daftar penerapan

`list-deployments` Contoh berikut mencantumkan revisi terbaru dari setiap penerapan yang ditentukan di AWS akun Anda di Wilayah saat ini.

```
aws greengrassv2 list-deployments
```

Output:

```
{
  "deployments": [
    {
      "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/MyGreengrassCoreGroup",
      "revisionId": "14",
    }
  ]
}
```



```

    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "deploymentName": "Deployment for MyGreengrassCoreGroup",
    "creationTimestamp": "2021-01-07T17:21:20.691000-08:00",
    "deploymentStatus": "ACTIVE",
    "isLatestForTarget": false
  },
  {
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/
MyGreengrassCore",
    "revisionId": "1",
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "deploymentName": "Deployment for MyGreengrassCore",
    "creationTimestamp": "2021-01-06T16:10:42.407000-08:00",
    "deploymentStatus": "COMPLETED",
    "isLatestForTarget": false
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menerapkan komponen ke perangkat](#) di Panduan Pengembang AWS Greengrass V2.

- Untuk detail API, lihat [ListDeployments](#) di Referensi AWS CLI Perintah.

## list-effective-deployments

Contoh kode berikut menunjukkan cara menggunakan `list-effective-deployments`.

### AWS CLI

Untuk membuat daftar pekerjaan penerapan

`list-effective-deployments` Contoh berikut mencantumkan penerapan yang berlaku untuk perangkat inti AWS Greengrass IoT.

```

aws greengrassv2 list-effective-deployments \
  --core-device-thing-name MyGreengrassCore

```

Output:

```

{

```

```

"effectiveDeployments": [
  {
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "deploymentName": "Deployment for MyGreengrassCore",
    "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/
MyGreengrassCore",
    "coreDeviceExecutionStatus": "COMPLETED",
    "reason": "SUCCESSFUL",
    "creationTimestamp": "2021-01-06T16:10:42.442000-08:00",
    "modifiedTimestamp": "2021-01-08T17:21:27.830000-08:00"
  },
  {
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "deploymentName": "Deployment for MyGreengrassCoreGroup",
    "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-
cdef-EXAMPLE44444",
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
    "coreDeviceExecutionStatus": "SUCCEEDED",
    "reason": "SUCCESSFUL",
    "creationTimestamp": "2021-01-07T17:19:20.394000-08:00",
    "modifiedTimestamp": "2021-01-07T17:21:20.721000-08:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Periksa status perangkat inti di Panduan Pengembang AWS Greengrass V2 IoT](#).

- Untuk detail API, lihat [ListEffectiveDeployments](#) di Referensi AWS CLI Perintah.

## list-installed-components

Contoh kode berikut menunjukkan cara menggunakan `list-installed-components`.

### AWS CLI

Untuk membuat daftar komponen yang diinstal pada perangkat inti

`list-installed-components` Contoh berikut mencantumkan komponen yang diinstal pada perangkat inti AWS Greengrass IoT.

```
aws greengrassv2 list-installed-components \  
  --core-device-thing-name MyGreengrassCore
```

Output:

```
{  
  "installedComponents": [  
    {  
      "componentName": "aws.greengrass.Cli",  
      "componentVersion": "2.0.3",  
      "lifecycleState": "RUNNING",  
      "isRoot": true  
    },  
    {  
      "componentName": "aws.greengrass.Nucleus",  
      "componentVersion": "2.0.3",  
      "lifecycleState": "FINISHED",  
      "isRoot": true  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Periksa status perangkat inti di Panduan Pengembang AWS Greengrass V2 IoT](#).

- Untuk detail API, lihat [ListInstalledComponents](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk perangkat inti AWS Greengrass IoT.

```
aws greengrassv2 list-tags-for-resource \  
  --resource-arn arn:aws:greengrass:us-  
west-2:123456789012:coreDevices:MyGreengrassCore
```

Output:

```
{
  "tags": {
    "Owner": "richard-roe"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tanda ke sumber daya

`tag-resource` Contoh berikut menambahkan tag pemilik ke perangkat inti AWS Greengrass IoT. Anda dapat menggunakan tag ini untuk mengontrol akses ke perangkat inti berdasarkan siapa yang memilikinya.

```
aws greengrassv2 tag-resource \
  --resource-arn arn:aws:greengrass:us-
west-2:123456789012:coreDevices:MyGreengrassCore \
  --tags Owner=richard-roe
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag pemilik dari perangkat inti AWS Greengrass IoT.

```
aws iotsitewise untag-resource \  
  --resource-arn arn:aws:greengrass:us-  
west-2:123456789012:coreDevices:MyGreengrassCore \  
  --tag-keys Owner
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan AWS Pengembang Greengrass V2.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-connectivity-info

Contoh kode berikut menunjukkan cara menggunakan `update-connectivity-info`.

## AWS CLI

Untuk memperbarui informasi konektivitas untuk perangkat inti Greengrass

`update-connectivity-info` Contoh berikut mendapatkan informasi konektivitas untuk perangkat inti Greengrass. Perangkat klien menggunakan informasi ini untuk terhubung ke broker MQTT yang berjalan pada perangkat inti ini.

```
aws greengrassv2 update-connectivity-info \  
  --thing-name MyGreengrassCore \  
  --cli-input-json file://core-device-connectivity-info.json
```

Isi dari `core-device-connectivity-info.json`:

```
{  
  "connectivityInfo": [  
    {  
      "hostAddress": "192.0.2.0",
```

```
        "portNumber": 8883,  
        "id": "localIP_192.0.2.0"  
    }  
]  
}
```

Output:

```
{  
  "version": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola titik akhir perangkat inti di Panduan Pengembang Greengrass V2 AWS IoT](#).

- Untuk detail API, lihat [UpdateConnectivityInfo](#) di Referensi AWS CLI Perintah.

## AWS IoT Jobs SDK release contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Jobs SDK release.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-job-execution**

Contoh kode berikut menunjukkan cara menggunakan `describe-job-execution`.

## AWS CLI

Untuk mendapatkan detail eksekusi pekerjaan

`describe-job-execution` Contoh berikut mengambil rincian eksekusi terbaru dari pekerjaan dan hal yang ditentukan.

```
aws iot-jobs-data describe-job-execution \  
  --job-id SampleJob \  
  --thing-name MotionSensor1 \  
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{  
  "execution": {  
    "approximateSecondsBeforeTimedOut": 88,  
    "executionNumber": 2939653338,  
    "jobId": "SampleJob",  
    "lastUpdatedAt": 1567701875.743,  
    "queuedAt": 1567701902.444,  
    "status": "QUEUED",  
    "thingName": "MotionSensor1 ",  
    "versionNumber": 3  
  }  
}
```

Untuk informasi selengkapnya, lihat [Perangkat dan Pekerjaan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [DescribeJobExecution](#) di Referensi AWS CLI Perintah.

## **get-pending-job-executions**

Contoh kode berikut menunjukkan cara menggunakan `get-pending-job-executions`.

## AWS CLI

Untuk mendapatkan daftar semua pekerjaan yang tidak dalam status terminal untuk suatu hal

`get-pending-job-executions` Contoh berikut menampilkan daftar semua pekerjaan yang tidak dalam status terminal untuk hal yang ditentukan.

```
aws iot-jobs-data get-pending-job-executions \  
  --thing-name MotionSensor1 \  
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{  
  "InProgressJobs": [  
  ],  
  "queuedJobs": [  
    {  
      "executionNumber": 2939653338,  
      "jobId": "SampleJob",  
      "lastUpdatedAt": 1567701875.743,  
      "queuedAt": 1567701902.444,  
      "versionNumber": 3  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Perangkat dan Pekerjaan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetPendingJobExecutions](#) di Referensi AWS CLI Perintah.

## start-next-pending-job-execution

Contoh kode berikut menunjukkan cara menggunakan `start-next-pending-job-execution`.

AWS CLI

Untuk mendapatkan dan memulai eksekusi pekerjaan tertunda berikutnya untuk suatu hal

`start-next-pending-job-execution` Contoh berikut mengambil dan memulai eksekusi pekerjaan berikutnya yang statusnya `IN_PROGRESS` atau `QUEUED` untuk hal yang ditentukan.

```
aws iot-jobs-data start-next-pending-job-execution \  
  --thing-name MotionSensor1 \  
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:



```
{
  "execution": {
    "approximateSecondsBeforeTimedOut": 88,
    "executionNumber": 2939653338,
    "jobId": "SampleJob",
    "lastUpdatedAt": 1567714853.743,
    "queuedAt": 1567701902.444,
    "startedAt": 1567714871.690,
    "status": "IN_PROGRESS",
    "thingName": "MotionSensor1 ",
    "versionNumber": 3
  }
}
```

Untuk informasi selengkapnya, lihat [Perangkat dan Pekerjaan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [StartNextPendingJobExecution](#) di Referensi AWS CLI Perintah.

## update-job-execution

Contoh kode berikut menunjukkan cara menggunakan `update-job-execution`.

### AWS CLI

Untuk memperbarui status eksekusi pekerjaan

`update-job-execution` Contoh berikut memperbarui status pekerjaan dan hal yang ditentukan.

```
aws iot-jobs-data update-job-execution \
  --job-id SampleJob \
  --thing-name MotionSensor1 \
  --status REMOVED \
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{
  "executionState": {
    "status": "REMOVED",
    "versionNumber": 3
  },
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Perangkat dan Pekerjaan](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateJobExecution](#) di Referensi AWS CLI Perintah.

## AWS IoT SiteWise contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT SiteWise.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-assets**

Contoh kode berikut menunjukkan cara menggunakan `associate-assets`.

AWS CLI

Untuk mengaitkan aset anak dengan aset orang tua

`associate-assets` Contoh berikut mengaitkan aset turbin angin ke aset ladang angin, di mana model aset turbin angin ada sebagai hierarki dalam model aset ladang angin.

```
aws iotsitewise associate-assets \  
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \  
  --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE \  
  --parent-asset-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

```
--child-asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan aset](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [AssociateAssets](#) di Referensi AWS CLI Perintah.

## batch-associate-project-assets

Contoh kode berikut menunjukkan cara menggunakan `batch-associate-project-assets`.

### AWS CLI

Untuk mengaitkan aset dengan proyek

`batch-associate-project-assets` Contoh berikut mengaitkan aset ladang angin ke sebuah proyek.

```
aws iotsitewise batch-associate-project-assets \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE \  
  --asset-ids a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan aset ke proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [BatchAssociateProjectAssets](#) di Referensi AWS CLI Perintah.

## batch-disassociate-project-assets

Contoh kode berikut menunjukkan cara menggunakan `batch-disassociate-project-assets`.

### AWS CLI

Untuk memisahkan aset dari proyek

`batch-disassociate-project-assets` Contoh berikut memisahkan aset ladang angin dari sebuah proyek.

```
aws iotsitewise batch-disassociate-project-assets \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE
```

```
--project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE \  
--asset-ids a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan aset ke proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [BatchDisassociateProjectAssets](#) di Referensi AWS CLI Perintah.

## batch-put-asset-property-value

Contoh kode berikut menunjukkan cara menggunakan `batch-put-asset-property-value`.

### AWS CLI

Untuk mengirim data ke properti aset

`batch-put-asset-property-value` Contoh berikut mengirimkan data daya dan suhu ke properti aset yang diidentifikasi oleh alias properti.

```
aws iotsitewise batch-put-asset-property-value \  
--cli-input-json file://batch-put-asset-property-value.json
```

Isi dari `batch-put-asset-property-value.json`:

```
{  
  "entries": [  
    {  
      "entryId": "1575691200-company-windfarm-3-turbine-7-power",  
      "propertyAlias": "company-windfarm-3-turbine-7-power",  
      "propertyValues": [  
        {  
          "value": {  
            "doubleValue": 4.92  
          },  
          "timestamp": {  
            "timeInSeconds": 1575691200  
          },  
          "quality": "GOOD"  
        }  
      ]  
    }  
  ]  
}
```

```

    ]
  },
  {
    "entryId": "1575691200-company-windfarm-3-turbine-7-temperature",
    "propertyAlias": "company-windfarm-3-turbine-7-temperature",
    "propertyValues": [
      {
        "value": {
          "integerValue": 38
        },
        "timestamp": {
          "timeInSeconds": 1575691200
        }
      }
    ]
  }
]
}

```

Output:

```

{
  "errorEntries": []
}

```

Untuk informasi selengkapnya, lihat [Menyerap data menggunakan API AWS SiteWise IoT](#) di Panduan Pengguna AWS IoT SiteWise .

- Untuk detail API, lihat [BatchPutAssetPropertyValue](#) di Referensi AWS CLI Perintah.

## create-access-policy

Contoh kode berikut menunjukkan cara menggunakan `create-access-policy`.

### AWS CLI

Contoh 1: Untuk memberikan akses administratif pengguna ke portal

`create-access-policy` Contoh berikut membuat kebijakan akses yang memberikan akses administratif pengguna ke portal web untuk perusahaan peternakan angin.

```
aws iotsitewise create-access-policy \
```

```
--cli-input-json file://create-portal-administrator-access-policy.json
```

Isi dari create-portal-administrator-access-policy.json:

```
{
  "accessPolicyIdentity": {
    "user": {
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbbEXAMPLE"
    }
  },
  "accessPolicyPermission": "ADMINISTRATOR",
  "accessPolicyResource": {
    "portal": {
      "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE"
    }
  }
}
```

Output:

```
{
  "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",
  "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan atau menghapus administrator portal](#) di Panduan Pengguna AWS SiteWise IoT.

Contoh 2: Untuk memberikan pengguna akses hanya-baca ke proyek

create-access-policyContoh berikut membuat kebijakan akses yang memberikan pengguna akses hanya-baca ke proyek ladang angin.

```
aws iotsitewise create-access-policy \
  --cli-input-json file://create-project-viewer-access-policy.json
```

Isi dari create-project-viewer-access-policy.json:

```
{
  "accessPolicyIdentity": {
```

```

    "user": {
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
    }
  },
  "accessPolicyPermission": "VIEWER",
  "accessPolicyResource": {
    "project": {
      "id": "a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE"
    }
  }
}

```

Output:

```

{
  "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE",
  "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE"
}

```

Untuk informasi selengkapnya, lihat [Menetapkan pemirsa proyek](#) di Panduan Aplikasi Monitor AWS SiteWise IoT.

- Untuk detail API, lihat [CreateAccessPolicy](#) di Referensi AWS CLI Perintah.

## create-asset-model

Contoh kode berikut menunjukkan cara menggunakan `create-asset-model`.

### AWS CLI

Untuk membuat model aset

`create-asset-model` Contoh berikut menciptakan model aset yang mendefinisikan turbin angin dengan sifat-sifat berikut:

Nomor seri - Nomor seri turbin angin  
 Daya energi - Aliran data daya yang dihasilkan dari turbin angin  
 Suhu C - Aliran data suhu dari turbin angin di Celsius  
 Temperature F - Titik data suhu yang dipetakan dari Celsius ke Fahrenheit

```

aws iotsitewise create-asset-model \
  --cli-input-json file://create-wind-turbine-model.json

```

## Isi dari create-wind-turbine-model.json:

```
{
  "assetModelName": "Wind Turbine Model",
  "assetModelDescription": "Represents a wind turbine",
  "assetModelProperties": [
    {
      "name": "Serial Number",
      "dataType": "STRING",
      "type": {
        "attribute": {}
      }
    },
    {
      "name": "Generated Power",
      "dataType": "DOUBLE",
      "unit": "kW",
      "type": {
        "measurement": {}
      }
    },
    {
      "name": "Temperature C",
      "dataType": "DOUBLE",
      "unit": "Celsius",
      "type": {
        "measurement": {}
      }
    },
    {
      "name": "Temperature F",
      "dataType": "DOUBLE",
      "unit": "Fahrenheit",
      "type": {
        "transform": {
          "expression": "temp_c * 9 / 5 + 32",
          "variables": [
            {
              "name": "temp_c",
              "value": {
                "propertyId": "Temperature C"
              }
            }
          ]
        }
      }
    }
  ]
}
```



```

    }
  },
  {
    "name": "Total Generated Power",
    "dataType": "DOUBLE",
    "unit": "kW",
    "type": {
      "metric": {
        "expression": "sum(power)",
        "variables": [
          {
            "name": "power",
            "value": {
              "propertyId": "Generated Power"
            }
          }
        ],
        "window": {
          "tumbling": {
            "interval": "1h"
          }
        }
      }
    }
  }
]
}

```

Output:

```

{
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "assetModelStatus": {
    "state": "CREATING"
  }
}

```

Untuk informasi selengkapnya, lihat [Mendefinisikan model aset](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [CreateAssetModel](#) di Referensi AWS CLI Perintah.

## create-asset

Contoh kode berikut menunjukkan cara menggunakan `create-asset`.

### AWS CLI

Untuk membuat aset

`create-asset` Contoh berikut menciptakan aset turbin angin dari model aset turbin angin.

```
aws iotsitewise create-asset \  
  --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --asset-name "Wind Turbine 1"
```

Output:

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
  "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
  "assetStatus": {  
    "state": "CREATING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat aset](#) di SiteWise Panduan Pengguna AWS IoT.

- Untuk detail API, lihat [CreateAsset](#) di Referensi AWS CLI Perintah.

## create-dashboard

Contoh kode berikut menunjukkan cara menggunakan `create-dashboard`.

### AWS CLI

Untuk membuat dasbor

`create-dashboard` Contoh berikut membuat dasbor dengan diagram garis yang menampilkan total daya yang dihasilkan untuk ladang angin.

```
aws iotsitewise create-dashboard \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE \  
  --dashboard-name "Wind Farm" \  
  --dashboard-definition file://create-wind-farm-dashboard.json
```

Isi dari create-wind-farm-dashboard.json:

```
{  
  "widgets": [  
    {  
      "type": "monitor-line-chart",  
      "title": "Generated Power",  
      "x": 0,  
      "y": 0,  
      "height": 3,  
      "width": 3,  
      "metrics": [  
        {  
          "label": "Power",  
          "type": "iotsitewise",  
          "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
          "propertyId": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE"  
        }  
      ]  
    }  
  ]  
}
```

Output:

```
{  
  "dashboardId": "a1b2c3d4-5678-90ab-cdef-ffffEXAMPLE",  
  "dashboardArn": "arn:aws:iotsitewise:us-west-2:123456789012:dashboard/  
a1b2c3d4-5678-90ab-cdef-ffffEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Membuat dasbor \(CLI\)](#) di Panduan Pengguna AWS IoT SiteWise .

- Untuk detail API, lihat [CreateDashboard](#) di Referensi AWS CLI Perintah.

## create-gateway

Contoh kode berikut menunjukkan cara menggunakan `create-gateway`.

### AWS CLI

Untuk membuat gateway

`create-gateway` Contoh berikut membuat gateway yang berjalan di AWS IoT Greengrass.

```
aws iotsitewise create-gateway \  
  --gateway-name ExampleCorpGateway \  
  --gateway-platform greengrass={groupArn=arn:aws:greengrass:us-  
west-2:123456789012:/greengrass/groups/a1b2c3d4-5678-90ab-cdef-1b1b1EXAMPLE}
```

Output:

```
{  
  "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",  
  "gatewayArn": "arn:aws:iotsitewise:us-west-2:123456789012:gateway/  
a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi gateway](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [CreateGateway](#) di Referensi AWS CLI Perintah.

## create-portal

Contoh kode berikut menunjukkan cara menggunakan `create-portal`.

### AWS CLI

Untuk membuat portal

`create-portal` Contoh berikut membuat portal web untuk perusahaan peternakan angin. Anda dapat membuat portal hanya di Wilayah yang sama tempat Anda mengaktifkan AWS Single Sign-On.

```
aws iotsitewise create-portal \  

```

```
--portal-name WindFarmPortal \  
--portal-description "A portal that contains wind farm projects for Example  
Corp." \  
--portal-contact-email support@example.com \  
--role-arn arn:aws:iam::123456789012:role/service-role/  
MySiteWiseMonitorServiceRole
```

### Output:

```
{  
  "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",  
  "portalArn": "arn:aws:iotsitewise:us-west-2:123456789012:portal/  
a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",  
  "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-  
aaaaaEXAMPLE.app.iotsitewise.aws",  
  "portalStatus": {  
    "state": "CREATING"  
  },  
  "ssoApplicationId": "ins-a1b2c3d4-EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Memulai SiteWise Monitor AWS IoT](#) di Panduan SiteWise Pengguna AWS IoT dan [Mengaktifkan SSO AWS di](#) Panduan Pengguna IoT..AWS SiteWise

- Untuk detail API, lihat [CreatePortal](#) di Referensi AWS CLI Perintah.

## create-project

Contoh kode berikut menunjukkan cara menggunakan `create-project`.

### AWS CLI

Untuk membuat proyek

`create-project` Contoh berikut menciptakan proyek ladang angin.

```
aws iotsitewise create-project \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE \  
  --project-name "Wind Farm 1" \  
  --project-description "Contains asset visualizations for Wind Farm #1 for  
Example Corp."
```

**Output:**

```
{
  "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE",
  "projectArn": "arn:aws:iotsitewise:us-west-2:123456789012:project/
a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Membuat proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [CreateProject](#) di Referensi AWS CLI Perintah.

**delete-access-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-access-policy`.

**AWS CLI**

Untuk mencabut akses pengguna ke proyek atau portal

`delete-access-policy` Contoh berikut menghapus kebijakan akses yang memberikan akses administratif pengguna ke portal.

```
aws iotsitewise delete-access-policy \
  --access-policy-id a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan atau menghapus administrator portal](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DeleteAccessPolicy](#) di Referensi AWS CLI Perintah.

**delete-asset-model**

Contoh kode berikut menunjukkan cara menggunakan `delete-asset-model`.

**AWS CLI**

Untuk menghapus model aset

`delete-asset-model` Contoh berikut menghapus model aset turbin angin.

```
aws iotsitewise delete-asset-model \  
  --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{  
  "assetModelStatus": {  
    "state": "DELETING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus model aset](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [DeleteAssetModel](#) di Referensi AWS CLI Perintah.

## **delete-asset**

Contoh kode berikut menunjukkan cara menggunakan `delete-asset`.

### AWS CLI

Untuk menghapus aset

`delete-asset` Contoh berikut menghapus aset turbin angin.

```
aws iotsitewise delete-asset \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Output:

```
{  
  "assetStatus": {  
    "state": "DELETING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus aset](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [DeleteAsset](#) di Referensi AWS CLI Perintah.

## **delete-dashboard**

Contoh kode berikut menunjukkan cara menggunakan `delete-dashboard`.

### AWS CLI

Untuk menghapus dasbor

`delete-dashboard` Contoh berikut menghapus dasbor turbin angin.

```
aws iotsitewise delete-dashboard \  
  --dashboard-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus dasbor](#) di Panduan Aplikasi Monitor AWS IoT SiteWise .

- Untuk detail API, lihat [DeleteDashboard](#) di Referensi AWS CLI Perintah.

## **delete-gateway**

Contoh kode berikut menunjukkan cara menggunakan `delete-gateway`.

### AWS CLI

Untuk menghapus gateway

`delete-gateway` Contoh berikut menghapus gateway.

```
aws iotsitewise delete-gateway \  
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyerap data menggunakan gateway](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DeleteGateway](#) di Referensi AWS CLI Perintah.



## delete-portal

Contoh kode berikut menunjukkan cara menggunakan `delete-portal`.

### AWS CLI

Untuk menghapus portal

`delete-portal` Contoh berikut menghapus portal web untuk perusahaan peternakan angin.

```
aws iotsitewise delete-portal \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Output:

```
{  
  "portalStatus": {  
    "state": "DELETING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus portal](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DeletePortal](#) di Referensi AWS CLI Perintah.

## delete-project

Contoh kode berikut menunjukkan cara menggunakan `delete-project`.

### AWS CLI

Untuk menghapus proyek

`delete-project` Contoh berikut menghapus proyek ladang angin.

```
aws iotsitewise delete-project \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus proyek](#) di Panduan Aplikasi Monitor AWS SiteWise IoT.

- Untuk detail API, lihat [DeleteProject](#) di Referensi AWS CLI Perintah.

## describe-access-policy

Contoh kode berikut menunjukkan cara menggunakan `describe-access-policy`.

### AWS CLI

Untuk mendeskripsikan kebijakan akses

`describe-access-policy` Contoh berikut menjelaskan kebijakan akses yang memberikan akses administratif pengguna ke portal web untuk perusahaan peternakan angin.

```
aws iotsitewise describe-access-policy \  
  --access-policy-id a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE
```

### Output:

```
{  
  "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",  
  "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/  
a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",  
  "accessPolicyIdentity": {  
    "user": {  
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbbEXAMPLE"  
    }  
  },  
  "accessPolicyResource": {  
    "portal": {  
      "id": "a1b2c3d4-5678-90ab-cdef-aaaaEXAMPLE"  
    }  
  },  
  "accessPolicyPermission": "ADMINISTRATOR",  
  "accessPolicyCreationDate": "2020-02-20T22:35:15.552880124Z",  
  "accessPolicyLastUpdateDate": "2020-02-20T22:35:15.552880124Z"  
}
```

Untuk informasi selengkapnya, lihat [Menambahkan atau menghapus administrator portal](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeAccessPolicy](#) di Referensi AWS CLI Perintah.

## describe-asset-model

Contoh kode berikut menunjukkan cara menggunakan `describe-asset-model`.

### AWS CLI

Untuk menggambarkan model aset

`describe-asset-model` Contoh berikut menjelaskan model aset ladang angin.

```
aws iotsitewise describe-asset-model \  
  --asset-model-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Output:

```
{  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/  
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetModelName": "Wind Farm Model",  
  "assetModelDescription": "Represents a wind farm that comprises many wind  
turbines",  
  "assetModelProperties": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "name": "Total Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW",  
      "type": {  
        "metric": {  
          "expression": "sum(power)",  
          "variables": [  
            {  
              "name": "power",  
              "value": {  
                "propertyId": "a1b2c3d4-5678-90ab-  
cdef-66666EXAMPLE",  
                "hierarchyId": "a1b2c3d4-5678-90ab-  
cdef-77777EXAMPLE"  
              }  
            }  
          ],  
          "window": {  
            "tumbling": {
```

```

        "interval": "1h"
      }
    }
  },
  {
    "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",
    "name": "Region",
    "dataType": "STRING",
    "type": {
      "attribute": {
        "defaultValue": " "
      }
    }
  }
],
"assetModelHierarchies": [
  {
    "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
    "name": "Wind Turbines",
    "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
  }
],
"assetModelCreationDate": 1575671284.0,
"assetModelLastUpdateDate": 1575671988.0,
"assetModelStatus": {
  "state": "ACTIVE"
}
}

```

Untuk informasi selengkapnya, lihat [Menjelaskan model aset tertentu](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeAssetModel](#) di Referensi AWS CLI Perintah.

## describe-asset-property

Contoh kode berikut menunjukkan cara menggunakan `describe-asset-property`.

### AWS CLI

Untuk menggambarkan properti aset

`describe-asset-property` Contoh berikut menjelaskan total properti daya yang dihasilkan aset ladang angin.

```
aws iotsitewise describe-asset-property \  
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-99999EXAMPLE
```

Output:

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
  "assetName": "Wind Farm 1",  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetProperty": {  
    "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
    "name": "Total Generated Power",  
    "notification": {  
      "topic": "$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/assets/a1b2c3d4-5678-90ab-cdef-44444EXAMPLE/properties/a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "state": "DISABLED"  
    },  
    "dataType": "DOUBLE",  
    "unit": "kW",  
    "type": {  
      "metric": {  
        "expression": "sum(power)",  
        "variables": [  
          {  
            "name": "power",  
            "value": {  
              "propertyId": "a1b2c3d4-5678-90ab-cdef-66666EXAMPLE",  
              "hierarchyId": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE"  
            }  
          }  
        ],  
        "window": {  
          "tumbling": {  
            "interval": "1h"  
          }  
        }  
      }  
    }  
  }  
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Menjelaskan properti aset tertentu](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeAssetProperty](#) di Referensi AWS CLI Perintah.

## describe-asset

Contoh kode berikut menunjukkan cara menggunakan `describe-asset`.

### AWS CLI

Untuk menggambarkan aset

`describe-asset` Contoh berikut menjelaskan aset ladang angin.

```
aws iotsitewise describe-asset \  
--asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Output:

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
  "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
  "assetName": "Wind Farm 1",  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetProperties": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",  
      "name": "Region",  
      "dataType": "STRING"  
    },  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "name": "Total Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW"  
    }  
  ],  
  "assetHierarchies": [  
    {  
      "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
      "assetName": "Wind Farm 1",  
      "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
      "assetProperties": [  
        {  
          "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",  
          "name": "Region",  
          "dataType": "STRING"  
        },  
        {  
          "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
          "name": "Total Generated Power",  
          "dataType": "DOUBLE",  
          "unit": "kW"  
        }  
      ],  
      "assetType": "Wind Farm"  
    }  
  ]  
}
```

```

    {
      "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
      "name": "Wind Turbines"
    }
  ],
  "assetCreationDate": 1575672453.0,
  "assetLastUpdateDate": 1575672453.0,
  "assetStatus": {
    "state": "ACTIVE"
  }
}

```

Untuk informasi selengkapnya, lihat [Menjelaskan aset tertentu](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeAsset](#) di Referensi AWS CLI Perintah.

## describe-dashboard

Contoh kode berikut menunjukkan cara menggunakan `describe-dashboard`.

### AWS CLI

Untuk menggambarkan dasbor

`describe-dashboard` Contoh berikut menjelaskan dasbor ladang angin yang ditentukan.

```

aws iotsitewise describe-dashboard \
  --dashboard-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE

```

Output:

```

{
  "dashboardId": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
  "dashboardArn": "arn:aws:iotsitewise:us-west-2:123456789012:dashboard/a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
  "dashboardName": "Wind Farm",
  "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE",
  "dashboardDefinition": "{\"widgets\": [{\"type\": \"monitor-line-chart\", \"title\": \"Generated Power\", \"x\": 0, \"y\": 0, \"height\": 3, \"width\": 3, \"metrics\": [{\"label\": \"Power\", \"type\": \"iotsitewise\", \"assetId\": \"a1b2c3d4-5678-90ab-cdef-44444EXAMPLE\", \"propertyId\": \"a1b2c3d4-5678-90ab-cdef-99999EXAMPLE\"}]}]}",
  "dashboardCreationDate": "2020-05-01T20:32:12.228476348Z",

```

```
"dashboardLastUpdateDate": "2020-05-01T20:32:12.228476348Z"
}
```

Untuk informasi selengkapnya, lihat [Melihat dasbor](#) di Panduan Aplikasi Monitor AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeDashboard](#) di Referensi AWS CLI Perintah.

## describe-gateway-capability-configuration

Contoh kode berikut menunjukkan cara menggunakan `describe-gateway-capability-configuration`.

### AWS CLI

Untuk menggambarkan kemampuan gateway

`describe-gateway-capability-configuration` Contoh berikut menjelaskan kemampuan sumber OPC-UA.

```
aws iotsitewise describe-gateway-capability-configuration \
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
  --capability-namespace "iotsitewise:opcuacollector:1"
```

### Output:

```
{
  "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
  "capabilityNamespace": "iotsitewise:opcuacollector:1",
  "capabilityConfiguration": "{\n\"sources\":[\n{\n\"name\":\n\"Wind Farm #1\",
\n\"endpoint\":{\n\"certificateTrust\":{\n\"type\":\n\"TrustAny\",
\n\"endpointUri\":\n\"opc.tcp://203.0.113.0:49320\",
\n\"securityPolicy\":\n\"BASIC256\",
\n\"messageSecurityMode\":\n\"SIGN_AND_ENCRYPT\",
\n\"identityProvider\":
{\n\"type\":\n\"Username\",
\n\"usernameSecretArn\":\n\"arn:aws:secretsmanager:us-east-1:123456789012:secret:green-grass-factory1-auth-3QNDmM\",
\n\"nodeFilterRules\":
[]},
\n\"measurementDataStreamPrefix\":\n\"\"}}],
  \"capabilitySyncStatus\": \"IN_SYNC\"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi sumber data](#) di Panduan AWS Pengguna SiteWise IoT.



- Untuk detail API, lihat [DescribeGatewayCapabilityConfiguration](#) di Referensi AWS CLI Perintah.

## describe-gateway

Contoh kode berikut menunjukkan cara menggunakan `describe-gateway`.

### AWS CLI

Untuk menggambarkan gateway

`describe-gateway` Contoh berikut menjelaskan gateway.

```
aws iotsitewise describe-gateway \
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE
```

### Output:

```
{
  "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
  "gatewayName": "ExampleCorpGateway",
  "gatewayArn": "arn:aws:iotsitewise:us-west-2:123456789012:gateway/
a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
  "gatewayPlatform": {
    "greengrass": {
      "groupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/a1b2c3d4-5678-90ab-cdef-1b1b1EXAMPLE"
    }
  },
  "gatewayCapabilitySummaries": [
    {
      "capabilityNamespace": "iotsitewise:opcuacollector:1",
      "capabilitySyncStatus": "IN_SYNC"
    }
  ],
  "creationDate": 1588369971.457,
  "lastUpdateDate": 1588369971.457
}
```

Untuk informasi selengkapnya, lihat [Menyerap data menggunakan gateway](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeGateway](#) di Referensi AWS CLI Perintah.

## describe-logging-options

Contoh kode berikut menunjukkan cara menggunakan `describe-logging-options`.

### AWS CLI

Untuk mengambil opsi pencatatan AWS SiteWise IoT saat ini

`describe-logging-options` Contoh berikut mengambil opsi pencatatan AWS SiteWise IoT saat ini untuk akun AWS Anda di Wilayah saat ini.

```
aws iotsitewise describe-logging-options
```

Output:

```
{
  "loggingOptions": {
    "level": "INFO"
  }
}
```

Untuk informasi selengkapnya, lihat [Memantau AWS IoT dengan CloudWatch Log SiteWise Amazon](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribeLoggingOptions](#) di Referensi AWS CLI Perintah.

## describe-portal

Contoh kode berikut menunjukkan cara menggunakan `describe-portal`.

### AWS CLI

Untuk menggambarkan portal

`describe-portal` Contoh berikut menjelaskan portal web untuk perusahaan peternakan angin.

```
aws iotsitewise describe-portal \
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Output:

```
{
  "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
  "portalArn": "arn:aws:iotsitewise:us-west-2:123456789012:portal/
a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
  "portalName": "WindFarmPortal",
  "portalDescription": "A portal that contains wind farm projects for Example
Corp.",
  "portalClientId": "E-a1b2c3d4e5f6_a1b2c3d4e5f6EXAMPLE",
  "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-
aaaaaEXAMPLE.app.iotsitewise.aws",
  "portalContactEmail": "support@example.com",
  "portalStatus": {
    "state": "ACTIVE"
  },
  "portalCreationDate": "2020-02-04T23:01:52.90248068Z",
  "portalLastUpdateDate": "2020-02-04T23:01:52.90248078Z",
  "roleArn": "arn:aws:iam::123456789012:role/MySiteWiseMonitorServiceRole"
}
```

Untuk informasi selengkapnya, lihat [Mengelola portal Anda](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [DescribePortal](#) di Referensi AWS CLI Perintah.

## describe-project

Contoh kode berikut menunjukkan cara menggunakan `describe-project`.

### AWS CLI

Untuk menggambarkan sebuah proyek

`describe-project` Contoh berikut menjelaskan proyek ladang angin.

```
aws iotsitewise describe-project \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE
```

Output:

```
{
  "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE",
```

```
"projectArn": "arn:aws:iotsitewise:us-west-2:123456789012:project/
a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE",
"projectName": "Wind Farm 1",
"portalId": "a1b2c3d4-5678-90ab-cdef-aaaaEXAMPLE",
"projectDescription": "Contains asset visualizations for Wind Farm #1 for
Example Corp.",
"projectCreationDate": "2020-02-20T21:58:43.362246001Z",
"projectLastUpdateDate": "2020-02-20T21:58:43.362246095Z"
}
```

Untuk informasi selengkapnya, lihat [Melihat detail proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [DescribeProject](#) di Referensi AWS CLI Perintah.

## disassociate-assets

Contoh kode berikut menunjukkan cara menggunakan `disassociate-assets`.

### AWS CLI

Untuk memisahkan aset anak dari aset orang tua

`disassociate-assets` Contoh berikut memisahkan aset turbin angin dari aset ladang angin.

```
aws iotsitewise disassociate-assets \
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
  --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE \
  --child-asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan aset](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [DisassociateAssets](#) di Referensi AWS CLI Perintah.

## get-asset-property-aggregates

Contoh kode berikut menunjukkan cara menggunakan `get-asset-property-aggregates`.

### AWS CLI

Untuk mengambil nilai rata-rata agregat dan menghitung nilai properti aset

`get-asset-property-aggregates` Contoh berikut mengambil daya total rata-rata aset turbin angin dan jumlah titik data daya total untuk periode 1 jam dalam waktu.

```
aws iotsitewise get-asset-property-aggregates \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \  
  --start-date 1580849400 \  
  --end-date 1580853000 \  
  --aggregate-types AVERAGE COUNT \  
  --resolution 1h
```

Output:

```
{  
  "aggregatedValues": [  
    {  
      "timestamp": 1580850000.0,  
      "quality": "GOOD",  
      "value": {  
        "average": 8723.46538886233,  
        "count": 12.0  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menanyakan agregat properti aset](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [GetAssetPropertyAggregates](#) di Referensi AWS CLI Perintah.

## `get-asset-property-value-history`

Contoh kode berikut menunjukkan cara menggunakan `get-asset-property-value-history`.

AWS CLI

Untuk mengambil nilai historis properti aset

`get-asset-property-value-history` Contoh berikut mengambil nilai daya total aset turbin angin untuk jangka waktu 20 menit.

```
aws iotsitewise get-asset-property-value-history \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \  
  --start-date 1580851800 \  
  --end-date 1580853000
```

### Output:

```
{  
  "assetPropertyValueHistory": [  
    {  
      "value": {  
        "doubleValue": 7217.787046814844  
      },  
      "timestamp": {  
        "timeInSeconds": 1580852100,  
        "offsetInNanos": 0  
      },  
      "quality": "GOOD"  
    },  
    {  
      "value": {  
        "doubleValue": 6941.242811875451  
      },  
      "timestamp": {  
        "timeInSeconds": 1580852400,  
        "offsetInNanos": 0  
      },  
      "quality": "GOOD"  
    },  
    {  
      "value": {  
        "doubleValue": 6976.797662266717  
      },  
      "timestamp": {  
        "timeInSeconds": 1580852700,  
        "offsetInNanos": 0  
      },  
      "quality": "GOOD"  
    },  
    {  
      "value": {  
        "doubleValue": 6890.8677520453875  
      },  
      "timestamp": {  
        "timeInSeconds": 1580853000,  
        "offsetInNanos": 0  
      },  
      "quality": "GOOD"  
    }  
  ]  
}
```

```

    },
    "timestamp": {
      "timeInSeconds": 1580853000,
      "offsetInNanos": 0
    },
    "quality": "GOOD"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menanyakan nilai properti aset historis](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [GetAssetPropertyValueHistory](#) di Referensi AWS CLI Perintah.

## get-asset-property-value

Contoh kode berikut menunjukkan cara menggunakan `get-asset-property-value`.

### AWS CLI

Untuk mengambil nilai aset properti saat ini

`get-asset-property-value` Contoh berikut mengambil daya total aset turbin angin saat ini.

```

aws iotsitewise get-asset-property-value \
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
  --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE

```

Output:

```

{
  "propertyValue": {
    "value": {
      "doubleValue": 6890.8677520453875
    },
    "timestamp": {
      "timeInSeconds": 1580853000,
      "offsetInNanos": 0
    },
    "quality": "GOOD"
  }
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Menanyakan nilai properti aset saat ini](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [GetAssetPropertyValue](#) di Referensi AWS CLI Perintah.

## list-access-policies

Contoh kode berikut menunjukkan cara menggunakan `list-access-policies`.

### AWS CLI

Untuk mencantumkan semua kebijakan akses

`list-access-policies` Contoh berikut mencantumkan semua kebijakan akses untuk pengguna yang merupakan administrator portal.

```
aws iotsitewise list-access-policies \  
  --identity-type USER \  
  --identity-id a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbbEXAMPLE
```

### Output:

```
{  
  "accessPolicySummaries": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",  
      "identity": {  
        "user": {  
          "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbbEXAMPLE"  
        }  
      },  
      "resource": {  
        "portal": {  
          "id": "a1b2c3d4-5678-90ab-cdef-aaaaEXAMPLE"  
        }  
      },  
      "permission": "ADMINISTRATOR"  
    }  
  ]  
}
```



Untuk informasi selengkapnya, lihat [Mengelola portal Anda](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [ListAccessPolicies](#) di Referensi AWS CLI Perintah.

## list-asset-models

Contoh kode berikut menunjukkan cara menggunakan `list-asset-models`.

### AWS CLI

Untuk membuat daftar semua model aset

`list-asset-models` Contoh berikut mencantumkan semua model aset yang ditentukan di AWS akun Anda di Wilayah saat ini.

```
aws iotsitewise list-asset-models
```

Output:

```
{
  "assetModelSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "name": "Wind Farm Model",
      "description": "Represents a wind farm that comprises many wind turbines",
      "creationDate": 1575671284.0,
      "lastUpdateDate": 1575671988.0,
      "status": {
        "state": "ACTIVE"
      }
    },
    {
      "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "name": "Wind Turbine Model",
      "description": "Represents a wind turbine manufactured by Example Corp",
      "creationDate": 1575671207.0,
      "lastUpdateDate": 1575686273.0,
    }
  ]
}
```

```
    "status": {
      "state": "ACTIVE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mencantumkan semua model aset](#) di SiteWise Panduan Pengguna AWS IoT.

- Untuk detail API, lihat [ListAssetModels](#) di Referensi AWS CLI Perintah.

## list-assets

Contoh kode berikut menunjukkan cara menggunakan `list-assets`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua aset tingkat atas

`list-assets` Contoh berikut mencantumkan semua aset yang berada di level teratas dalam pohon hierarki aset dan ditentukan di AWS akun Anda di Wilayah saat ini.

```
aws iotsitewise list-assets \
  --filter TOP_LEVEL
```

Output:

```
{
  "assetSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
      "name": "Wind Farm 1",
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "creationDate": 1575672453.0,
      "lastUpdateDate": 1575672453.0,
      "status": {
        "state": "ACTIVE"
      },
      "hierarchies": [
```

```

        {
            "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
            "name": "Wind Turbines"
        }
    ]
}

```

Untuk informasi selengkapnya, lihat [Mencantumkan aset](#) di SiteWise Panduan Pengguna AWS IoT.

Contoh 2: Untuk membuat daftar semua aset berdasarkan model aset

`list-assets` Contoh berikut mencantumkan semua aset berdasarkan model aset dan didefinisikan di AWS akun Anda di Wilayah saat ini.

```

aws iotsitewise list-assets \
  --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE

```

Output:

```

{
  "assetSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "name": "Wind Turbine 1",
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "creationDate": 1575671550.0,
      "lastUpdateDate": 1575686308.0,
      "status": {
        "state": "ACTIVE"
      },
      "hierarchies": []
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Mencantumkan aset](#) di SiteWise Panduan Pengguna AWS IoT.

- Untuk detail API, lihat [ListAssets](#) di Referensi AWS CLI Perintah.

## list-associated-assets

Contoh kode berikut menunjukkan cara menggunakan `list-associated-assets`.

### AWS CLI

Untuk mencantumkan semua aset yang terkait dengan aset dalam hierarki tertentu

`list-associated-assets` Contoh berikut mencantumkan semua aset turbin angin yang terkait dengan aset ladang angin yang ditentukan.

```
aws iotsitewise list-associated-assets \
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
  --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE
```

Output:

```
{
  "assetSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "name": "Wind Turbine 1",
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "creationDate": 1575671550.0,
      "lastUpdateDate": 1575686308.0,
      "status": {
        "state": "ACTIVE"
      },
      "hierarchies": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mencantumkan aset yang terkait dengan aset tertentu](#) di SiteWise Panduan Pengguna AWS IoT.

- Untuk detail API, lihat [ListAssociatedAssets](#) di Referensi AWS CLI Perintah.

## list-dashboards

Contoh kode berikut menunjukkan cara menggunakan `list-dashboards`.

### AWS CLI

Untuk membuat daftar semua dasbor dalam sebuah proyek

`list-dashboards` Contoh berikut mencantumkan semua dasbor yang didefinisikan dalam sebuah proyek.

```
aws iotsitewise list-dashboards \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE
```

Output:

```
{
  "dashboardSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-ffffEXAMPLE",
      "name": "Wind Farm",
      "creationDate": "2020-05-01T20:32:12.228476348Z",
      "lastUpdateDate": "2020-05-01T20:32:12.228476348Z"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat dasbor](#) di Panduan Aplikasi Monitor AWS SiteWise IoT.

- Untuk detail API, lihat [ListDashboards](#) di Referensi AWS CLI Perintah.

## list-gateways

Contoh kode berikut menunjukkan cara menggunakan `list-gateways`.

### AWS CLI

Untuk daftar semua gateway

`list-gateways` Contoh berikut mencantumkan semua gateway yang ditentukan di AWS akun Anda di Wilayah saat ini.

```
aws iotsitewise list-gateways
```

Output:

```
{
  "gatewaySummaries": [
    {
      "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
      "gatewayName": "ExampleCorpGateway",
      "gatewayCapabilitySummaries": [
        {
          "capabilityNamespace": "iotsitewise:opcuacollector:1",
          "capabilitySyncStatus": "IN_SYNC"
        }
      ],
      "creationDate": 1588369971.457,
      "lastUpdateDate": 1588369971.457
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyerap data menggunakan gateway](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [ListGateways](#) di Referensi AWS CLI Perintah.

## list-portals

Contoh kode berikut menunjukkan cara menggunakan `list-portals`.

AWS CLI

Untuk daftar semua portal

`list-portals` Contoh berikut mencantumkan semua portal yang ditentukan di AWS akun Anda di Wilayah saat ini.

```
aws iotsitewise list-portals
```

Output:

```
{
```

```

    "portalSummaries": [
      {
        "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
        "name": "WindFarmPortal",
        "description": "A portal that contains wind farm projects for Example Corp.",
        "startUrl": "https://a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE.app.iotsitewise.aws",
        "creationDate": "2020-02-04T23:01:52.90248068Z",
        "lastUpdateDate": "2020-02-04T23:01:52.90248078Z",
        "roleArn": "arn:aws:iam::123456789012:role/service-role/MySiteWiseMonitorServiceRole"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola portal Anda](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [ListPortals](#) di Referensi AWS CLI Perintah.

## list-project-assets

Contoh kode berikut menunjukkan cara menggunakan `list-project-assets`.

### AWS CLI

Untuk mencantumkan semua aset yang terkait dengan proyek

`list-project-assets` Contoh berikut mencantumkan semua aset yang terkait dengan proyek ladang angin.

```

aws iotsitewise list-projects \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE

```

Output:

```

{
  "assetIds": [
    "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
  ]
}

```

Untuk informasi selengkapnya, lihat [Menambahkan aset ke proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [ListProjectAssets](#) di Referensi AWS CLI Perintah.

## list-projects

Contoh kode berikut menunjukkan cara menggunakan `list-projects`.

### AWS CLI

Untuk membuat daftar semua proyek di portal

`list-projects` Contoh berikut mencantumkan semua proyek yang didefinisikan dalam portal.

```
aws iotsitewise list-projects \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Output:

```
{  
  "projectSummaries": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE",  
      "name": "Wind Farm 1",  
      "description": "Contains asset visualizations for Wind Farm #1 for  
Example Corp.",  
      "creationDate": "2020-02-20T21:58:43.362246001Z",  
      "lastUpdateDate": "2020-02-20T21:58:43.362246095Z"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat detail proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [ListProjects](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.



## AWS CLI

Untuk mencantumkan semua tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk aset turbin angin.

```
aws iotsitewise list-tags-for-resource \  
  --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-3333EXAMPLE
```

Output:

```
{  
  "tags": {  
    "Owner": "richard-roe"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-logging-options

Contoh kode berikut menunjukkan cara menggunakan `put-logging-options`.

## AWS CLI

Untuk menentukan tingkat logging

`put-logging-options` Contoh berikut memungkinkan INFO level logging di AWS IoT SiteWise. Level lainnya termasuk DEBUG dan OFF.

```
aws iotsitewise put-logging-options \  
  --logging-options level=INFO
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memantau AWS IoT dengan CloudWatch Log SiteWise Amazon](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [PutLoggingOptions](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tanda ke sumber daya

`tag-resource` Contoh berikut menambahkan tag pemilik ke aset turbin angin. Ini memungkinkan Anda mengontrol akses ke aset berdasarkan siapa yang memilikinya.

```
aws iotsitewise tag-resource \  
  --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --tags Owner=richard-roe
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag pemilik dari aset turbin angin.

```
aws iotsitewise untag-resource \  
  --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --tag-keys Owner
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Anda](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-access-policy

Contoh kode berikut menunjukkan cara menggunakan `update-access-policy`.

### AWS CLI

Untuk memberikan kepemilikan proyek kepada penampil proyek

`update-access-policy` Contoh berikut memperbarui kebijakan akses yang memberikan kepemilikan proyek kepada penampil proyek.

```
aws iotsitewise update-access-policy \  
  --access-policy-id a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE \  
  --cli-input-json file://update-project-viewer-access-policy.json
```

Isi dari `update-project-viewer-access-policy.json`:

```
{  
  "accessPolicyIdentity": {  
    "user": {  
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbbEXAMPLE"  
    }  
  },  
  "accessPolicyPermission": "ADMINISTRATOR",  
  "accessPolicyResource": {  
    "project": {  
      "id": "a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE"  
    }  
  }  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menetapkan pemilik proyek](#) di Panduan Aplikasi Monitor AWS SiteWise IoT.

- Untuk detail API, lihat [UpdateAccessPolicy](#) di Referensi AWS CLI Perintah.

## update-asset-model

Contoh kode berikut menunjukkan cara menggunakan `update-asset-model`.

## AWS CLI

Untuk memperbarui model aset

`update-asset-model` Contoh berikut memperbarui deskripsi model aset ladang angin. Contoh ini mencakup ID dan definisi model yang ada, karena `update-asset-model` menimpa model yang ada dengan model baru.

```
aws iotsitewise update-asset-model \  
  --cli-input-json file://update-wind-farm-model.json
```

Isi dari `update-wind-farm-model.json`:

```
{  
  "assetModelName": "Wind Farm Model",  
  "assetModelDescription": "Represents a wind farm that comprises many wind  
turbines",  
  "assetModelProperties": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",  
      "name": "Region",  
      "dataType": "STRING",  
      "type": {  
        "attribute": {}  
      }  
    },  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "name": "Total Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW",  
      "type": {  
        "metric": {  
          "expression": "sum(power)",  
          "variables": [  
            {  
              "name": "power",  
              "value": {  
                "hierarchyId": "a1b2c3d4-5678-90ab-  
cdef-77777EXAMPLE",  
                "propertyId": "a1b2c3d4-5678-90ab-cdef-66666EXAMPLE"  
              }  
            }  
          ]  
        }  
      }  
    }  
  ]  
}
```

```

    ],
    "window": {
      "tumbling": {
        "interval": "1h"
      }
    }
  }
],
"assetModelHierarchies": [
  {
    "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
    "name": "Wind Turbines",
    "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
  }
]
}

```

Output:

```

{
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetModelStatus": {
    "state": "CREATING"
  }
}

```

Untuk informasi selengkapnya, lihat [Memperbarui model aset](#) di SiteWise Panduan Pengguna AWS IoT.

- Untuk detail API, lihat [UpdateAssetModel](#) di Referensi AWS CLI Perintah.

## update-asset-property

Contoh kode berikut menunjukkan cara menggunakan `update-asset-property`.

### AWS CLI

Contoh 1: Untuk memperbarui alias properti aset

`update-asset-property` Contoh berikut memperbarui alias properti daya aset turbin angin.

```
aws iotsitewise update-asset-property \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-55555EXAMPLE \  
  --property-alias "/examplecorp/windfarm/1/turbine/1/power" \  
  --property-notification-state DISABLED
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memetakan aliran data industri ke properti aset](#) di Panduan Pengguna AWS SiteWise IoT.

Contoh 2: Untuk mengaktifkan notifikasi properti aset

`update-asset-property` Contoh berikut memungkinkan pemberitahuan pembaruan properti aset untuk properti daya aset turbin angin. Pembaruan nilai properti dipublikasikan ke topik MQTT\$aws/sitewise/asset-models/<assetModelId>/assets/<assetId>/properties/<propertyId>, di mana setiap ID diganti dengan properti, aset, dan ID model properti aset.

```
aws iotsitewise update-asset-property \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \  
  --property-notification-state ENABLED \  
  --property-alias "/examplecorp/windfarm/1/turbine/1/power"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Berinteraksi dengan layanan lain](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [UpdateAssetProperty](#) di Referensi AWS CLI Perintah.

## update-asset

Contoh kode berikut menunjukkan cara menggunakan `update-asset`.

### AWS CLI

Untuk memperbarui nama aset

`update-asset` Contoh berikut memperbarui nama aset turbin angin.

```
aws iotsitewise update-asset \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --asset-name "Wind Turbine 2"
```

Output:

```
{  
  "assetStatus": {  
    "state": "UPDATING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui aset](#) di SiteWise Panduan Pengguna AWS IoT.

- Untuk detail API, lihat [UpdateAsset](#) di Referensi AWS CLI Perintah.

## update-dashboard

Contoh kode berikut menunjukkan cara menggunakan `update-dashboard`.

### AWS CLI

Untuk memperbarui dasbor

`update-dashboard` Contoh berikut mengubah judul bagan garis dasbor yang menampilkan total daya yang dihasilkan untuk ladang angin.

```
aws iotsitewise update-dashboard \  
  --project-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE \  
  --dashboard-name "Wind Farm" \  
  --dashboard-definition file://update-wind-farm-dashboard.json
```

Isi dari `update-wind-farm-dashboard.json`:

```
{  
  "widgets": [  
    {  
      "type": "monitor-line-chart",  
      "title": "Total Generated Power",
```

```

    "x": 0,
    "y": 0,
    "height": 3,
    "width": 3,
    "metrics": [
      {
        "label": "Power",
        "type": "iotsitewise",
        "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
        "propertyId": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE"
      }
    ]
  }
]
}

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dasbor \(CLI\)](#) di Panduan Pengguna AWS IoT SiteWise .

- Untuk detail API, lihat [UpdateDashboard](#) di Referensi AWS CLI Perintah.

## update-gateway-capability-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-gateway-capability-configuration`.

### AWS CLI

Untuk memperbarui kemampuan gateway

`update-gateway-capability-configuration` Contoh berikut mengonfigurasi sumber OPC-UA dengan properti berikut:

Mempercayai sertifikat apa pun. Menggunakan algoritma Basic256 untuk mengamankan pesan. Menggunakan SignAndEncrypt mode untuk mengamankan koneksi. Menggunakan kredensi otentikasi yang disimpan dalam rahasia Secrets Manager. AWS

```

aws iotsitewise update-gateway-capability-configuration \
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
  --capability-namespace "iotsitewise:opcuacollector:1" \

```



```
--capability-configuration file://opc-ua-capability-configuration.json
```

Isi dari `opc-ua-capability-configuration.json`:

```
{
  "sources": [
    {
      "name": "Wind Farm #1",
      "endpoint": {
        "certificateTrust": {
          "type": "TrustAny"
        },
        "endpointUri": "opc.tcp://203.0.113.0:49320",
        "securityPolicy": "BASIC256",
        "messageSecurityMode": "SIGN_AND_ENCRYPT",
        "identityProvider": {
          "type": "Username",
          "usernameSecretArn": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:greenrass-windfarm1-auth-1ABCDE"
        },
        "nodeFilterRules": []
      },
      "measurementDataStreamPrefix": ""
    }
  ]
}
```

Output:

```
{
  "capabilityNamespace": "iotsitewise:opcuacollector:1",
  "capabilitySyncStatus": "OUT_OF_SYNC"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi sumber data](#) di Panduan AWS Pengguna SiteWise IoT.

- Untuk detail API, lihat [UpdateGatewayCapabilityConfiguration](#) di Referensi AWS CLI Perintah.

## update-gateway

Contoh kode berikut menunjukkan cara menggunakan `update-gateway`.

## AWS CLI

Untuk memperbarui nama gateway

`update-gateway` Contoh berikut memperbarui nama gateway.

```
aws iotsitewise update-gateway \  
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \  
  --gateway-name ExampleCorpGateway1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyerap data menggunakan gateway](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [UpdateGateway](#) di Referensi AWS CLI Perintah.

## `update-portal`

Contoh kode berikut menunjukkan cara menggunakan `update-portal`.

## AWS CLI

Untuk memperbarui detail portal

`update-portal` Contoh berikut memperbarui portal web untuk perusahaan peternakan angin.

```
aws iotsitewise update-portal \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE \  
  --portal-name WindFarmPortal \  
  --portal-description "A portal that contains wind farm projects for Example  
Corp." \  
  --portal-contact-email support@example.com \  
  --role-arn arn:aws:iam::123456789012:role/MySiteWiseMonitorServiceRole
```

Output:

```
{  
  "portalStatus": {  
    "state": "UPDATING"  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengelola portal Anda](#) di Panduan Pengguna AWS SiteWise IoT.

- Untuk detail API, lihat [UpdatePortal](#) di Referensi AWS CLI Perintah.

## update-project

Contoh kode berikut menunjukkan cara menggunakan `update-project`.

### AWS CLI

Untuk memperbarui detail proyek

`update-project` Contoh berikut memperbarui proyek ladang angin.

```
aws iotsitewise update-project \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE \  
  --project-name "Wind Farm 1" \  
  --project-description "Contains asset visualizations for Wind Farm #1 for  
Example Corp."
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengubah detail proyek](#) di Panduan Aplikasi SiteWise Monitor AWS IoT.

- Untuk detail API, lihat [UpdateProject](#) di Referensi AWS CLI Perintah.

## AWS IoT Things Graph contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Things Graph.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-entity-to-thing**

Contoh kode berikut menunjukkan cara menggunakan `associate-entity-to-thing`.

AWS CLI

Untuk mengaitkan sesuatu dengan perangkat

`associate-entity-to-thing` Contoh berikut mengaitkan sesuatu dengan perangkat. Contoh menggunakan perangkat sensor gerak yang ada di namespace publik.

```
aws iotthingsgraph associate-entity-to-thing \  
  --thing-name "MotionSensorName" \  
  --entity-id "urn:tdm:aws/examples:Device:HCSR501MotionSensor"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan Mengunggah Model](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [AssociateEntityToThing](#) di Referensi AWS CLI Perintah.

### **create-flow-template**

Contoh kode berikut menunjukkan cara menggunakan `create-flow-template`.

AWS CLI

Untuk membuat aliran

`create-flow-template` Contoh berikut menciptakan alur (alur kerja). Nilai `MyFlowDefinition` adalah GraphQL yang memodelkan aliran.

```
aws iotthingsgraph create-flow-template \  
  --definition language=GRAPHQL,text="MyFlowDefinition"
```

Output:

```
{  
  "summary": {  
    "createdAt": 1559248067.545,  
    "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",  
    "revisionNumber": 1  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [CreateFlowTemplate](#) di Referensi AWS CLI Perintah.

## create-system-instance

Contoh kode berikut menunjukkan cara menggunakan `create-system-instance`.

### AWS CLI

Untuk membuat instance sistem

`create-system-instance` Contoh berikut membuat instance sistem. Nilai `MySystemInstanceDefinition` adalah GraphQL yang memodelkan instance sistem.

```
aws iotthingsgraph create-system-instance -\  
  -definition language=GRAPHQL,text="MySystemInstanceDefinition" \  
  --target CLOUD \  
  --flow-actions-role-arn myRoleARN
```

Output:

```
{  
  "summary": {  
    "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",  
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/  
Room218",
```

```
    "status": "NOT_DEPLOYED",
    "target": "CLOUD",
    "createdAt": 1559249315.208,
    "updatedAt": 1559249315.208
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [CreateSystemInstance](#) di Referensi AWS CLI Perintah.

## create-system-template

Contoh kode berikut menunjukkan cara menggunakan `create-system-template`.

### AWS CLI

Untuk membuat sistem

`create-system-template` Contoh berikut menciptakan sistem. Nilai `MySystemDefinition` adalah GraphQL yang memodelkan sistem.

```
aws iotthingsgraph create-system-template \
  --definition language=GRAPHQL,text="MySystemDefinition"
```

Output:

```
{
  "summary": {
    "createdAt": 1559249776.254,
    "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/MySystem",
    "revisionNumber": 1
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Sistem](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [CreateSystemTemplate](#) di Referensi AWS CLI Perintah.

## delete-flow-template

Contoh kode berikut menunjukkan cara menggunakan `delete-flow-template`.

### AWS CLI

Untuk menghapus aliran

`delete-flow-template` Contoh berikut menghapus alur (alur kerja).

```
aws iotthingsgraph delete-flow-template \  
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [DeleteFlowTemplate](#) di Referensi AWS CLI Perintah.

## delete-namespace

Contoh kode berikut menunjukkan cara menggunakan `delete-namespace`.

### AWS CLI

Untuk menghapus namespace

`delete-namespace` Contoh berikut menghapus namespace.

```
aws iotthingsgraph delete-namespace
```

Output:

```
{  
  "namespaceArn": "arn:aws:iotthingsgraph:us-west-2:123456789012",  
  "namespaceName": "us-west-2/123456789012/default"  
}
```

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [DeleteNamespaced](#) di Referensi AWS CLI Perintah.

## delete-system-instance

Contoh kode berikut menunjukkan cara menggunakan `delete-system-instance`.

### AWS CLI

Untuk menghapus instance sistem

`delete-system-instance` Contoh berikut menghapus instance sistem.

```
aws iotthingsgraph delete-system-instance \  
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [DeleteSystemInstance](#) di Referensi AWS CLI Perintah.

## delete-system-template

Contoh kode berikut menunjukkan cara menggunakan `delete-system-template`.

### AWS CLI

Untuk menghapus sistem

`delete-system-template` Contoh berikut menghapus sistem.

```
aws iotthingsgraph delete-system-template \  
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [DeleteSystemTemplate](#) di Referensi AWS CLI Perintah.

## deploy-system-instance

Contoh kode berikut menunjukkan cara menggunakan `deploy-system-instance`.



## AWS CLI

Untuk menyebarkan instance sistem

`delete-system-template` Contoh berikut menyebarkan instance sistem.

```
aws iotthingsgraph deploy-system-instance \  
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Output:

```
{  
  "summary": {  
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment:Room218",  
    "createdAt": 1559249776.254,  
    "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",  
    "status": "DEPLOYED_IN_TARGET",  
    "target": "CLOUD",  
    "updatedAt": 1559249776.254  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [DeploySystemInstance](#) di Referensi AWS CLI Perintah.

## `deprecate-flow-template`

Contoh kode berikut menunjukkan cara menggunakan `deprecate-flow-template`.

### AWS CLI

Untuk menghentikan aliran

`deprecate-flow-template` Contoh berikut menghentikan alur (alur kerja).

```
aws iotthingsgraph deprecate-flow-template \  
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [DeprecateFlowTemplate](#) di Referensi AWS CLI Perintah.

## deprecate-system-template

Contoh kode berikut menunjukkan cara menggunakan `deprecate-system-template`.

### AWS CLI

Untuk menghentikan sistem

`deprecate-system-template` Contoh berikut mencela sistem.

```
aws iotthingsgraph deprecate-system-template \  
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [DeprecateSystemTemplate](#) di Referensi AWS CLI Perintah.

## describe-namespace

Contoh kode berikut menunjukkan cara menggunakan `describe-namespace`.

### AWS CLI

Untuk mendapatkan deskripsi namespace Anda

`describe-namespace` Contoh berikut mendapatkan deskripsi namespace Anda.

```
aws iotthingsgraph describe-namespace
```

Output:

```
{  
  "namespaceName": "us-west-2/123456789012/default",
```

```
"trackingNamespaceName": "aws",  
"trackingNamespaceVersion": 1,  
"namespaceVersion": 5  
}
```

Untuk informasi selengkapnya, lihat [Ruang nama](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [DescribeNamespaced](#) di Referensi AWS CLI Perintah.

## dissociate-entity-from-thing

Contoh kode berikut menunjukkan cara menggunakan `dissociate-entity-from-thing`.

### AWS CLI

Untuk memisahkan sesuatu dari perangkat

`dissociate-entity-from-thing` Contoh berikut memisahkan sesuatu dari perangkat.

```
aws iotthingsgraph dissociate-entity-from-thing \  
  --thing-name "MotionSensorName" \  
  --entity-type "DEVICE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat dan Mengunggah Model](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [DissociateEntityFromThing](#) di Referensi AWS CLI Perintah.

## get-entities

Contoh kode berikut menunjukkan cara menggunakan `get-entities`.

### AWS CLI

Untuk mendapatkan definisi untuk entitas

`get-entities` Contoh berikut mendapatkan definisi untuk model perangkat.

```
aws iotthingsgraph get-entities \  
  --ids "urn:tdm:aws/examples:DeviceModel:MotionSensor"
```

**Output:**

```
{
  "descriptions": [
    {
      "id": "urn:tdm:aws/examples:DeviceModel:MotionSensor",
      "type": "DEVICE_MODEL",
      "createdAt": 1559256190.599,
      "definition": {
        "language": "GRAPHQL",
        "text": "##\n# Specification of motion sensor devices interface.\n##
\n#type MotionSensor @deviceModel(id: \"urn:tdm:aws/examples:deviceModel:MotionSensor
\n\", \n      capability: \"urn:tdm:aws/examples:capability:MotionSensorCapability\")
      {ignore:void}"
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengunggah Model](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetEntities](#) di Referensi AWS CLI Perintah.

**get-flow-template-revisions**

Contoh kode berikut menunjukkan cara menggunakan `get-flow-template-revisions`.

**AWS CLI**

Untuk mendapatkan informasi revisi tentang aliran

`get-flow-template-revisions` Contoh berikut mendapatkan informasi revisi tentang alur (alur kerja).

```
aws iotthingsgraph get-flow-template-revisions \
  --id urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow
```

**Output:**

```
{
  "summaries": [
```

```

    {
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.292
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetFlowTemplateRevisions](#) di Referensi AWS CLI Perintah.

## get-flow-template

Contoh kode berikut menunjukkan cara menggunakan `get-flow-template`.

### AWS CLI

Untuk mendapatkan definisi aliran

`get-flow-template` Contoh berikut mendapat definisi untuk alur (alur kerja).

```

aws iotthingsgraph get-flow-template \
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"

```

Output:

```

{
  "description": {
    "summary": {
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.292
    },
    "definition": {
      "language": "GRAPHQL",
      "text": "{\nquery MyFlow($camera: string!, $screen: string!)
@workflowType(id: \"urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow\")
@annotation(type: \"tgc:FlowEvent\", id: \"sledged790c1b2bcd949e09da0c9bfc077f79d
\", x: 1586, y: 653) @triggers(definition: \"{MotionSensor(description:
\\\\\"\\\\\") @position(x: 1045, y: 635.6666564941406) {\n  condition(expr:

```

```

    \\\"devices[name == \\\"\\\\\\\\\\\\\\\\\"motionSensor\\\\\\\\\\\\\\\\\"].events[name == \\\"
    \\\"StateChanged\\\\\\\\\\\\\\\\\"].lastEvent\\\\\")\\n  action(expr: \\\"\\\\\\\\\\\\\\\\\")\\
    \\n}}\\") {\n  variables {\n    cameraResult @property(id: \\\"urn:tdm:aws/
    examples:property:CameraStateProperty\\\")\n  }\n  steps {\n    step(name: \\\"Camera
    \\", outEvent: [\\\"sledged790c1b2bcd949e09da0c9bfc077f79d\\\"] @position(x: 1377,
    y: 638.6666564941406) {\n      DeviceActivity(deviceModel: \\\"urn:tdm:aws/
    examples:deviceModel:Camera\\", out: \\\"cameraResult\\", deviceId: \\\"${camera}\\")
    {\n      capture\n    }\n  }\n  step(name: \\\"Screen\\", inEvent:
    [\\\"sledged790c1b2bcd949e09da0c9bfc077f79d\\\"] @position(x: 1675.6666870117188,
    y: 637.9999847412109) {\n      DeviceActivity(deviceModel: \\\"urn:tdm:aws/
    examples:deviceModel:Screen\\", deviceId: \\\"${screen}\\") {\n        display(imageUrl:
    \\\"${cameraResult.lastClickedImage}\\\")\n      }\n    }\n  }\n}\n}\n}
  },
  "validatedNamespaceVersion": 5
}
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetFlowTemplated](#) di Referensi AWS CLI Perintah.

## get-namespace-deletion-status

Contoh kode berikut menunjukkan cara menggunakan `get-namespace-deletion-status`.

### AWS CLI

Untuk mendapatkan status tugas penghapusan namespace

`get-namespace-deletion-status` Contoh berikut mendapatkan status tugas penghapusan namespace.

```
aws iotthingsgraph get-namespace-deletion-status
```

Output:

```
{
  "namespaceArn": "arn:aws:iotthingsgraph:us-west-2:123456789012",
  "namespaceName": "us-west-2/123456789012/default"
  "status": "SUCCEEDED "
}
```

Untuk informasi selengkapnya, lihat [Ruang nama](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetNamespaceDeletionStatus](#) di Referensi AWS CLI Perintah.

## get-system-instance

Contoh kode berikut menunjukkan cara menggunakan `get-system-instance`.

### AWS CLI

Untuk mendapatkan instance sistem

`get-system-instance` Contoh berikut mendapat definisi untuk instance sistem.

```
aws iotthingsgraph get-system-instance \
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Output:

```
{
  "description": {
    "summary": {
      "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218",
      "status": "NOT_DEPLOYED",
      "target": "CLOUD",
      "createdAt": 1559249315.208,
      "updatedAt": 1559249315.208
    },
    "definition": {
      "language": "GRAPHQL",
      "text": "{\r\nquery Room218 @deployment(id: \"urn:tdm:us-
west-2/123456789012/default:Deployment:Room218\", systemId: \"urn:tdm:us-
west-2/123456789012/default:System:SecurityFlow\") {\r\n  motionSensor(deviceId:
\"MotionSensorName\")\r\n  screen(deviceId: \"ScreenName\")\r\n
camera(deviceId: \"CameraName\") \r\n  triggers {MotionEventTrigger(description:
\"a trigger\") { \r\n    condition(expr: \"devices[name ==
'motionSensor'].events[name == 'StateChanged'].lastEvent\") \r\n    action(expr:
\"ThingsGraph.startFlow('SecurityFlow', bindings[name == 'camera'].deviceId,
bindings[name == 'screen'].deviceId))\r\n  }\r\n  }\r\n  }\r\n  }\r\n  }"
    },
    "metricsConfiguration": {
```

```
        "cloudMetricEnabled": false
    },
    "validatedNamespaceVersion": 5,
    "flowActionsRoleArn": "arn:aws:iam::123456789012:role/ThingsGraphRole"
}
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetSystemInstanced](#) di Referensi AWS CLI Perintah.

## get-system-template-revisions

Contoh kode berikut menunjukkan cara menggunakan `get-system-template-revisions`.

### AWS CLI

Untuk mendapatkan informasi revisi tentang suatu sistem

`get-system-template-revisions` Contoh berikut mendapat informasi revisi tentang suatu sistem.

```
aws iotthingsgraph get-system-template-revisions \
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Output:

```
{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/MySystem",
      "revisionNumber": 1,
      "createdAt": 1559247540.656
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.



- Untuk detail API, lihat [GetSystemTemplateRevisions](#) di Referensi AWS CLI Perintah.

## get-system-template

Contoh kode berikut menunjukkan cara menggunakan `get-system-template`.

### AWS CLI

Untuk mendapatkan sistem

`get-system-template` Contoh berikut mendapat definisi untuk suatu sistem.

```
aws iotthingsgraph get-system-template \
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Output:

```
{
  "description": {
    "summary": {
      "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.656
    },
    "definition": {
      "language": "GRAPHQL",
      "text": "{\n  type MySystem @systemType(id: \"urn:tdm:us-west-2/123456789012/default:System:MySystem\", description: \"\") {\n    camera: Camera @thing(id: \"urn:tdm:aws/examples:deviceModel:Camera\")\n    screen: Screen @thing(id: \"urn:tdm:aws/examples:deviceModel:Screen\")\n    motionSensor: MotionSensor @thing(id: \"urn:tdm:aws/examples:deviceModel:MotionSensor\")\n    MyFlow: MyFlow @workflow(id: \"urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow\")\n  }\n}"
    },
    "validatedNamespaceVersion": 5
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetSystemTemplatedi Referensi AWS CLI Perintah](#).

## get-upload-status

Contoh kode berikut menunjukkan cara menggunakan `get-upload-status`.

### AWS CLI

Untuk mendapatkan status upload entitas Anda

`get-upload-status` Contoh berikut mendapatkan status operasi upload entitas Anda. Nilai `MyUploadId` adalah nilai ID yang dikembalikan oleh `upload-entity-definitions` operasi.

```
aws iotthingsgraph get-upload-status \  
  --upload-id "MyUploadId"
```

Output:

```
{  
  "namespaceName": "us-west-2/123456789012/default",  
  "namespaceVersion": 5,  
  "uploadId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",  
  "uploadStatus": "SUCCEEDED"  
}
```

Untuk informasi selengkapnya, lihat [Entitas Pemodelan](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [GetUploadStatusdi Referensi AWS CLI Perintah](#).

## list-flow-execution-messages

Contoh kode berikut menunjukkan cara menggunakan `list-flow-execution-messages`.

### AWS CLI

Untuk mendapatkan informasi tentang peristiwa dalam eksekusi aliran

`list-flow-execution-messages` Contoh berikut mendapatkan informasi tentang peristiwa dalam eksekusi aliran.

```
aws iotthingsgraph list-flow-execution-messages \  
  --namespace-name "us-west-2/123456789012/default" \  
  --namespace-version 5
```

```
--flow-execution-id "urn:tdm:us-west-2/123456789012/
default:Workflow:SecurityFlow_2019-05-11T19:39:55.317Z_MotionSensor_69b151ad-
a611-42f5-ac21-fe537f9868ad"
```

### Output:

```
{
  "messages": [
    {
      "eventType": "EXECUTION_STARTED",
      "messageId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",
      "payload": "Flow execution started",
      "timestamp": 1559247540.656
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [ListFlowExecutionMessages](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan semua tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk sumber daya Grafik AWS IoT Things.

```
aws iotthingsgraph list-tags-for-resource \
  --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
  default/Room218"
```

### Output:

```
{
  "tags": [
    {
```

```

        "key": "Type",
        "value": "Residential"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Grafik AWS IoT Things Anda](#) di Panduan Pengguna AWS Grafik IoT Things.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## search-entities

Contoh kode berikut menunjukkan cara menggunakan `search-entities`.

### AWS CLI

Untuk mencari entitas

`search-entities` Contoh berikut mencari semua entitas tipe `EVENT`.

```

aws iotthingsgraph search-entities \
  --entity-types "EVENT"

```

Output:

```

{
  "descriptions": [
    {
      "id": "urn:tdm:aws/examples:Event:MotionSensorEvent",
      "type": "EVENT",
      "definition": {
        "language": "GRAPHQL",
        "text": "##\n# Description of events emitted by motion
sensor.\n##\n# type MotionSensorEvent @eventType(id: \"urn:tdm:aws/
examples:event:MotionSensorEvent\", \n          payload: \"urn:tdm:aws/
examples:property:MotionSensorStateProperty\") {ignore:void}"
      }
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Event:CameraClickedEventV2",
      "type": "EVENT",

```

```

      "definition": {
        "language": "GraphQL",
        "text": "type CameraClickedEventV2 @eventType(id: \"urn:tdm:us-
west-2/123456789012/default:event:CameraClickedEventV2\", \r\npayload:
 \"urn:tdm:aws:Property:Boolean\") {ignore:void}"
      }
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Event:MotionSensorEventV2",
      "type": "EVENT",
      "definition": {
        "language": "GraphQL",
        "text": "# Event emitted by the motion sensor.\r\nntype
MotionSensorEventV2 @eventType(id: \"urn:tdm:us-west-2/123456789012/
default:event:MotionSensorEventV2\", \r\npayload: \"urn:tdm:us-west-2/123456789012/
default:property:MotionSensorStateProperty2\") {ignore:void}"
      }
    }
  ],
  "nextToken": "urn:tdm:us-west-2/123456789012/default:Event:MotionSensorEventV2"
}

```

Untuk informasi selengkapnya, lihat Referensi [Model Data Grafik AWS IoT Things](#) di Panduan AWS Pengguna Grafik IoT Things.

- Untuk detail API, lihat [SearchEntities](#) di Referensi AWS CLI Perintah.

## search-flow-executions

Contoh kode berikut menunjukkan cara menggunakan `search-flow-executions`.

### AWS CLI

Untuk mencari eksekusi aliran

`search-flow-executions` Contoh berikut mencari semua eksekusi aliran dalam contoh sistem tertentu.

```

aws iotthingsgraph search-flow-executions \
  --system-instance-id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"

```

Output:

```
{
  "summaries": [
    {
      "createdAt": 1559247540.656,
      "flowExecutionId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",
      "flowTemplateId": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
      "status": "RUNNING ",
      "systemInstanceId": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
      "updatedAt": 1559247540.656
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [SearchFlowExecutions](#) di Referensi AWS CLI Perintah.

## search-flow-templates

Contoh kode berikut menunjukkan cara menggunakan `search-flow-templates`.

### AWS CLI

Untuk mencari alur (atau alur kerja)

`search-flow-templates` Contoh berikut mencari semua alur (alur kerja) yang berisi model perangkat Kamera.

```
aws iotthingsgraph search-flow-templates \
  --filters name="DEVICE_MODEL_ID",value="urn:tdm:aws/examples:DeviceModel:Camera"
```

Output:

```
{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.292
    },
  ],
}
```

```

    {
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:SecurityFlow",
      "revisionNumber": 3,
      "createdAt": 1548283099.27
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [SearchFlowTemplates](#) di Referensi AWS CLI Perintah.

## search-system-instances

Contoh kode berikut menunjukkan cara menggunakan `search-system-instances`.

### AWS CLI

Untuk mencari instance sistem

`search-system-instances` Contoh berikut mencari semua instance sistem yang berisi sistem yang ditentukan.

```

aws iotthingsgraph search-system-instances \
  --filters name="SYSTEM_TEMPLATE_ID",value="urn:tdm:us-west-2/123456789012/
default:System:SecurityFlow"

```

Output:

```

{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:DeploymentForSample",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/DeploymentForSample",
      "status": "NOT_DEPLOYED",
      "target": "GREENGRASS",
      "greengrassGroupName": "ThingsGraphGrnGr",
      "createdAt": 1555716314.707,
      "updatedAt": 1555716314.707
    },
  ],
}

```

```
{
  "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:MockDeployment",
  "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/MockDeployment",
  "status": "DELETED_IN_TARGET",
  "target": "GREENGRASS",
  "greengrassGroupName": "ThingsGraphGrnGr",
  "createdAt": 1549416462.049,
  "updatedAt": 1549416722.361,
  "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
  "greengrassGroupVersionId": "7365aed7-2d3e-4d13-aad8-75443d45eb05"
},
{
  "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:MockDeployment2",
  "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/MockDeployment2",
  "status": "DEPLOYED_IN_TARGET",
  "target": "GREENGRASS",
  "greengrassGroupName": "ThingsGraphGrnGr",
  "createdAt": 1549572385.774,
  "updatedAt": 1549572418.408,
  "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
  "greengrassGroupVersionId": "bfa70ab3-2bf7-409c-a4d4-bc8328ae5b86"
},
{
  "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room215",
  "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room215",
  "status": "NOT_DEPLOYED",
  "target": "GREENGRASS",
  "greengrassGroupName": "ThingsGraphGG",
  "createdAt": 1547056918.413,
  "updatedAt": 1547056918.413
},
{
  "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
  "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218",
  "status": "NOT_DEPLOYED",
  "target": "CLOUD",
  "createdAt": 1559249315.208,
  "updatedAt": 1559249315.208
}
```



```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Sistem dan Konfigurasi Aliran](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [SearchSystemInstances](#) di Referensi AWS CLI Perintah.

## search-system-templates

Contoh kode berikut menunjukkan cara menggunakan `search-system-templates`.

### AWS CLI

Untuk mencari sistem

`search-system-templates` Contoh berikut mencari semua sistem yang berisi aliran yang ditentukan.

```
aws iotthingsgraph search-system-templates \  
  --filters name="FLOW_TEMPLATE_ID",value="urn:tdm:us-west-2/123456789012/  
  default:Workflow:SecurityFlow"
```

Output:

```
{  
  "summaries": [  
    {  
      "id": "urn:tdm:us-west-2/123456789012/default:System:SecurityFlow",  
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/  
SecurityFlow",  
      "revisionNumber": 1,  
      "createdAt": 1548283099.433  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [SearchSystemTemplates](#) di Referensi AWS CLI Perintah.

## search-things

Contoh kode berikut menunjukkan cara menggunakan `search-things`.

### AWS CLI

Untuk mencari hal-hal yang terkait dengan perangkat dan model perangkat

`search-things` Contoh berikut mencari semua hal yang terkait dengan perangkat HCSR501MotionSensor .

```
aws iotthingsgraph search-things \  
  --entity-id "urn:tdm:aws/examples:Device:HCSR501MotionSensor"
```

Output:

```
{  
  "things": [  
    {  
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MotionSensor1",  
      "thingName": "MotionSensor1"  
    },  
    {  
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/TG_MS",  
      "thingName": "TG_MS"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengunggah Model](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [SearchThings](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk membuat tag untuk sumber daya

`tag-resource` Contoh berikut membuat tag untuk sumber daya yang ditentukan.

```
aws iotthingsgraph tag-resource \  
  --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/  
default/Room218" \  
  --tags key="Type",value="Residential"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Grafik AWS IoT Things Anda](#) di Panduan Pengguna AWS Grafik IoT Things.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **undeploy-system-instance**

Contoh kode berikut menunjukkan cara menggunakan `undeploy-system-instance`.

AWS CLI

Untuk menghapus instans sistem dari targetnya

`undeploy-system-instance` Contoh berikut menghapus instance sistem dari targetnya.

```
aws iotthingsgraph undeploy-system-instance \  
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room215"
```

Output:

```
{  
  "summary": {  
    "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room215",  
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/  
Room215",  
    "status": "PENDING_DELETE",  
    "target": "GREENGRASS",  
    "greengrassGroupName": "ThingsGraphGrnGr",  
    "createdAt": 1553189694.255,  
    "updatedAt": 1559344549.601,  
    "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",  
    "greengrassGroupVersionId": "731b371d-d644-4b67-ac64-3934e99b75d7"  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Manajemen Siklus Hidup untuk Entitas, Aliran, Sistem, dan Penerapan Grafik AWS IoT Things di Panduan Pengguna Grafik IoT Things.AWS](#)

- Untuk detail API, lihat [UndeploySystemInstance](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag untuk sumber daya

`untag-resource` Contoh berikut menghapus tag untuk sumber daya yang ditentukan.

```
aws iotthingsgraph untag-resource \  
  --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/  
default/Room218" \  
  --tag-keys "Type"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Grafik AWS IoT Things Anda](#) di Panduan Pengguna AWS Grafik IoT Things.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-flow-template

Contoh kode berikut menunjukkan cara menggunakan `update-flow-template`.

### AWS CLI

Untuk memperbarui alur

`update-flow-template` Contoh berikut memperbarui alur (alur kerja). Nilai `MyFlowDefinition` adalah GraphQL yang memodelkan aliran.

```
aws iotthingsgraph update-flow-template \  
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow" \  
  --definition "GraphQL query"
```

```
--definition language=GRAPHQL,text="MyFlowDefinition"
```

Output:

```
{
  "summary": {
    "createdAt": 1559248067.545,
    "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
    "revisionNumber": 2
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alur](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [UpdateFlowTemplate](#) di Referensi AWS CLI Perintah.

## update-system-template

Contoh kode berikut menunjukkan cara menggunakan `update-system-template`.

AWS CLI

Untuk memperbarui sistem

`update-system-template` Contoh berikut memperbarui sistem. Nilai `MySystemDefinition` adalah GraphQL yang memodelkan sistem.

```
aws iotthingsgraph update-system-template \
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem" \
  --definition language=GRAPHQL,text="MySystemDefinition"
```

Output:

```
{
  "summary": {
    "createdAt": 1559249776.254,
    "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/MySystem",
    "revisionNumber": 2
  }
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Membuat Sistem](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [UpdateSystemTemplate](#) di Referensi AWS CLI Perintah.

## upload-entity-definitions

Contoh kode berikut menunjukkan cara menggunakan `upload-entity-definitions`.

### AWS CLI

Untuk mengunggah definisi entitas

`upload-entity-definitions` Contoh berikut mengunggah definisi entitas ke namespace Anda. Nilai `MyEntityDefinitions` adalah GraphQL yang memodelkan entitas.

```
aws iotthingsgraph upload-entity-definitions \  
  --document language=GRAPHQL,text="MyEntityDefinitions"
```

Output:

```
{  
  "uploadId": "f6294f1e-b109-4bbe-9073-f451a2dda2da"  
}
```

Untuk informasi selengkapnya, lihat [Entitas Pemodelan](#) di Panduan Pengguna Grafik AWS IoT Things.

- Untuk detail API, lihat [UploadEntityDefinitions](#) di Referensi AWS CLI Perintah.

## AWS IoT Wireless contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS IoT Wireless.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-aws-account-with-partner-account**

Contoh kode berikut menunjukkan cara menggunakan `associate-aws-account-with-partner-account`.

AWS CLI

Untuk mengaitkan akun mitra dengan AWS akun Anda

`associate-aws-account-with-partner-account` Contoh berikut mengaitkan kredensi akun Sidewalk berikut dengan akun Anda. AWS

```
aws iotwireless associate-aws-account-with-partner-account \
  --sidewalk
  AmazonId="12345678901234",AppServerPrivateKey="a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
```

Output:

```
{
  "Sidewalk": {
    "AmazonId": "12345678901234",
    "AppServerPrivateKey":
    "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
  }
}
```

Untuk informasi selengkapnya, lihat [Integrasi Sidewalk Amazon untuk AWS IoT Core](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [AssociateAwsAccountWithPartnerAccount](#) di Referensi AWS CLI Perintah.

## associate-wireless-device-with-thing

Contoh kode berikut menunjukkan cara menggunakan `associate-wireless-device-with-thing`.

### AWS CLI

Untuk mengaitkan sesuatu ke perangkat nirkabel

`associate-wireless-device-with-thing` Contoh berikut mengaitkan sesuatu ke perangkat nirkabel Anda yang memiliki ID yang ditentukan.

```
aws iotwireless associate-wireless-device-with-thing \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --thing-arn "arn:aws:iot:us-east-1:123456789012:thing/MyIoTWirelessThing"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [AssociateWirelessDeviceWithThing](#) di Referensi AWS CLI Perintah.

## associate-wireless-gateway-with-certificate

Contoh kode berikut menunjukkan cara menggunakan `associate-wireless-gateway-with-certificate`.

### AWS CLI

Untuk mengaitkan sertifikat dengan gateway nirkabel

Berikut ini `associate-wireless-gateway-with-certificate` mengaitkan gateway nirkabel dengan sertifikat.

```
aws iotwireless associate-wireless-gateway-with-certificate \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --iot-certificate-id  
  "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
```

Output:



```
{
  "IotCertificateId":
  "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [AssociateWirelessGatewayWithCertificate](#) di Referensi AWS CLI Perintah.

## **associate-wireless-gateway-with-thing**

Contoh kode berikut menunjukkan cara menggunakan `associate-wireless-gateway-with-thing`.

### AWS CLI

Untuk mengaitkan sesuatu ke gateway nirkabel

`associate-wireless-gateway-with-thing` Contoh berikut mengaitkan sesuatu ke gateway nirkabel.

```
aws iotwireless associate-wireless-gateway-with-thing \
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
  --thing-arn "arn:aws:iot:us-east-1:123456789012:thing/MyIoTWirelessThing"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [AssociateWirelessGatewayWithThing](#) di Referensi AWS CLI Perintah.

## **create-destination**

Contoh kode berikut menunjukkan cara menggunakan `create-destination`.

### AWS CLI

Untuk membuat tujuan nirkabel IoT

`create-destination` Contoh berikut membuat tujuan untuk memetakan pesan perangkat ke aturan AWS IoT. Sebelum Anda menjalankan perintah ini, Anda harus telah membuat peran IAM yang memberikan AWS IoT Core LoRa untuk WAN izin yang diperlukan untuk mengirim data ke aturan IoT. AWS

```
aws iotwireless create-destination \  
  --name IoTWirelessDestination \  
  --expression-type RuleName \  
  --expression IoTWirelessRule \  
  --role-arn arn:aws:iam::123456789012:role/IoTWirelessDestinationRole
```

Output:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/  
IoTWirelessDestination",  
  "Name": "IoTWirelessDestination"  
}
```

Untuk informasi selengkapnya, lihat [Menambahkan tujuan ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreateDestination](#) di Referensi AWS CLI Perintah.

## **create-device-profile**

Contoh kode berikut menunjukkan cara menggunakan `create-device-profile`.

AWS CLI

Untuk membuat profil perangkat baru

`create-device-profile` Contoh berikut membuat profil perangkat nirkabel IoT baru.

```
aws iotwireless create-device-profile
```

Output:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-  
a1b2-3c45-67d8-e90fa1b2c34d",
```

```
"Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreateDeviceProfile](#) di Referensi AWS CLI Perintah.

## **create-service-profile**

Contoh kode berikut menunjukkan cara menggunakan `create-service-profile`.

### AWS CLI

Untuk membuat profil layanan baru

`create-service-profile` Contoh berikut membuat profil layanan nirkabel IoT baru.

```
aws iotwireless create-service-profile
```

Output:

```
{
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [CreateServiceProfile](#) di Referensi AWS CLI Perintah.

## **create-wireless-device**

Contoh kode berikut menunjukkan cara menggunakan `create-wireless-device`.

### AWS CLI

Untuk membuat perangkat nirkabel IoT

`create-wireless-device` Contoh berikut membuat sumber daya perangkat nirkabel dari tipe LoRa WAN.

```
aws iotwireless create-wireless-device \  
--cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "Description": "My LoRaWAN wireless device"  
  "DestinationName": "IoTWirelessDestination"  
  "LoRaWAN": {  
    "DeviceProfileId": "ab0c23d3-b001-45ef-6a01-2bc3de4f5333",  
    "ServiceProfileId": "fe98dc76-cd12-001e-2d34-5550432da100",  
    "OtaaV1_1": {  
      "AppKey": "3f4ca100e2fc675ea123f4eb12c4a012",  
      "JoinEui": "b4c231a359bc2e3d",  
      "NwkKey": "01c3f004a2d6efffe32c4eda14bcd2b4"  
    },  
    "DevEui": "ac12efc654d23fc2"  
  },  
  "Name": "SampleIoTWirelessThing"  
  "Type": LoRaWAN  
}
```

Output:

```
{  
  "Arn": "arn:aws:iotwireless:us-  
east-1:123456789012:WirelessDevice/1ffd32c8-8130-4194-96df-622f072a315f",  
  "Id": "1ffd32c8-8130-4194-96df-622f072a315f"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CreateWirelessDevice](#) di Referensi AWS CLI Perintah.

## **create-wireless-gateway-task-definition**

Contoh kode berikut menunjukkan cara menggunakan `create-wireless-gateway-task-definition`.

## AWS CLI

Untuk membuat definisi tugas gateway nirkabel

Berikut ini `create-wireless-gateway-task-definition` secara otomatis membuat tugas menggunakan definisi tugas ini untuk semua gateway dengan versi saat ini yang ditentukan.

```
aws iotwireless create-wireless-gateway-task-definition \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "AutoCreateTasks": true,  
  "Name": "TestAutoUpdate",  
  "Update": {  
    "UpdateDataSource" : "s3://cupsalphagafirmwarebin/station",  
    "UpdateDataRole" : "arn:aws:iam::001234567890:role/SDK_Test_Role",  
    "LoRaWAN" : {  
      "CurrentVersion" : {  
        "PackageVersion" : "1.0.0",  
        "Station" : "2.0.5",  
        "Model" : "linux"  
      },  
      "UpdateVersion" : {  
        "PackageVersion" : "1.0.1",  
        "Station" : "2.0.5",  
        "Model" : "minihub"  
      }  
    }  
  }  
}
```

Output:

```
{  
  "Id": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CreateWirelessGatewayTaskDefinition](#) di Referensi AWS CLI Perintah.

## **create-wireless-gateway-task**

Contoh kode berikut menunjukkan cara menggunakan `create-wireless-gateway-task`.

### AWS CLI

Untuk membuat tugas untuk gateway nirkabel

`create-wireless-gateway-task` Contoh berikut membuat tugas untuk gateway nirkabel.

```
aws iotwireless create-wireless-gateway-task \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --wireless-gateway-task-definition-id "aa000102-0304-b0cd-ef56-a1b23cde456a"
```

Output:

```
{  
  "WirelessGatewayTaskDefinitionId": "aa204003-0604-30fb-ac82-a4f95aaf450a",  
  "Status": "Success"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CreateWirelessGatewayTask](#) di Referensi AWS CLI Perintah.

## **create-wireless-gateway**

Contoh kode berikut menunjukkan cara menggunakan `create-wireless-gateway`.

### AWS CLI

Untuk membuat gateway nirkabel

`create-wireless-gateway` Contoh berikut membuat gateway perangkat LoRa WAN nirkabel.

```
aws iotwireless create-wireless-gateway \  
  --lorawan GatewayEui="a1b2c3d4567890ab",RfRegion="US915" \  
  --region us-east-1
```

```
--name "myFirstLoRaWANGateway" \  
--description "Using my first LoRaWAN gateway"
```

Output:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/12345678-  
a1b2-3c45-67d8-e90fa1b2c34d",  
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [CreateWirelessGateway](#) di Referensi AWS CLI Perintah.

## **delete-destination**

Contoh kode berikut menunjukkan cara menggunakan `delete-destination`.

AWS CLI

Untuk menghapus tujuan nirkabel IoT

`delete-destination` Contoh berikut menghapus sumber daya tujuan nirkabel dengan nama `IoTWirelessDestination` yang Anda buat.

```
aws iotwireless delete-destination \  
--name "IoTWirelessDestination"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan tujuan ke AWS IoT Core untuk LoRa WAN di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteDestination](#) di Referensi AWS CLI Perintah.

## **delete-device-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-device-profile`.

## AWS CLI

Untuk menghapus profil perangkat

`delete-device-profile` Contoh berikut menghapus profil perangkat dengan ID tertentu yang Anda buat.

```
aws iotwireless delete-device-profile \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteDeviceProfile](#) di Referensi AWS CLI Perintah.

## **delete-service-profile**

Contoh kode berikut menunjukkan cara menggunakan `delete-service-profile`.

## AWS CLI

Untuk menghapus profil layanan

`delete-service-profile` Contoh berikut menghapus profil layanan dengan ID tertentu yang Anda buat.

```
aws iotwireless delete-service-profile \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [DeleteServiceProfile](#) di Referensi AWS CLI Perintah.

## **delete-wireless-device**

Contoh kode berikut menunjukkan cara menggunakan `delete-wireless-device`.



## AWS CLI

Untuk menghapus perangkat nirkabel

`delete-wireless-device` Contoh berikut menghapus perangkat nirkabel yang memiliki ID yang ditentukan.

```
aws iotwireless delete-wireless-device \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteWirelessDevice](#) di Referensi AWS CLI Perintah.

## `delete-wireless-gateway-task-definition`

Contoh kode berikut menunjukkan cara menggunakan `delete-wireless-gateway-task-definition`.

## AWS CLI

Untuk menghapus definisi tugas gateway nirkabel

`delete-wireless-gateway-task-definition` Contoh berikut menghapus definisi tugas gateway nirkabel yang Anda buat dengan ID berikut.

```
aws iotwireless delete-wireless-gateway-task-definition \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteWirelessGatewayTaskDefinition](#) di Referensi AWS CLI Perintah.

## `delete-wireless-gateway-task`

Contoh kode berikut menunjukkan cara menggunakan `delete-wireless-gateway-task`.

## AWS CLI

Untuk menghapus tugas gateway nirkabel

`delete-wireless-gateway-task` Contoh berikut menghapus tugas gateway nirkabel yang memiliki ID tertentu.

```
aws iotwireless delete-wireless-gateway-task \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteWirelessGatewayTask](#) di Referensi AWS CLI Perintah.

## **delete-wireless-gateway**

Contoh kode berikut menunjukkan cara menggunakan `delete-wireless-gateway`.

## AWS CLI

Untuk menghapus gateway nirkabel

`delete-wireless-gateway` Contoh berikut menghapus gateway nirkabel yang memiliki ID tertentu.

```
aws iotwireless delete-wireless-gateway \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DeleteWirelessGateway](#) di Referensi AWS CLI Perintah.

## **disassociate-aws-account-from-partner-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-aws-account-from-partner-account`.

## AWS CLI

Untuk memisahkan akun mitra dari akun AWS

`disassociate-aws-account-from-partner-account` Contoh berikut memisahkan akun mitra dari akun Anda yang saat ini terkait AWS .

```
aws iotwireless disassociate-aws-account-from-partner-account \  
  --partner-account-id "12345678901234" \  
  --partner-type "Sidewalk"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DisassociateAwsAccountFromPartnerAccount](#) di Referensi AWS CLI Perintah.

## **disassociate-wireless-device-from-thing**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-wireless-device-from-thing`.

## AWS CLI

Untuk memisahkan benda dari perangkat nirkabel

`disassociate-wireless-device-from-thing` Contoh berikut memisahkan perangkat nirkabel dari hal yang saat ini terkait.

```
aws iotwireless disassociate-wireless-device-from-thing \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DisassociateWirelessDeviceFromThing](#) di Referensi AWS CLI Perintah.

## **disassociate-wireless-gateway-from-certificate**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-wireless-gateway-from-certificate`.

### AWS CLI

Untuk memisahkan sertifikat dari gateway nirkabel

Berikut ini `disassociate-wireless-gateway-from-certificate` memisahkan gateway nirkabel dari sertifikat yang saat ini terkait.

```
aws iotwireless disassociate-wireless-gateway-from-certificate \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DisassociateWirelessGatewayFromCertificate](#) di Referensi AWS CLI Perintah.

## **disassociate-wireless-gateway-from-thing**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-wireless-gateway-from-thing`.

### AWS CLI

Untuk memisahkan benda itu dari gateway nirkabel

`disassociate-wireless-gateway-from-thing` Contoh berikut memisahkan gateway nirkabel dari hal yang saat ini terkait.

```
aws iotwireless disassociate-wireless-gateway-from-thing \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan gateway dan perangkat nirkabel Anda ke AWS IoT Core for WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [DisassociateWirelessGatewayFromThing](#) di Referensi AWS CLI Perintah.

## get-destination

Contoh kode berikut menunjukkan cara menggunakan `get-destination`.

### AWS CLI

Untuk mendapatkan informasi tentang tujuan nirkabel IoT

`get-destination` Contoh berikut mendapatkan informasi tentang sumber daya tujuan dengan nama `IoTWirelessDestination` yang Anda buat.

```
aws iotwireless get-destination \  
  --name "IoTWirelessDestination"
```

Output:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/  
IoTWirelessDestination",  
  "Name": "IoTWirelessDestination",  
  "Expression": "IoTWirelessRule",  
  "ExpressionType": "RuleName",  
  "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"  
}
```

Untuk informasi selengkapnya, lihat [Menambahkan tujuan ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [GetDestination](#) di Referensi AWS CLI Perintah.

## get-device-profile

Contoh kode berikut menunjukkan cara menggunakan `get-device-profile`.

### AWS CLI

Untuk mendapatkan informasi tentang profil perangkat

`get-device-profile` Contoh berikut mendapatkan informasi tentang profil perangkat dengan ID tertentu yang Anda buat.

```
aws iotwireless get-device-profile \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

### Output:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-  
a1b2-3c45-67d8-e90fa1b2c34d",  
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",  
  "LoRaWAN": {  
    "MacVersion": "1.0.3",  
    "MaxDutyCycle": 10,  
    "Supports32BitFCnt": false,  
    "RegParamsRevision": "RP002-1.0.1",  
    "SupportsJoin": true,  
    "RfRegion": "US915",  
    "MaxEirp": 13,  
    "SupportsClassB": false,  
    "SupportsClassC": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [GetDeviceProfile](#) di Referensi AWS CLI Perintah.

## get-partner-account

Contoh kode berikut menunjukkan cara menggunakan `get-partner-account`.

### AWS CLI

Untuk mendapatkan informasi akun mitra

`get-partner-account` Contoh berikut mendapatkan informasi tentang akun Sidewalk Anda yang memiliki ID berikut.

```
aws iotwireless get-partner-account \  
  --partner-account-id "12345678901234" \  
  --partner-type "Sidewalk"
```

**Output:**

```
{
  "Sidewalk": {
    "AmazonId": "12345678901234",
    "Fingerprint":
"a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
  },
  "AccountLinked": false
}
```

Untuk informasi selengkapnya, lihat [Integrasi Sidewalk Amazon untuk AWS IoT Core](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [GetPartnerAccount](#) di Referensi AWS CLI Perintah.

**get-service-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `get-service-endpoint`.

**AWS CLI**

Untuk mendapatkan titik akhir layanan

`get-service-endpoint` Contoh berikut mendapatkan titik akhir khusus akun untuk protokol CUPS.

```
aws iotwireless get-service-endpoint
```

**Output:**

```
{
  "ServiceType": "CUPS",
  "ServiceEndpoint": "https://A1RMKZ37ACAG0T.cups.lorawan.us-east-1.amazonaws.com:443",
  "ServerTrust": "-----BEGIN CERTIFICATE-----\n
MIIESTCCAzGgAwIBAgITBn+UV4WH6Kx33rJTMlu8mYtWDTANBgkqhkiG9w0BAQsF\n
ADA5MQswCQYDVQQGEwJVUzEPMA0GA1UEChMGQW1hcm9uMRkwFwYDVQQDExBBbWF6\n
b24gUm9vdCBDQSAxMB4XDTE1MTAyMjAwMDAwMFoXDTE1MTAxOTAwMDAwMFowRjEL\n
MAKGA1UEBhMCVVMxMjAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAw\n
IDFCMQ8wDQYDVQQDEwZBbWF6b24wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK\n
AoIBAQCThZn3c68asg3Wuw6MLAd5tES6BIOsMzoKcG5b1PVo+sD0RrMd4f2AbnZ\n
-----"
```

```

cMzPa43j4wNxhplty6aUKk4T1qe9B0wKFjwK6zmxLlVYo7bHViXsPlJ6q0MpFge5\n
b1DP+18x+B26A0piiQ0uPkfyDyeR4xQghfj66Yo19V+emU3nazfvpFA+R0z6WoVm\n
B5x+F2pV8xeKNR7u6azDdU5YVX1TawpirmxRC1+WsAYmz6qP+z8ArDITC2FMVy2fw\n
0IjK0tEXc/VfmtTFch5+AfGYMGmqqvJ6LcXiAhqG5TI+Dr0RtM88k+8XUBCeQ8IG\n
KuANaL7TiItKZYxK1MMuTJtV9Ib1AgMBAAGjggE7MIIBNzASBgNVHRMBAf8ECDAG\n
AQH/AgEAMA4GA1UdDwEB/wQEAwIBhjAdBgNVHQ4EFgQUWaRmBlKge5WSPK0UByeW\n
dFv5PdAwHwYDVR0jBBgwFoAUhBjMhTTsvAyUlC4IWZzHshB0CggwewYIKwYBBQUH\n
AQEEbzBtMC8GCCsGAQUFBzABhiNodHRwOi8vb2NzcC5yb290Y2ExLmFtYXpvbnRy\n
dXN0LmNvbTA6BgggrBgEFBQcwAoYuaHR0cDovL2Nydc5yb290Y2ExLmFtYXpvbnRy\n
dXN0LmNvbS9yb290Y2ExLmNlclA/BgNVHR8EODA2MDSGmQAwHi5odHRwOi8vY3Js\n
LnJvb3RjYTEuYW1hem9udHJ1c3QuY29tL3Jvb3RjYTEuY3JsMBMGA1UdIAQMMAow\n
CAYGZ4EMAQIBMA0GCSqGSIb3DQEBwUAA4IBAQCfkr41u3nPo4FCH0TjY3NT0VI1\n
59Gt/a6ZiqyJEi+752+a1U5y6iAwYfmXss2lJwJFqMp2PphKg5625kXg8kP2CN5t\n
6G7bMQcT8C8xDZntYtd7WPD8UZiRKAJPBXa30/AbwuZe0GaFEQ8ugcYQgSn+IGBI\n
8/LwhBNTZTUVEWuCUUBVV18YtbAiPq3yXqMB480z+ctBWuZSkbvkNodPLamkB2g1\n
upRyzQ7qDn1X8nn8N8V7YJ6y68AtkHcNSRAnpTitxBKjtkPISLMVCx7i4hncxHZS\n
yLyKQXhw2W2Xs0qLeC1etA+jTGDK4UfLeC0SF7FSi8o5LL21L8IzApar2pR/\n
-----END CERTIFICATE-----\n"
}

```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetServiceEndpoint](#) di Referensi AWS CLI Perintah.

## get-service-profile

Contoh kode berikut menunjukkan cara menggunakan `get-service-profile`.

### AWS CLI

Untuk mendapatkan informasi tentang profil layanan

`get-service-profile` Contoh berikut mendapatkan informasi tentang profil layanan dengan ID tertentu yang Anda buat.

```

aws iotwireless get-service-profile \
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"

```

Output:

```
{
```



```
"Arn": "arn:aws:iotwireless:us-east-1:651419225604:ServiceProfile/538185bb-
d7e7-4b95-96a0-c51aa4a5b9a0",
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
  "LoRaWAN": {
    "HrAllowed": false,
    "NwkGeoLoc": false,
    "DrMax": 15,
    "UlBucketSize": 4096,
    "PrAllowed": false,
    "ReportDevStatusBattery": false,
    "DrMin": 0,
    "DlRate": 60,
    "AddGwMetadata": false,
    "ReportDevStatusMargin": false,
    "MinGwDiversity": 1,
    "RaAllowed": false,
    "DlBucketSize": 4096,
    "DevStatusReqFreq": 24,
    "TargetPer": 5,
    "UlRate": 60
  }
}
```

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [GetServiceProfile](#) di Referensi AWS CLI Perintah.

## get-wireless-device-statistics

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-device-statistics`.

### AWS CLI

Untuk mendapatkan informasi pengoperasian tentang perangkat nirkabel

`get-wireless-device-statistics` Contoh berikut mendapatkan informasi pengoperasian tentang perangkat nirkabel.

```
aws iotwireless get-wireless-device-statistics \
  --wireless-device-id "1ffd32c8-8130-4194-96df-622f072a315f"
```

Output:

```
{
  "WirelessDeviceId": "1ffd32c8-8130-4194-96df-622f072a315f"
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessDeviceStatistics](#) di Referensi AWS CLI Perintah.

## get-wireless-device

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-device`.

### AWS CLI

Untuk mendapatkan informasi tentang perangkat nirkabel

`get-wireless-device` Contoh berikut mencantumkan widget yang tersedia di AWS akun Anda.

```
aws iotwireless get-wireless-device \
  --identifier "1ffd32c8-8130-4194-96df-622f072a315f" \
  --identifier-type WirelessDeviceID
```

### Output:

```
{
  "Name": "myLoRaWANDevice",
  "ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/44b87eb4-9bce-423d-b5fc-973f5ecc358b",
  "DestinationName": "IoTWirelessDestination",
  "Id": "1ffd32c8-8130-4194-96df-622f072a315f",
  "ThingName": "44b87eb4-9bce-423d-b5fc-973f5ecc358b",
  "Type": "LoRaWAN",
  "LoRaWAN": {
    "DeviceProfileId": "ab0c23d3-b001-45ef-6a01-2bc3de4f5333",
    "ServiceProfileId": "fe98dc76-cd12-001e-2d34-5550432da100",
    "OtaaV1_1": {
      "AppKey": "3f4ca100e2fc675ea123f4eb12c4a012",
      "JoinEui": "b4c231a359bc2e3d",
      "NwkKey": "01c3f004a2d6efffe32c4eda14bcd2b4"
    },
    "DevEui": "ac12efc654d23fc2"
  }
}
```

```
  },
  "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessDevice/1fffd32c8-8130-4194-96df-622f072a315f",
  "Description": "My LoRaWAN wireless device"
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessDevice](#) di Referensi AWS CLI Perintah.

## **get-wireless-gateway-certificate**

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-gateway-certificate`.

### AWS CLI

Untuk mendapatkan ID sertifikat yang terkait dengan gateway nirkabel

`get-wireless-gateway-certificate` Contoh berikut mendapatkan ID sertifikat yang terkait dengan gateway nirkabel yang memiliki ID yang ditentukan.

```
aws iotwireless get-wireless-gateway-certificate \
  --id "6c44ab31-8b4d-407a-bed3-19b6c7cda551"
```

Output:

```
{
  "IotCertificateId":
  "8ea4aeae3db34c78cce75d9abd830356869ead6972997e0603e5fd032c804b6f"
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessGatewayCertificate](#) di Referensi AWS CLI Perintah.

## **get-wireless-gateway-firmware-information**

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-gateway-firmware-information`.

## AWS CLI

Untuk mendapatkan informasi firmware tentang gateway nirkabel

`get-wireless-gateway-firmware-information` Contoh berikut mendapatkan versi firmware dan informasi lain tentang gateway nirkabel.

```
aws iotwireless get-wireless-gateway-firmware-information \  
  --id "3039b406-5cc9-4307-925b-9948c63da25b"
```

Output:

```
{  
  "LoRaWAN" :{  
    "CurrentVersion" :{  
      "PackageVersion" : "1.0.0",  
      "Station" : "2.0.5",  
      "Model" : "linux"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessGatewayFirmwareInformation](#) di Referensi AWS CLI Perintah.

## `get-wireless-gateway-statistics`

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-gateway-statistics`.

## AWS CLI

Untuk mendapatkan informasi pengoperasian tentang gateway nirkabel

`get-wireless-gateway-statistics` Contoh berikut mendapatkan informasi operasi tentang gateway nirkabel.

```
aws iotwireless get-wireless-gateway-statistics \  
  --wireless-gateway-id "3039b406-5cc9-4307-925b-9948c63da25b"
```

Output:

```
{
  "WirelessGatewayId": "3039b406-5cc9-4307-925b-9948c63da25b"
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessGatewayStatistics](#) di Referensi AWS CLI Perintah.

## get-wireless-gateway-task-definition

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-gateway-task-definition`.

### AWS CLI

Untuk mendapatkan informasi tentang definisi tugas gateway nirkabel

`get-wireless-gateway-task-definition` Contoh berikut mendapatkan informasi tentang definisi tugas nirkabel dengan ID yang ditentukan.

```
aws iotwireless get-wireless-gateway-task-definition \
  --id "b7d3baad-25c7-35e7-a4e1-1683a0d61da9"
```

### Output:

```
{
  "AutoCreateTasks": true,
  "Name": "TestAutoUpdate",
  "Update": {
    "UpdateDataSource" : "s3://cupsalphagafirmwarebin/station",
    "UpdateDataRole" : "arn:aws:iam::001234567890:role/SDK_Test_Role",
    "LoRaWAN" : {
      "CurrentVersion" : {
        "PackageVersion" : "1.0.0",
        "Station" : "2.0.5",
        "Model" : "linux"
      },
      "UpdateVersion" : {
        "PackageVersion" : "1.0.1",
        "Station" : "2.0.5",

```

```
    "Model" : "minihub"  
  }  
} }  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessGatewayTaskDefinition](#) di Referensi AWS CLI Perintah.

## get-wireless-gateway-task

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-gateway-task`.

### AWS CLI

Untuk mendapatkan informasi tentang tugas gateway nirkabel

`get-wireless-gateway-task` Contoh berikut mendapatkan informasi tentang tugas gateway nirkabel dengan ID yang ditentukan.

```
aws iotwireless get-wireless-gateway-task \  
  --id "11693a46-6866-47c3-a031-c9a616e7644b"
```

Output:

```
{  
  "WirelessGatewayId": "6c44ab31-8b4d-407a-bed3-19b6c7cda551",  
  "WirelessGatewayTaskDefinitionId": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9",  
  "Status": "Success"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessGatewayTask](#) di Referensi AWS CLI Perintah.

## get-wireless-gateway

Contoh kode berikut menunjukkan cara menggunakan `get-wireless-gateway`.

## AWS CLI

Untuk mendapatkan informasi tentang gateway nirkabel

`get-wireless-gateway` Contoh berikut mendapatkan informasi tentang gateway nirkabel `myFirstLoRaWANGateway`.

```
aws iotwireless get-wireless-gateway \
  --identifier "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
  --identifier-type WirelessGatewayId
```

Output:

```
{
  "Description": "My first LoRaWAN gateway",
  "ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/a1b2c3d4-5678-90ab-cdef-12ab345c67de",
  "LoRaWAN": {
    "RfRegion": "US915",
    "GatewayEui": "a1b2c3d4567890ab"
  },
  "ThingName": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/6c44ab31-8b4d-407a-bed3-19b6c7cda551",
  "Name": "myFirstLoRaWANGateway"
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [GetWirelessGateway](#) di Referensi AWS CLI Perintah.

## **list-destinations**

Contoh kode berikut menunjukkan cara menggunakan `list-destinations`.

## AWS CLI

Untuk membuat daftar tujuan nirkabel

`list-destinations` Contoh berikut mencantumkan tujuan yang tersedia yang terdaftar ke AWS akun Anda.

```
aws iotwireless list-destinations
```

Output:

```
{
  "DestinationList": [
    {
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination",
      "Name": "IoTWirelessDestination",
      "Expression": "IoTWirelessRule",
      "Description": "Destination for messages processed using
IoTWirelessRule",
      "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
    },
    {
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination2",
      "Name": "IoTWirelessDestination2",
      "Expression": "IoTWirelessRule2",
      "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menambahkan tujuan ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListDestinations](#) di Referensi AWS CLI Perintah.

## list-device-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-device-profiles`.

### AWS CLI

Untuk membuat daftar profil perangkat

`list-device-profiles` Contoh berikut mencantumkan profil perangkat yang tersedia yang terdaftar ke AWS akun Anda.

```
aws iotwireless list-device-profiles
```



**Output:**

```
{
  "DeviceProfileList": [
    {
      "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-a1b2-3c45-67d8-e90fa1b2c34d"
    },
    {
      "Id": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/a1b2c3d4-5678-90ab-cdef-12ab345c67de"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListDeviceProfiles](#) di Referensi AWS CLI Perintah.

**list-partner-accounts**

Contoh kode berikut menunjukkan cara menggunakan `list-partner-accounts`.

**AWS CLI**

Untuk membuat daftar akun mitra

`list-partner-accounts` Contoh berikut mencantumkan akun mitra yang tersedia yang terkait dengan AWS akun Anda.

```
aws iotwireless list-partner-accounts
```

**Output:**

```
{
  "Sidewalk": [
    {
      "AmazonId": "78965678771228",

```

```

    "Fingerprint":
    "bd96d8ef66dbfd2160eb60e156849e82ad7018b8b73c1ba0b4fc65c32498ee35"
  },
  {
    "AmazonId": "89656787651228",
    "Fingerprint":
    "bc5e99e151c07be14be7e6603e4489c53f858b271213a36ebe3370777ba06e9b"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Integrasi Sidewalk Amazon untuk AWS IoT Core](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [ListPartnerAccounts](#) di Referensi AWS CLI Perintah.

## list-service-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-service-profiles`.

### AWS CLI

Untuk membuat daftar profil layanan

`list-service-profiles` Contoh berikut mencantumkan profil layanan yang tersedia yang terdaftar ke AWS akun Anda.

```
aws iotwireless list-service-profiles
```

Output:

```

{
  "ServiceProfileList": [
    {
      "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/538185bb-d7e7-4b95-96a0-c51aa4a5b9a0"
    },
    {
      "Id": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/ea8bc823-5d13-472e-8d26-9550737d8100"
    }
  ]
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Menambahkan profil ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListServiceProfiles](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang ditetapkan ke sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag yang ditetapkan ke sumber daya tujuan nirkabel.

```
aws iotwireless list-tags-for-resource \
  --resource-arn "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination"
```

Output:

```
{
  "Tags": [
    {
      "Value": "MyValue",
      "Key": "MyTag"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menjelaskan Inti AWS IoT Anda untuk sumber daya LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **list-wireless-devices**

Contoh kode berikut menunjukkan cara menggunakan `list-wireless-devices`.

## AWS CLI

Untuk daftar perangkat nirkabel yang tersedia

`list-wireless-devices` Contoh berikut mencantumkan perangkat nirkabel yang tersedia yang terdaftar ke AWS akun Anda.

```
aws iotwireless list-wireless-devices
```

Output:

```
{
  "WirelessDeviceList": [
    {
      "Name": "myLoRaWANDevice",
      "DestinationName": "IoTWirelessDestination",
      "Id": "1fffd32c8-8130-4194-96df-622f072a315f",
      "Type": "LoRaWAN",
      "LoRaWAN": {
        "DevEui": "ac12efc654d23fc2"
      },
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessDevice/1fffd32c8-8130-4194-96df-622f072a315f"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListWirelessDevices](#) di Referensi AWS CLI Perintah.

## `list-wireless-gateway-task-definitions`

Contoh kode berikut menunjukkan cara menggunakan `list-wireless-gateway-task-definitions`.

## AWS CLI

Untuk membuat daftar definisi tugas gateway nirkabel

`list-wireless-gateway-task-definitions` Contoh berikut mencantumkan definisi tugas gateway nirkabel yang tersedia yang terdaftar ke AWS akun Anda.

```
aws iotwireless list-wireless-gateway-task-definitions
```

Output:

```
{
  "TaskDefinitions": [
    {
      "Id": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9",
      "LoRaWAN" :
      {
        "CurrentVersion" :{
          "PackageVersion" : "1.0.0",
          "Station" : "2.0.5",
          "Model" : "linux"
        },
        "UpdateVersion" :{
          "PackageVersion" : "1.0.1",
          "Station" : "2.0.5",
          "Model" : "minihub"
        }
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembangan AWS IoT](#).

- Untuk detail API, lihat [ListWirelessGatewayTaskDefinitions](#) di Referensi AWS CLI Perintah.

## **list-wireless-gateways**

Contoh kode berikut menunjukkan cara menggunakan `list-wireless-gateways`.

AWS CLI

Untuk membuat daftar gateway nirkabel

`list-wireless-gateways` Contoh berikut mencantumkan gateway nirkabel yang tersedia di akun Anda AWS .

```
aws iotwireless list-wireless-gateways
```

Output:

```
{
  "WirelessGatewayList": [
    {
      "Description": "My first LoRaWAN gateway",
      "LoRaWAN": {
        "RfRegion": "US915",
        "GatewayEui": "dac632ebc01d23e4"
      },
      "Id": "3039b406-5cc9-4307-925b-9948c63da25b",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/3039b406-5cc9-4307-925b-9948c63da25b",
      "Name": "myFirstLoRaWANGateway"
    },
    {
      "Description": "My second LoRaWAN gateway",
      "LoRaWAN": {
        "RfRegion": "US915",
        "GatewayEui": "cda123fffe92ecd2"
      },
      "Id": "3285bdc7-5a12-4991-84ed-dadca65e342e",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/3285bdc7-5a12-4991-84ed-dadca65e342e",
      "Name": "mySecondLoRaWANGateway"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [ListWirelessGateways](#) di Referensi AWS CLI Perintah.

## send-data-to-wireless-device

Contoh kode berikut menunjukkan cara menggunakan `send-data-to-wireless-device`.

## AWS CLI

Untuk mengirim data ke perangkat nirkabel

`send-data-to-wireless-device` Contoh berikut mengirimkan bingkai data aplikasi yang didekripsi ke perangkat nirkabel.

```
aws iotwireless send-data-to-wireless-device \  
  --id "11aa5eae-2f56-4b8e-a023-b28d98494e49" \  
  --transmit-mode "1" \  
  --payload-data "SGVsbG8gVG8gRGV2c2lt" \  
  --wireless-metadata LoRaWAN={FPort=1}
```

Output:

```
{  
  MessageId: "6011dd36-0043d6eb-0072-0008"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembangan AWS IoT](#).

- Untuk detail API, lihat [SendDataToWirelessDevice](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menentukan kunci tag dan nilai untuk sumber daya

`tag-resource` Contoh berikut menandai tujuan nirkabel `IoTWirelessDestination` dengan kunci `MyTag` dan nilai `MyValue`.

```
aws iotwireless tag-resource \  
  --resource-arn "arn:aws:iotwireless:us-east-1:651419225604:Destination/  
IoTWirelessDestination" \  
  --tags Key="MyTag",Value="MyValue"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menjelaskan Inti AWS IoT Anda untuk sumber daya LoRa WAN di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## test-wireless-device

Contoh kode berikut menunjukkan cara menggunakan `test-wireless-device`.

### AWS CLI

Untuk menguji perangkat nirkabel

`test-wireless-device` Contoh berikut mengirimkan data uplink Hello ke perangkat dengan ID tertentu.

```
aws iotwireless test-wireless-device \  
  --id "11aa5eae-2f56-4b8e-a023-b28d98494e49"
```

Output:

```
{  
  Result: "Test succeeded. one message is sent with payload: hello"  
}
```

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [TestWirelessDevice](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus satu atau beberapa tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag `MyTag` dan nilainya dari tujuan nirkabel `IoTWirelessDestination`.



```
aws iotwireless untag-resource \  
  --resource-arn "arn:aws:iotwireless:us-east-1:123456789012:Destination/  
IoTWirelessDestination" \  
  --tag-keys "MyTag"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menjelaskan Inti AWS IoT Anda untuk sumber daya LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-destination

Contoh kode berikut menunjukkan cara menggunakan `update-destination`.

### AWS CLI

Untuk memperbarui properti tujuan

`update-destination` Contoh berikut memperbarui properti deskripsi tujuan nirkabel.

```
aws iotwireless update-destination \  
  --name "IoTWirelessDestination" \  
  --description "Destination for messages processed using IoTWirelessRule"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan tujuan ke AWS IoT Core untuk LoRa WAN di Panduan](#) Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateDestination](#) di Referensi AWS CLI Perintah.

## update-partner-account

Contoh kode berikut menunjukkan cara menggunakan `update-partner-account`.

### AWS CLI

Untuk memperbarui properti akun mitra

Berikut ini `update-partner-account` memperbarui `AppServerPrivateKey` untuk akun yang memiliki ID yang ditentukan.

```
aws iotwireless update-partner-account \  
  --partner-account-id "78965678771228" \  
  --partner-type "Sidewalk" \  
  --sidewalk  
AppServerPrivateKey="f798ab4899346a88599180fee9e14fa1ada7b6df989425b7c6d2146dd6c815bb"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Integrasi Sidewalk Amazon untuk AWS IoT Core](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdatePartnerAccount](#) di Referensi AWS CLI Perintah.

## update-wireless-device

Contoh kode berikut menunjukkan cara menggunakan `update-wireless-device`.

### AWS CLI

Untuk memperbarui properti perangkat nirkabel

`update-wireless-device` Contoh berikut memperbarui properti perangkat nirkabel yang terdaftar ke AWS akun Anda.

```
aws iotwireless update-wireless-device \  
  --id "1ffd32c8-8130-4194-96df-622f072a315f" \  
  --destination-name IoTWirelessDestination2 \  
  --description "Using my first LoRaWAN device"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa](#) di Panduan Pengembang AWS IoT.

- Untuk detail API, lihat [UpdateWirelessDevice](#) di Referensi AWS CLI Perintah.

## update-wireless-gateway

Contoh kode berikut menunjukkan cara menggunakan `update-wireless-gateway`.

## AWS CLI

Untuk memperbarui gateway nirkabel

`update-wireless-gateway` Contoh berikut memperbarui deskripsi gateway nirkabel Anda.

```
aws iotwireless update-wireless-gateway \  
  --id "3285bdc7-5a12-4991-84ed-dadca65e342e" \  
  --description "Using my LoRaWAN gateway"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghubungkan perangkat dan gateway ke AWS IoT Core untuk WAN LoRa di Panduan Pengembang AWS IoT](#).

- Untuk detail API, lihat [UpdateWirelessGateway](#) di Referensi AWS CLI Perintah.

## Contoh Amazon IVS menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan Amazon IVS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-get-channel**

Contoh kode berikut menunjukkan cara menggunakan `batch-get-channel`.

## AWS CLI

Untuk mendapatkan informasi konfigurasi saluran tentang beberapa saluran

`batch-get-channel` Contoh berikut mencantumkan informasi tentang saluran yang ditentukan.

```
aws ivs batch-get-channel \
  --arns arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
  arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl
```

Output:

```
{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "authorized": false,
      "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
      "insecureIngest": false,
      "latencyMode": "LOW",
      "name": "channel-1",
      "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel-1.abcdEFGH.m3u8",
      "preset": "",
      "playbackRestrictionPolicyArn": "",
      "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",
      "srt": {
        "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
        "passphrase": "AB1C2defGHijklMNop3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
      },
      "tags": {},
      "type": "STANDARD"
    },
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
      "authorized": false,
      "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
      "insecureIngest": true,
      "latencyMode": "LOW",
      "name": "channel-2",
```

```

        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/
api/video/v1/us-west-2.123456789012.channel-2.abcdEFGH.m3u8",
        "preset": "",
        "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABCdef34ghIJ",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
"BA1C2defGHijklMNop3PqQRstUvwxyzABCDefghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "tags": {},
        "type": "STANDARD"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [BatchGetChannel](#) di Referensi AWS CLI Perintah.

## batch-get-stream-key

Contoh kode berikut menunjukkan cara menggunakan `batch-get-stream-key`.

### AWS CLI

Untuk mendapatkan informasi tentang beberapa tombol aliran

`batch-get-stream-key` Contoh berikut mendapat informasi tentang kunci aliran yang ditentukan.

```

aws ivs batch-get-stream-key \
  --arns arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh \
  arn:aws:ivs:us-west-2:123456789012:stream-key/skSKIJKLmnop

```

### Output:

```

{
  "streamKeys": [
    {

```

```

    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  },
  {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKIJKLmnop",
    "value": "sk_us-west-2_abcdABCDefgh_567890ghijkl",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [BatchGetStreamKey](#) di Referensi AWS CLI Perintah.

## batch-start-viewer-session-revocation

Contoh kode berikut menunjukkan cara menggunakan `batch-start-viewer-session-revocation`.

### AWS CLI

Untuk mencabut sesi penampil untuk beberapa pasangan Channel-ARN dan Viewer-ID

`batch-start-viewer-session-revocation` Contoh berikut melakukan pencabutan sesi pada beberapa pasangan Channel-ARN dan Viewer-ID secara bersamaan. Permintaan dapat selesai secara normal tetapi mengembalikan nilai di bidang kesalahan jika pemanggil tidak memiliki izin untuk mencabut sesi tertentu.

```

aws ivs batch-start-viewer-session-revocation \
  --viewer-sessions '[{"channelArn":"arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh1","viewerId":"abcdefg1","viewerSessionVersionsLessThanOrEqualTo":1234567890}, \
  [{"channelArn":"arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh2","viewerId":"abcdefg2","viewerSessionVersionsLessThanOrEqualTo":1234567890}]'

```

Output:

```
{
  "errors": [
    {
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh1",
      "viewerId": "abcdefg1",
      "code": "403",
      "message": "not authorized",
    },
    {
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh2",
      "viewerId": "abcdefg2",
      "code": "403",
      "message": "not authorized",
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Saluran Privat](#) dalam Panduan Pengguna Amazon Interactive Video Service.

- Untuk detail API, lihat [BatchStartViewerSessionRevocation](#) di Referensi AWS CLI Perintah.

## create-channel

Contoh kode berikut menunjukkan cara menggunakan `create-channel`.

### AWS CLI

Contoh 1: Untuk membuat saluran tanpa rekaman

`create-channel` Contoh berikut membuat saluran baru dan kunci aliran terkait untuk memulai streaming.

```
aws ivs create-channel \
  --name "test-channel" \
  --no-insecure-ingest
```

Output:

```
{
```

```

"channel": {
  "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
  "authorized": false,
  "name": "test-channel",
  "latencyMode": "LOW",
  "playbackRestrictionPolicyArn": "",
  "recordingConfigurationArn": "",
  "srt": {
    "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
    "passphrase":
"AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
  },
  "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
  "insecureIngest": false,
  "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
  "preset": "",
  "tags": {},
  "type": "STANDARD"
},
"streamKey": {
  "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",
  "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
  "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
  "tags": {}
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

Contoh 2: Untuk membuat saluran dengan perekaman diaktifkan, menggunakan RecordingConfiguration sumber daya yang ditentukan oleh ARN-nya

`create-channel` Contoh berikut membuat saluran baru dan kunci streaming terkait untuk memulai streaming, dan mengatur perekaman untuk saluran tersebut.

```

aws ivs create-channel \
  --name test-channel-with-recording \
  --insecure-ingest \
  --recording-configuration-arn "arn:aws:ivs:us-west-2:123456789012:recording-
configuration/ABCD12cdEFgh"

```



## Output:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-recording",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "",
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"BA1C2defGHijklMN03PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": true,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {},
    "type": "STANDARD"
  },
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3 di Panduan Pengguna Latensi Rendah IVS](#).

Contoh 3: Untuk membuat saluran dengan kebijakan pembatasan pemutaran yang ditentukan oleh ARN

`create-channel` Contoh berikut membuat saluran baru dan kunci streaming terkait untuk memulai streaming, dan menyiapkan kebijakan pembatasan pemutaran untuk saluran tersebut.

```
aws ivs create-channel \
```

```
--name test-channel-with-playback-restriction-policy \
--insecure-ingest \
--playback-restriction-policy-arn "arn:aws:ivs:us-west-2:123456789012:playback-
restriction-policy/ABcdef34ghIJ"
```

### Output:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-playback-restriction-policy",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
    "recordingConfigurationArn": "",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2edfGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": true,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {},
    "type": "STANDARD"
  },
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [CreateChannel](#) di Referensi AWS CLI Perintah.

## create-playback-restriction-policy

Contoh kode berikut menunjukkan cara menggunakan `create-playback-restriction-policy`.

### AWS CLI

Untuk membuat kebijakan pembatasan pemutaran

`create-playback-restriction-policy` Contoh berikut membuat kebijakan restriction pemutaran baru.

```
aws ivs create-playback-restriction-policy \
  --name "test-playback-restriction-policy" \
  --enable-strict-origin-enforcement \
  --tags "key1=value1, key2=value2" \
  --allowed-countries US MX \
  --allowed-origins https://www.website1.com https://www.website2.com
```

Output:

```
{
  "playbackRestrictionPolicy": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ",
    "allowedCountries": [
      "US",
      "MX"
    ],
    "allowedOrigins": [
      "https://www.website1.com",
      "https://www.website2.com"
    ],
    "enableStrictOriginEnforcement": true,
    "name": "test-playback-restriction-policy",
    "tags": {
      "key1": "value1",
      "key2": "value2"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [CreatePlaybackRestrictionPolicy](#) di Referensi AWS CLI Perintah.

## create-recording-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-recording-configuration`.

### AWS CLI

Untuk membuat sumber `RecordingConfiguration` daya

`create-recording-configuration` Contoh berikut membuat `RecordingConfiguration` sumber daya untuk mengaktifkan perekaman ke Amazon S3.

```
aws ivs create-recording-configuration \
  --name "test-recording-config" \
  --recording-reconnect-window-seconds 60 \
  --tags "key1=value1, key2=value2" \
  --rendition-configuration renditionSelection="CUSTOM",renditions="HD" \
  --thumbnail-configuration
recordingMode="INTERVAL",targetIntervalSeconds=1,storage="LATEST",resolution="LOWEST_RESOLUTION" \
  --destination-configuration s3={bucketName=demo-recording-bucket}
```

### Output:

```
{
  "recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCdef34ghIJ",
    "name": "test-recording-config",
    "destinationConfiguration": {
      "s3": {
        "bucketName": "demo-recording-bucket"
      }
    },
    "state": "CREATING",
    "tags": {
      "key1": "value1",
      "key2": "value2"
    },
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
```

```

        "targetIntervalSeconds": 1,
        "resolution": "LOWEST_RESOLUTION",
        "storage": [
            "LATEST"
        ]
    },
    "recordingReconnectWindowSeconds": 60,
    "renditionConfiguration": {
        "renditionSelection": "CUSTOM",
        "renditions": [
            "HD"
        ]
    }
}
}

```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateRecordingConfiguration](#) di Referensi AWS CLI Perintah.

## create-stream-key

Contoh kode berikut menunjukkan cara menggunakan `create-stream-key`.

### AWS CLI

Untuk membuat kunci aliran

`create-stream-key` Contoh berikut membuat kunci aliran untuk ARN tertentu (Amazon Resource Name).

```

aws ivs create-stream-key \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh

```

Output:

```

{
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}

```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [CreateStreamKey](#) di Referensi AWS CLI Perintah.

## **delete-channel**

Contoh kode berikut menunjukkan cara menggunakan `delete-channel`.

### AWS CLI

Untuk menghapus saluran dan kunci aliran terkait

`delete-channel` Contoh berikut menghapus saluran dengan ARN yang ditentukan (Amazon Resource Name).

```
aws ivs delete-channel \  
  --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [DeleteChannel](#) di Referensi AWS CLI Perintah.

## **delete-playback-key-pair**

Contoh kode berikut menunjukkan cara menggunakan `delete-playback-key-pair`.

### AWS CLI

Untuk menghapus key pair playback yang ditentukan

`delete-playback-key-pair` Contoh berikut mengembalikan sidik jari dari key pair yang ditentukan.

```
aws ivs delete-playback-key-pair \  
  --arn arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan Saluran Privat](#) dalam Panduan Pengguna Amazon Interactive Video Service.

- Untuk detail API, lihat [DeletePlaybackKeyPair](#) di Referensi AWS CLI Perintah.

## **delete-playback-restriction-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-playback-restriction-policy`.

### AWS CLI

Untuk menghapus kebijakan pembatasan pemutaran

`delete-playback-restriction-policy` Contoh berikut menghapus kebijakan pemutaran ulang dengan kebijakan yang ditentukan ARN (Nama Sumber Daya Amazon).

```
aws ivs delete-playback-restriction-policy \  
  --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
  ABCdef34ghIJ"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [DeletePlaybackRestrictionPolicy](#) di Referensi AWS CLI Perintah.

## **delete-recording-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-recording-configuration`.

### AWS CLI

Untuk menghapus `RecordingConfiguration` sumber daya yang ditentukan oleh ARN-nya

`delete-recording-configuration` Contoh berikut menghapus `RecordingConfiguration` sumber daya dengan ARN yang ditentukan.

```
aws ivs delete-recording-configuration \  
  --arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCdef34ghIJ"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteRecordingConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-stream-key**

Contoh kode berikut menunjukkan cara menggunakan `delete-stream-key`.

### AWS CLI

Untuk menghapus kunci aliran

`delete-stream-key` Contoh berikut menghapus kunci aliran untuk ARN tertentu (Amazon Resource Name), sehingga tidak dapat lagi digunakan untuk streaming.

```
aws ivs delete-stream-key \  
  --arn arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [DeleteStreamKey](#) di Referensi AWS CLI Perintah.

## **get-channel**

Contoh kode berikut menunjukkan cara menggunakan `get-channel`.

### AWS CLI

Untuk mendapatkan informasi konfigurasi saluran

`get-channel` Contoh berikut mendapatkan konfigurasi saluran untuk saluran tertentu ARN (Amazon Resource Name).

```
aws ivs get-channel \  
  --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Output:



```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "channel-1",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "",
    "preset": "",
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [GetChannel](#) di Referensi AWS CLI Perintah.

## get-playback-key-pair

Contoh kode berikut menunjukkan cara menggunakan `get-playback-key-pair`.

### AWS CLI

Untuk mendapatkan key pair playback yang ditentukan

`get-playback-key-pair` Contoh berikut mengembalikan sidik jari dari key pair yang ditentukan.

```
aws ivs get-playback-key-pair \
  --arn arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh
```

**Output:**

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
    "name": "my-playback-key",
    "fingerprint": "0a:1b:2c:ab:cd:ef:34:56:70:b1:b2:71:01:2a:a3:72",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Saluran Privat](#) dalam Panduan Pengguna Amazon Interactive Video Service.

- Untuk detail API, lihat [GetPlaybackKeyPair](#) di Referensi AWS CLI Perintah.

**get-playback-restriction-policy**

Contoh kode berikut menunjukkan cara menggunakan `get-playback-restriction-policy`.

**AWS CLI**

Untuk mendapatkan informasi konfigurasi kebijakan pembatasan pemutaran

`get-playback-restriction-policy` Contoh berikut mendapatkan konfigurasi kebijakan pembatasan pemutaran dengan kebijakan yang ditentukan ARN (Nama Sumber Daya Amazon).

```
aws ivs get-playback-restriction-policy \
  --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
  ABcdef34ghIJ"
```

**Output:**

```
{
  "playbackRestrictionPolicy": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
  ABcdef34ghIJ",
    "allowedCountries": [
      "US",
      "MX"
    ],
    "allowedOrigins": [
```

```

        "https://www.website1.com",
        "https://www.website2.com"
    ],
    "enableStrictOriginEnforcement": true,
    "name": "test-playback-restriction-policy",
    "tags": {
        "key1": "value1",
        "key2": "value2"
    }
}
}

```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [GetPlaybackRestrictionPolicy](#) di Referensi AWS CLI Perintah.

## get-recording-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-recording-configuration`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber `RecordingConfiguration` daya

`get-recording-configuration` Contoh berikut mendapatkan informasi tentang `RecordingConfiguration` sumber daya untuk ARN yang ditentukan.

```

aws ivs get-recording-configuration \
  --arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABcdef34ghIJ"

```

Output:

```

{
  "recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABcdef34ghIJ",
    "destinationConfiguration": {
      "s3": {
        "bucketName": "demo-recording-bucket"
      }
    },
    "name": "test-recording-config",

```

```
    "recordingReconnectWindowSeconds": 60,
    "state": "ACTIVE",
    "tags": {
      "key1" : "value1",
      "key2" : "value2"
    },
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
      "targetIntervalSeconds": 1,
      "resolution": "LOWEST_RESOLUTION",
      "storage": [
        "LATEST"
      ]
    },
    "renditionConfiguration": {
      "renditionSelection": "CUSTOM",
      "renditions": [
        "HD"
      ]
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetRecordingConfiguration](#) di Referensi AWS CLI Perintah.

## get-stream-key

Contoh kode berikut menunjukkan cara menggunakan `get-stream-key`.

### AWS CLI

Untuk mendapatkan informasi tentang aliran

`get-stream-key` Contoh berikut mendapatkan informasi tentang kunci aliran yang ditentukan.

```
aws ivs get-stream-key \
  --arn arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh --region=us-
west-2
```

Output:

```
{
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [GetStreamKey](#) di Referensi AWS CLI Perintah.

## get-stream-session

Contoh kode berikut menunjukkan cara menggunakan `get-stream-session`.

### AWS CLI

Untuk mendapatkan metadata untuk aliran tertentu

`get-stream-session` Contoh berikut mendapatkan konfigurasi metadata untuk saluran yang ditentukan ARN (Nama Sumber Daya Amazon) dan aliran yang ditentukan; jika StreamID tidak disediakan, aliran terbaru untuk saluran dipilih.

```
aws ivs get-stream-session \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
  --stream-id "mystream"
```

Output:

```
{
  "streamSession": {
    "streamId": "mystream1",
    "startTime": "2023-06-26T19:09:28+00:00",
    "channel": {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "name": "mychannel",
      "latencyMode": "LOW",
      "type": "STANDARD",
    }
  }
}
```

```
    "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABcdef34ghIJ",
    "srt": {
        "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
        "passphrase":
"AB1C2defGHijklMN03PqQRstUvwxyzABCDefghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "playbackUrl": "url-string",
    "authorized": false,
    "insecureIngest": false,
    "preset": ""
},
"ingestConfiguration": {
    "video": {
        "avcProfile": "Baseline",
        "avcLevel": "4.2",
        "codec": "avc1.42C02A",
        "encoder": "Lavf58.45.100",
        "targetBitrate": 8789062,
        "targetFramerate": 60,
        "videoHeight": 1080,
        "videoWidth": 1920
    },
    "audio": {
        "codec": "mp4a.40.2",
        "targetBitrate": 46875,
        "sampleRate": 8000,
        "channels": 2
    }
},
"recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABcdef34ghIJ",
    "name": "test-recording-config",
    "destinationConfiguration": {
        "s3": {
            "bucketName": "demo-recording-bucket"
        }
    },
    "state": "ACTIVE",
    "tags": {
        "key1": "value1",
        "key2": "value2"
    }
}
```

```
    },
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
      "targetIntervalSeconds": 1,
      "resolution": "LOWEST_RESOLUTION",
      "storage": [
        "LATEST"
      ]
    },
    "recordingReconnectWindowSeconds": 60,
    "renditionConfiguration": {
      "renditionSelection": "CUSTOM",
      "renditions": [
        "HD"
      ]
    }
  },
  "truncatedEvents": [
    {
      "name": "Recording Start",
      "type": "IVS Recording State Change",
      "eventTime": "2023-06-26T19:09:35+00:00"
    },
    {
      "name": "Stream Start",
      "type": "IVS Stream State Change",
      "eventTime": "2023-06-26T19:09:34+00:00"
    },
    {
      "name": "Session Created",
      "type": "IVS Stream State Change",
      "eventTime": "2023-06-26T19:09:28+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [GetStreamSession](#) di Referensi AWS CLI Perintah.

## get-stream

Contoh kode berikut menunjukkan cara menggunakan `get-stream`.

### AWS CLI

Untuk mendapatkan informasi tentang aliran

`get-stream` Contoh berikut mendapatkan informasi tentang aliran untuk saluran yang ditentukan.

```
aws ivs get-stream \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Output:

```
{
  "stream": {
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "startTime": "2020-05-05T21:55:38Z",
    "state": "LIVE",
    "health": "HEALTHY",
    "streamId": "st-ABCDefghij01234KLMN5678",
    "viewerCount": 1
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [GetStream](#) di Referensi AWS CLI Perintah.

## import-playback-key-pair

Contoh kode berikut menunjukkan cara menggunakan `import-playback-key-pair`.

### AWS CLI

Untuk mengimpor bagian publik dari key pair baru



`import-playback-key-pair` Contoh berikut mengimpor kunci publik yang ditentukan (ditentukan sebagai string dalam format PEM) dan mengembalikan arn dan sidik jari dari key pair baru.

```
aws ivs import-playback-key-pair \  
  --name "my-playback-key" \  
  --public-key-material "G11bnQx0TA3BgNVBAMMMFdoeSBhcmUgeW91IGR1..."
```

Output:

```
{  
  "keyPair": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",  
    "name": "my-playback-key",  
    "fingerprint": "0a:1b:2c:ab:cd:ef:34:56:70:b1:b2:71:01:2a:a3:72",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Saluran Privat](#) dalam Panduan Pengguna Amazon Interactive Video Service.

- Untuk detail API, lihat [ImportPlaybackKeyPair](#) di Referensi AWS CLI Perintah.

## list-channels

Contoh kode berikut menunjukkan cara menggunakan `list-channels`.

AWS CLI

Contoh 1: Untuk mendapatkan informasi ringkasan tentang semua saluran

`list-channels` Contoh berikut mencantumkan semua saluran untuk AWS akun Anda.

```
aws ivs list-channels
```

Output:

```
{  
  "channels": [  

```

```

    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "name": "channel-1",
      "latencyMode": "LOW",
      "authorized": false,
      "insecureIngest": false,
      "preset": "",
      "playbackRestrictionPolicyArn": "",
      "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
      "tags": {},
      "type": "STANDARD"
    },
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
      "name": "channel-2",
      "latencyMode": "LOW",
      "authorized": false,
      "preset": "",
      "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABCdef34ghIJ",
      "recordingConfigurationArn": "",
      "tags": {},
      "type": "STANDARD"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

Contoh 2: Untuk mendapatkan informasi ringkasan tentang semua saluran, disaring oleh RecordingConfiguration ARN yang ditentukan

`list-channels` Contoh berikut mencantumkan semua saluran untuk AWS akun Anda, yang terkait dengan RecordingConfiguration ARN yang ditentukan.

```

aws ivs list-channels \
  --filter-by-recording-configuration-arn "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh"

```

Output:

```
{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "name": "channel-1",
      "latencyMode": "LOW",
      "authorized": false,
      "insecureIngest": false,
      "preset": "",
      "playbackRestrictionPolicyArn": "",
      "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
      "tags": {},
      "type": "STANDARD"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3 di Panduan Pengguna Latensi Rendah IVS](#).

Contoh 3: Untuk mendapatkan informasi ringkasan tentang semua saluran, disaring oleh PlaybackRestrictionPolicy ARN yang ditentukan

`list-channels` Contoh berikut mencantumkan semua saluran untuk AWS akun Anda, yang terkait dengan PlaybackRestrictionPolicy ARN yang ditentukan.

```
aws ivs list-channels \
  --filter-by-playback-restriction-policy-arn "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ"
```

Output:

```
{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
      "name": "channel-2",
      "latencyMode": "LOW",
      "authorized": false,
      "preset": "",

```

```

        "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
        "recordingConfigurationArn": "",
        "tags": {},
        "type": "STANDARD"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [ListChannels](#) di Referensi AWS CLI Perintah.

## list-playback-key-pairs

Contoh kode berikut menunjukkan cara menggunakan `list-playback-key-pairs`.

### AWS CLI

Untuk mendapatkan informasi ringkasan tentang semua pasangan kunci pemutaran

`list-playback-key-pairs` Contoh berikut mengembalikan informasi tentang semua pasangan kunci.

```
aws ivs list-playback-key-pairs
```

Output:

```

{
  "keyPairs": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
      "name": "test-key-0",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/ijkl5678mnop",
      "name": "test-key-1",
      "tags": {}
    }
  ]
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Menyiapkan Saluran Privat](#) dalam Panduan Pengguna Amazon Interactive Video Service.

- Untuk detail API, lihat [ListPlaybackKeyPairs](#) di Referensi AWS CLI Perintah.

## **list-playback-restriction-policies**

Contoh kode berikut menunjukkan cara menggunakan `list-playback-restriction-policies`.

### AWS CLI

Untuk mendapatkan informasi ringkasan tentang semua kebijakan pembatasan pemutaran

`list-playback-restriction-policies` Contoh berikut mencantumkan semua kebijakan pembatasan pemutaran untuk AWS akun Anda.

```
aws ivs list-playback-restriction-policies
```

### Output:

```
{
  "playbackRestrictionPolicies": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABCdef34ghIJ",
      "allowedCountries": [
        "US",
        "MX"
      ],
      "allowedOrigins": [
        "https://www.website1.com",
        "https://www.website2.com"
      ],
      "enableStrictOriginEnforcement": true,
      "name": "test-playback-restriction-policy",
      "tags": {
        "key1": "value1",
        "key2": "value2"
      }
    }
  ]
}
```

```
    }  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [ListPlaybackRestrictionPolicies](#) di Referensi AWS CLI Perintah.

## list-recording-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-recording-configurations`.

### AWS CLI

Untuk membuat daftar semua `RecordingConfiguration` sumber daya yang dibuat di akun ini

`list-recording-configurations` Contoh berikut mendapatkan informasi tentang semua `RecordingConfiguration` sumber daya di akun Anda.

```
aws ivs list-recording-configurations
```

### Output:

```
{  
  "recordingConfigurations": [  
    {  
      "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/  
ABcdef34ghIJ",  
      "name": "test-recording-config-1",  
      "destinationConfiguration": {  
        "s3": {  
          "bucketName": "demo-recording-bucket-1"  
        }  
      },  
      "state": "ACTIVE",  
      "tags": {}  
    },  
    {  
      "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/  
CD12abcdGHIJ",
```

```

        "name": "test-recording-config-2",
        "destinationConfiguration": {
            "s3": {
                "bucketName": "demo-recording-bucket-2"
            }
        },
        "state": "ACTIVE",
        "tags": {}
    }
]
}

```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListRecordingConfigurations](#) di Referensi AWS CLI Perintah.

## list-stream-keys

Contoh kode berikut menunjukkan cara menggunakan `list-stream-keys`.

### AWS CLI

Untuk mendapatkan daftar tombol aliran

`list-stream-keys` Contoh berikut mencantumkan semua kunci aliran untuk ARN tertentu (Amazon Resource Name).

```

aws ivs list-stream-keys \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh

```

Output:

```

{
  "streamKeys": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "tags": {}
    }
  ]
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran di Panduan Pengguna Latensi Rendah IVS](#).

- Untuk detail API, lihat [ListStreamKeys](#) di Referensi AWS CLI Perintah.

## list-stream-sessions

Contoh kode berikut menunjukkan cara menggunakan `list-stream-sessions`.

### AWS CLI

Untuk mendapatkan ringkasan aliran saat ini dan sebelumnya untuk saluran tertentu di wilayah saat ini AWS

`list-stream-sessions` Contoh berikut melaporkan informasi ringkasan untuk aliran untuk saluran tertentu ARN (Nama Sumber Daya Amazon).

```
aws ivs list-stream-sessions \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
  --max-results 25 \
  --next-token ""
```

### Output:

```
{
  "nextToken": "set-2",
  "streamSessions": [
    {
      "startTime": 1641578182,
      "endTime": 1641579982,
      "hasErrorEvent": false,
      "streamId": "mystream"
    }
    ...
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.



- Untuk detail API, lihat [ListStreamSessions](#) di Referensi AWS CLI Perintah.

## list-streams

Contoh kode berikut menunjukkan cara menggunakan `list-streams`.

### AWS CLI

Untuk mendapatkan daftar streaming langsung dan negara bagiannya

`list-streams` Contoh berikut mencantumkan semua streaming langsung untuk AWS akun Anda.

```
aws ivs list-streams
```

Output:

```
{
  "streams": [
    {
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "state": "LIVE",
      "health": "HEALTHY",
      "streamId": "st-ABCDEFghij01234KLMN5678",
      "viewerCount": 1
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [ListStreams](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan semua tag untuk AWS sumber daya (misalnya: saluran, kunci aliran)

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk ARN sumber daya tertentu (Amazon Resource Name).

```
aws ivs list-tags-for-resource \  
  --resource-arn arn:aws:ivs:us-west-2:12345689012:channel/abcdABCDefgh
```

Output:

```
{  
  "tags":  
  {  
    "key1": "value1",  
    "key2": "value2"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menandai](#) di Referensi API Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-metadata

Contoh kode berikut menunjukkan cara menggunakan `put-metadata`.

### AWS CLI

Untuk menyisipkan metadata ke aliran aktif untuk saluran tertentu

`put-metadata` Contoh berikut menyisipkan metadata yang diberikan ke dalam aliran untuk saluran yang ditentukan.

```
aws ivs put-metadata \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --metadata '{"my": "metadata"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [PutMetadata](#) di Referensi AWS CLI Perintah.

## start-viewer-session-revocation

Contoh kode berikut menunjukkan cara menggunakan `start-viewer-session-revocation`.

### AWS CLI

Untuk mencabut sesi penampil untuk pasangan Channel-ARN dan Viewer-ID yang diberikan

`start-viewer-session-revocation` Contoh berikut memulai proses pencabutan sesi penampil yang terkait dengan ARN saluran tertentu dan ID penampil, hingga dan termasuk nomor versi sesi yang ditentukan. Jika versi tidak disediakan, defaultnya ke 0.

```
aws ivs batch-start-viewer-session-revocation \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --viewer-id abcdefg \  
  --viewer-session-versions-less-than-or-equal-to 1234567890
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan Saluran Privat](#) dalam Panduan Pengguna Amazon Interactive Video Service.

- Untuk detail API, lihat [StartViewerSessionRevocation](#) di Referensi AWS CLI Perintah.

## stop-stream

Contoh kode berikut menunjukkan cara menggunakan `stop-stream`.

### AWS CLI

Untuk menghentikan aliran tertentu

`stop-stream` Contoh berikut menghentikan aliran pada saluran yang ditentukan.

```
aws ivs stop-stream \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [StopStream](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambah atau memperbarui tag untuk AWS sumber daya (misalnya: channel, stream key)

`tag-resource` Contoh berikut menambahkan atau memperbarui tag untuk ARN sumber daya tertentu (Amazon Resource Name).

```
aws ivs tag-resource \  
  --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --tags "tagkey1=tagvalue1, tagkey2=tagvalue2"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Referensi API Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag untuk AWS sumber daya (misalnya: saluran, kunci aliran)

`untag-resource` Contoh berikut menghapus tag yang ditentukan untuk ARN sumber daya tertentu (Amazon Resource Name).

```
aws ivs untag-resource \  
  --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --tag-keys "tagkey1, tagkey2"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Referensi API Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-channel

Contoh kode berikut menunjukkan cara menggunakan `update-channel`.

### AWS CLI

Contoh 1: Untuk memperbarui informasi konfigurasi saluran

`update-channel` Contoh berikut memperbarui konfigurasi saluran untuk ARN saluran tertentu untuk mengubah nama saluran. Ini tidak memengaruhi aliran saluran ini yang sedang berlangsung; Anda harus menghentikan dan memulai ulang aliran agar perubahan diterapkan.

```
aws ivs update-channel \  
  --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --name "channel-1" \  
  --insecure-ingest
```

Output:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "name": "channel-1",  
    "latencyMode": "LOW",  
    "type": "STANDARD",  
    "playbackRestrictionPolicyArn": "",  
    "recordingConfigurationArn": "",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"  
    },  
      "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
      "insecureIngest": true,  
      "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/  
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
      "preset": "",  
      "authorized": false,  
      "tags": {}  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan Pengguna Latensi Rendah IVS.

## Contoh 2: Untuk memperbarui konfigurasi saluran untuk mengaktifkan perekaman

`update-channel` Contoh berikut memperbarui konfigurasi saluran untuk ARN saluran tertentu untuk mengaktifkan perekaman. Ini tidak memengaruhi aliran saluran ini yang sedang berlangsung; Anda harus menghentikan dan memulai ulang aliran agar perubahan diterapkan.

```
aws ivs update-channel \  
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \  
  --no-insecure-ingest \  
  --recording-configuration-arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh"
```

### Output:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "name": "test-channel-with-recording",  
    "latencyMode": "LOW",  
    "type": "STANDARD",  
    "playbackRestrictionPolicyArn": "",  
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"BA1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"  
    },  
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
    "insecureIngest": false,  
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
    "preset": "",  
    "authorized": false,  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3 di Panduan Pengguna Latensi Rendah IVS](#).

## Contoh 3: Untuk memperbarui konfigurasi saluran untuk menonaktifkan perekaman

`update-channel` Contoh berikut memperbarui konfigurasi saluran untuk ARN saluran tertentu untuk menonaktifkan perekaman. Ini tidak memengaruhi aliran saluran ini yang sedang berlangsung; Anda harus menghentikan dan memulai ulang aliran agar perubahan diterapkan.

```
aws ivs update-channel \
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \
  --recording-configuration-arn ""
```

Output:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-recording",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "",
    "recordingConfigurationArn": "",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2edfGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Merekam ke Amazon S3 di Panduan Pengguna Latensi Rendah IVS](#).

Contoh 4: Untuk memperbarui konfigurasi saluran untuk mengaktifkan pembatasan pemutaran

`update-channel` Contoh berikut memperbarui konfigurasi saluran untuk ARN saluran tertentu untuk menerapkan kebijakan pembatasan pemutaran. Ini tidak memengaruhi aliran saluran ini yang sedang berlangsung; Anda harus menghentikan dan memulai ulang aliran agar perubahan diterapkan.

```
aws ivs update-channel \
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \
  --no-insecure-ingest \
  --playback-restriction-policy-arn "arn:aws:ivs:us-west-2:123456789012:playback-
restriction-policy/ABcdef34ghIJ"
```

### Output:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-playback-restriction-policy",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
    "recordingConfigurationArn": "",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2defGHijkLMNo3PqQRstUvwxyzaCBDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

Contoh 5: Untuk memperbarui konfigurasi saluran untuk menonaktifkan pembatasan pemutaran

`update-channel` Contoh berikut memperbarui konfigurasi saluran untuk ARN saluran tertentu untuk menonaktifkan pembatasan pemutaran. Ini tidak memengaruhi aliran saluran ini yang sedang berlangsung; Anda harus menghentikan dan memulai ulang aliran agar perubahan diterapkan.



```
aws ivs update-channel \  
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \  
  --playback-restriction-policy-arn ""
```

### Output:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "name": "test-channel-with-playback-restriction-policy",  
    "latencyMode": "LOW",  
    "type": "STANDARD",  
    "playbackRestrictionPolicyArn": "",  
    "recordingConfigurationArn": "",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"AB1C2defGHijklMNop3PqQRstUvwxyzABCDeFghh4ijklMN5opqrStuVWXYZAbCDEfghIJ"  
    },  
      "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
      "insecureIngest": false,  
      "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/  
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
      "preset": "",  
      "authorized": false,  
      "tags": {}  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [UpdateChannel](#) di Referensi AWS CLI Perintah.

## update-playback-restriction-policy

Contoh kode berikut menunjukkan cara menggunakan `update-playback-restriction-policy`.

### AWS CLI

Untuk memperbarui kebijakan pembatasan pemutaran

`update-playback-restriction-policy` Contoh berikut memperbarui kebijakan pembatasan pemutaran dengan kebijakan ARN yang ditentukan untuk menonaktifkan penegakan asal yang ketat. Ini tidak memengaruhi aliran saluran terkait yang sedang berlangsung; Anda harus menghentikan dan memulai ulang aliran agar perubahan diterapkan.

```
aws ivs update-playback-restriction-policy \  
  --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
ABcdef34ghIJ" \  
  --no-enable-strict-origin-enforcement
```

Output:

```
{  
  "playbackRestrictionPolicy": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
ABcdef34ghIJ",  
    "allowedCountries": [  
      "US",  
      "MX"  
    ],  
    "allowedOrigins": [  
      "https://www.website1.com",  
      "https://www.website2.com"  
    ],  
    "enableStrictOriginEnforcement": false,  
    "name": "test-playback-restriction-policy",  
    "tags": {  
      "key1": "value1",  
      "key2": "value2"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Konten dan Pemirsa yang Tidak Diinginkan](#) di Panduan Pengguna Latensi Rendah IVS.

- Untuk detail API, lihat [UpdatePlaybackRestrictionPolicy](#) di Referensi AWS CLI Perintah.

## Contoh Obrolan Amazon IVS menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan menerapkan skenario umum AWS Command Line Interface dengan menggunakan Obrolan IVS Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-chat-token**

Contoh kode berikut menunjukkan cara menggunakan `create-chat-token`.

AWS CLI

Untuk membuat token obrolan

`create-chat-token` Contoh berikut membuat token obrolan terenkripsi yang digunakan untuk membuat WebSocket koneksi individual ke sebuah ruangan. Token berlaku selama satu menit, dan koneksi (sesi) yang dibuat dengan token berlaku untuk durasi yang ditentukan.

```
aws ivschat create-chat-token \
  --roomIdIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6", \
  --userId "11231234" \
  --capabilities "SEND_MESSAGE", \
  --sessionDurationInMinutes 30
```

Output:

```
{
  "token": "ACEGmnoq#1rstu2...BDFH3vxwy!4hlm!#5",
  "sessionExpirationTime": "2022-03-16T04:44:09+00:00"
  "state": "CREATING",
  "tokenExpirationTime": "2022-03-16T03:45:09+00:00"
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Mengautentikasi dan Mengotorisasi Klien Obrolan](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateChatToken](#) di Referensi AWS CLI Perintah.

## create-logging-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-logging-configuration`.

### AWS CLI

Untuk membuat `LoggingConfiguration` sumber daya obrolan

`create-logging-configuration` Contoh berikut menciptakan `LoggingConfiguration` sumber daya yang memungkinkan klien untuk menyimpan dan merekam pesan terkirim.

```
aws ivschat create-logging-configuration \
  --destination-configuration s3={bucketName=demo-logging-bucket} \
  --name "test-logging-config" \
  --tags "key1=value1, key2=value2"
```

### Output:

```
{
  "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ",
  "createTime": "2022-09-14T17:48:00.653000+00:00",
  "destinationConfiguration": {
    "s3": {
      "bucketName": "demo-logging-bucket"
    }
  },
  "id": "ABcdef34ghIJ",
  "name": "test-logging-config",
}
```

```
"state": "ACTIVE",
"tags": { "key1" : "value1", "key2" : "value2" },
"updateTime": "2022-09-14T17:48:01.104000+00:00"
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## create-room

Contoh kode berikut menunjukkan cara menggunakan `create-room`.

### AWS CLI

Untuk membuat ruangan

`create-room` Contoh berikut menciptakan ruangan baru.

```
aws ivschat create-room \
  --name "test-room-1" \
  --logging-configuration-identifiers "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ" \
  --maximum-message-length 256 \
  --maximum-message-rate-per-second 5
```

Output:

```
{
  "arn": "arn:aws:ivschat:us-west-2:123456789012:room/g1H2I3j4k5L6",
  "id": "g1H2I3j4k5L6",
  "createTime": "2022-03-16T04:44:09+00:00",
  "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ"],
  "maximumMessageLength": 256,
  "maximumMessageRatePerSecond": 5,
  "name": "test-room-1",
  "tags": {}
  "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Untuk informasi selengkapnya, lihat [Langkah 2: Membuat Ruang Obrolan](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateRoom](#) di Referensi AWS CLI Perintah.

## **delete-logging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-logging-configuration`.

### AWS CLI

Untuk menghapus `LoggingConfiguration` sumber daya obrolan

`delete-logging-configuration` Contoh berikut menghapus `LoggingConfiguration` sumber daya untuk ARN yang ditentukan.

```
aws ivschat delete-logging-configuration \  
  --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
ABcdef34ghIJ"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-message**

Contoh kode berikut menunjukkan cara menggunakan `delete-message`.

### AWS CLI

Untuk menghapus pesan dari ruangan tertentu

`delete-message` Contoh berikut mengirimkan genap ke ruang yang ditentukan, yang mengarahkan klien untuk menghapus pesan yang ditentukan: yaitu, membatalkan render dari tampilan dan menghapusnya dari riwayat obrolan klien.

```
aws ivschat delete-message \  
  --roomIdIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \  
  --id "ABC123def456" \  
  --
```

```
--reason "Message contains profanity"
```

Output:

```
{
  "id": "12345689012"
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteMessage](#) di Referensi AWS CLI Perintah.

## **delete-room**

Contoh kode berikut menunjukkan cara menggunakan `delete-room`.

### AWS CLI

Untuk menghapus ruangan

`delete-room` Contoh berikut menghapus ruangan yang ditentukan. Klien yang terhubung terputus. Setelah berhasil ia mengembalikan HTTP 204 dengan badan respons kosong.

```
aws ivschat delete-room \
  --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteRoom](#) di Referensi AWS CLI Perintah.

## **disconnect-user**

Contoh kode berikut menunjukkan cara menggunakan `disconnect-user`.

### AWS CLI

Untuk memutuskan koneksi pengguna dari sebuah ruangan

`disconnect-user` Contoh berikut memutus semua koneksi untuk pengguna yang ditentukan dari ruangan yang ditentukan. Setelah berhasil ia mengembalikan HTTP 200 dengan badan respons kosong.

```
aws ivschat disconnect-user \  
  --roomIdIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \  
  --userId "ABC123def456" \  
  --reason "Violated terms of service"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DisconnectUser](#) di Referensi AWS CLI Perintah.

## get-logging-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-logging-configuration`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber LoggingConfiguration daya

`get-logging-configuration` Contoh berikut mendapatkan informasi tentang LoggingConfiguration sumber daya untuk ARN yang ditentukan.

```
aws ivschat get-logging-configuration \  
  --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
  ABcdef34ghIJ"
```

Output:

```
{  
  "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
  ABcdef34ghIJ",  
  "createTime": "2022-09-14T17:48:00.653000+00:00",  
  "destinationConfiguration": {  
    "s3": {  
      "bucketName": "demo-logging-bucket"  
    }  
  },  
}
```



```
"id": "ABCdef34ghIJ",
"name": "test-logging-config",
"state": "ACTIVE",
"tags": { "key1" : "value1", "key2" : "value2" },
"updateTime": "2022-09-14T17:48:01.104000+00:00"
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## get-room

Contoh kode berikut menunjukkan cara menggunakan `get-room`.

### AWS CLI

Untuk mendapatkan ruangan yang ditentukan

`get-room` Contoh berikut mendapat informasi tentang ruangan yang ditentukan.

```
aws ivschat get-room \
  --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6"
```

Output:

```
{
  "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
  "createTime": "2022-03-16T04:44:09+00:00",
  "id": "g1H2I3j4k5L6",
  "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ"],
  "maximumMessageLength": 256,
  "maximumMessageRatePerSecond": 5,
  "name": "test-room-1",
  "tags": {},
  "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetRoom](#) di Referensi AWS CLI Perintah.

## list-logging-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-logging-configurations`.

### AWS CLI

Untuk mendapatkan informasi ringkasan tentang semua konfigurasi logging untuk pengguna di AWS wilayah tempat permintaan API diproses

`list-logging-configurations` Contoh berikut mencantumkan informasi tentang semua `LoggingConfiguration` sumber daya untuk pengguna di AWS wilayah tempat permintaan API diproses.

```
aws ivschat list-logging-configurations \
  --max-results 2 \
  --next-token ""
```

### Output:

```
{
  "nextToken": "set-2",
  "loggingConfigurations": [
    {
      "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ",
      "createTime": "2022-09-14T17:48:00.653000+00:00",
      "destinationConfiguration": {
        "s3": {
          "bucketName": "demo-logging-bucket"
        }
      },
      "id": "ABcdef34ghIJ",
      "name": "test-logging-config",
      "state": "ACTIVE",
      "tags": { "key1" : "value1", "key2" : "value2" },
      "updateTime": "2022-09-14T17:48:01.104000+00:00"
    }
    ...
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListLoggingConfigurations](#) di Referensi AWS CLI Perintah.

## list-rooms

Contoh kode berikut menunjukkan cara menggunakan `list-rooms`.

### AWS CLI

Untuk mendapatkan informasi ringkasan tentang semua kamar Anda di wilayah saat ini

`list-rooms` Contoh berikut mendapatkan informasi ringkasan tentang semua kamar di AWS wilayah tempat permintaan diproses. Hasil diurutkan dalam urutan menurun dari `UpdateTime`.

```
aws ivschat list-rooms \
  --logging-configuration-identifier "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ" \
  --max-results 10 \
  --next-token ""
```

### Output:

```
{
  "nextToken": "page3",
  "rooms": [
    {
      "arn:aws:ivschat:us-west-2:123456789012:room/g1H2I3j4k5L6",
      "createTime": "2022-03-16T04:44:09+00:00",
      "id": "g1H2I3j4k5L6",
      "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ"],
      "name": "test-room-1",
      "tags": {},
      "updateTime": "2022-03-16T07:22:09+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListRooms](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan semua tag untuk AWS sumber daya (misalnya: Room)

`list-tags-for-resource` Contoh berikut mencantumkan semua tag untuk ARN sumber daya tertentu (Amazon Resource Name).

```
aws ivschat list-tags-for-resource \
  --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6
```

Output:

```
{
  "tags":
  {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai](#) di Referensi API Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## send-event

Contoh kode berikut menunjukkan cara menggunakan `send-event`.

### AWS CLI

Untuk mengirim acara ke sebuah ruangan

`send-event` Contoh berikut mengirimkan acara yang diberikan ke ruangan yang ditentukan.

```
aws ivschat send-event \
  --roomIdIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
```

```
--eventName "SystemMessage" \  
--attributes \  
  "msgType"="user-notification", \  
  "msgText"="This chat room will close in 15 minutes."
```

Output:

```
{  
  "id": "12345689012"  
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [SendEvent](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambah atau memperbarui tag untuk AWS sumber daya (misalnya: Room)

`tag-resource` Contoh berikut menambahkan atau memperbarui tag untuk ARN sumber daya tertentu (Amazon Resource Name). Setelah berhasil ia mengembalikan HTTP 200 dengan badan respons kosong.

```
aws ivschat tag-resource \  
  --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6 \  
  --tags "tagkey1=tagkeyvalue1, tagkey2=tagkeyvalue2"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Referensi API Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus tag untuk AWS sumber daya (misalnya: Room)

`untag-resource` Contoh berikut menghapus tag yang ditentukan untuk ARN sumber daya tertentu (Amazon Resource Name). Setelah berhasil ia mengembalikan HTTP 200 dengan badan respons kosong.

```
aws ivschat untag-resource \  
  --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6 \  
  --tag-keys "tagkey1, tagkey2"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai](#) di Referensi API Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-logging-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-logging-configuration`.

## AWS CLI

Untuk memperbarui konfigurasi logging ruangan

`update-logging-configuration` Contoh berikut memperbarui LoggingConfiguration sumber daya dengan data yang diberikan.

```
aws ivschat update-logging-configuration \  
  --destination-configuration s3={bucketName=demo-logging-bucket} \  
  --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
ABcdef34ghIJ" \  
  --name "test-logging-config"
```

Output:

```
{  
  "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
ABcdef34ghIJ",  
  "createTime": "2022-09-14T17:48:00.653000+00:00",
```

```

    "destinationConfiguration": {
      "s3": {
        "bucketName": "demo-logging-bucket"
      }
    },
    "id": "ABcdef34ghIJ",
    "name": "test-logging-config",
    "state": "ACTIVE",
    "tags": { "key1" : "value1", "key2" : "value2" },
    "updateTime": "2022-09-14T17:48:01.104000+00:00"
  }
}

```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [UpdateLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## update-room

Contoh kode berikut menunjukkan cara menggunakan `update-room`.

### AWS CLI

Untuk memperbarui konfigurasi ruangan

`update-room` Contoh berikut memperbarui konfigurasi ruangan yang ditentukan dengan data yang diberikan.

```

aws ivschat update-room \
  --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
  --logging-configuration-identifiers "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ" \
  --name "chat-room-a" \
  --maximum-message-length 256 \
  --maximum-message-rate-per-second 5

```

Output:

```

{
  "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
  "createTime": "2022-03-16T04:44:09+00:00",
  "id": "g1H2I3j4k5L6",

```

```
"loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-west-2:123456789012:logging-configuration/ABCdef34ghIJ"],
  "maximumMessageLength": 256,
  "maximumMessageRatePerSecond": 5,
  "name": "chat-room-a",
  "tags": {},
  "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Untuk informasi selengkapnya, lihat [Memulai Obrolan Amazon IVS](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [UpdateRoom](#) di Referensi AWS CLI Perintah.

## Contoh Streaming Waktu Nyata Amazon IVS menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan menerapkan skenario umum AWS Command Line Interface dengan menggunakan Streaming Real-Time Amazon IVS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-encoder-configuration**

Contoh kode berikut menunjukkan cara menggunakan `create-encoder-configuration`.

AWS CLI

Untuk membuat konfigurasi encoder komposisi



`create-encoder-configuration` Contoh berikut membuat konfigurasi encoder komposisi dengan properti yang ditentukan.

```
aws ivs-realtime create-encoder-configuration \  
  --name test-ec --video bitrate=3500000,framerate=30.0,height=1080,width=1920
```

Output:

```
{  
  "encoderConfiguration": {  
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/  
ABabCDcdEFef",  
    "name": "test-ec",  
    "tags": {},  
    "video": {  
      "bitrate": 3500000,  
      "framerate": 30,  
      "height": 1080,  
      "width": 1920  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateEncoderConfiguration](#) di Referensi AWS CLI Perintah.

## `create-participant-token`

Contoh kode berikut menunjukkan cara menggunakan `create-participant-token`.

### AWS CLI

Untuk membuat token peserta panggung

`create-participant-token` Contoh berikut membuat toke peserta untuk tahap yang ditentukan.

```
aws ivs-realtime create-participant-token \  
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \  
  --stage-name test-stage
```

```
--user-id bob
```

Output:

```
{
  "participantToken": {
    "expirationTime": "2023-03-07T09:47:43+00:00",
    "participantId": "ABCDEFghij01234KLMN6789",
    "token": "abcd1234defg5678"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateParticipantToken](#) di Referensi AWS CLI Perintah.

## create-stage

Contoh kode berikut menunjukkan cara menggunakan `create-stage`.

### AWS CLI

Untuk membuat panggung

`create-stage` Contoh berikut membuat token peserta tahap dan tahap untuk pengguna tertentu.

```
aws ivs-realtime create-stage \
  --name stage1 \
  --participant-token-configurations userId=alice
```

Output:

```
{
  "participantTokens": [
    {
      "participantId": "ABCDEFghij01234KLMN5678",
      "token": "a1b2c3d4567890ab",
      "userId": "alice"
    }
  ],
  "stage": {
```

```
"activeSessionId": "st-a1b2c3d4e5f6g",
"arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
"name": "stage1",
"tags": {}
}
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateStage](#) di Referensi AWS CLI Perintah.

## create-storage-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-storage-configuration`.

### AWS CLI

Untuk membuat konfigurasi penyimpanan komposisi

`create-storage-configuration` Contoh berikut membuat konfigurasi penyimpanan komposisi dengan properti yang ditentukan.

```
aws ivs-realtime create-storage-configuration \
  --name "test-sc" --s3 "bucketName=test-bucket-name"
```

Output:

```
{
  "storageConfiguration": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
ABabCDcdEFef",
    "name": "test-sc",
    "s3": {
      "bucketName": "test-bucket-name"
    },
    "tags": {}
  }
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [CreateStorageConfiguration](#) di Referensi AWS CLI Perintah.

## delete-encoder-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-encoder-configuration`.

### AWS CLI

Untuk menghapus konfigurasi encoder komposisi

Berikut ini `delete-encoder-configuration` menghapus konfigurasi encoder komposisi yang ditentukan oleh ARN yang diberikan (Amazon Resource Name).

```
aws ivs-realtime delete-encoder-configuration \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/  
ABabCDcdEFef"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteEncoderConfiguration](#) di Referensi AWS CLI Perintah.

## delete-stage

Contoh kode berikut menunjukkan cara menggunakan `delete-stage`.

### AWS CLI

Untuk menghapus panggung

`delete-stage` Contoh berikut menghapus tahap yang ditentukan.

```
aws ivs-realtime delete-stage \  
  --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteStage](#) di Referensi AWS CLI Perintah.

## delete-storage-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-storage-configuration`.

### AWS CLI

Untuk menghapus konfigurasi penyimpanan komposisi

Berikut ini `delete-storage-configuration` menghapus konfigurasi penyimpanan komposisi yang ditentukan oleh ARN yang diberikan (Nama Sumber Daya Amazon).

```
aws ivs-realtime delete-storage-configuration \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/  
  ABabCDcdEFef"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DeleteStorageConfiguration](#) di Referensi AWS CLI Perintah.

## disconnect-participant

Contoh kode berikut menunjukkan cara menggunakan `disconnect-participant`.

### AWS CLI

Untuk memutuskan sambungan peserta panggung

`disconnect-participant` Contoh berikut memutus peserta yang ditentukan dari tahap yang ditentukan.

```
aws ivs-realtime disconnect-participant \  
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \  
  --participant-id ABCDEfghij01234KLMN5678
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [DisconnectParticipant](#) di Referensi AWS CLI Perintah.

## get-composition

Contoh kode berikut menunjukkan cara menggunakan `get-composition`.

### AWS CLI

Contoh 1: Untuk mendapatkan komposisi dengan pengaturan tata letak default

`get-composition` Contoh berikut mendapatkan komposisi untuk ARN (Amazon Resource Name) yang ditentukan.

```
aws ivs-realtime get-composition \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Output:

```
{  
  "composition": {  
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",  
    "destinations": [  
      {  
        "configuration": {  
          "channel": {  
            "channelArn": "arn:aws:ivs:ap-northeast-1:123456789012:channel/abcABCdefDEg",  
            "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"  
          },  
          "name": ""  
        },  
        "id": "AabBCcdDEefF",  
        "startTime": "2023-10-16T23:26:00+00:00",  
        "state": "ACTIVE"  
      },  
      {  
        "configuration": {  
          "name": "",  
          "s3": {  
            "encoderConfigurationArns": [  
              "arn:aws:ivs:arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"  
            ]  
          }  
        }  
      }  
    ]  
  }  
}
```

```

        ],
        "recordingConfiguration": {
            "format": "HLS"
        },
        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
    }
},
"detail": {
    "s3": {
        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
    }
},
"id": "GHFabcgefABC",
"startTime": "2023-10-16T23:26:00+00:00",
"state": "STARTING"
}
],
"layout": {
    "grid": {
        "featuredParticipantAttribute": ""
        "gridGap": 2,
        "omitStoppedVideo": false,
        "videoAspectRatio": "VIDEO",
        "videoFillMode": ""
    }
},
"stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
"startTime": "2023-10-16T23:24:00+00:00",
"state": "ACTIVE",
"tags": {}
}
}

```

Untuk informasi selengkapnya, lihat [Perekaman Komposit \(Streaming Waktu Nyata\)](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

Contoh 2: Untuk mendapatkan komposisi dengan tata letak PiP

get-composition Contoh berikut mendapatkan komposisi untuk ARN (Amazon Resource Name) yang ditentukan, yang menggunakan tata letak PiP.

```
aws ivs-realtime get-composition \
```

```
--arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs"
```

### Output:

```
{
  "composition": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs",
    "destinations": [
      {
        "configuration": {
          "channel": {
            "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
            "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
          },
          "name": ""
        },
        "id": "AabBCcdDEefF",
        "startTime": "2023-10-16T23:26:00+00:00",
        "state": "ACTIVE"
      },
      {
        "configuration": {
          "name": "",
          "s3": {
            "encoderConfigurationArns": [
              "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
            ],
            "recordingConfiguration": {
              "format": "HLS"
            },
            "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
          }
        },
        "detail": {
          "s3": {
            "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgfABC/
composite"
          }
        }
      }
    ]
  }
}
```



```

        "id": "GHFabcgefABC",
        "startTime": "2023-10-16T23:26:00+00:00",
        "state": "STARTING"
    }
],
"layout": {
    "pip": {
        "featuredParticipantAttribute": "abcdefg",
        "gridGap": 0,
        "omitStoppedVideo": false,
        "pipBehavior": "STATIC",
        "pipOffset": 0,
        "pipParticipantAttribute": "",
        "pipPosition": "BOTTOM_RIGHT",
        "videoFillMode": "COVER"
    }
},
"stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
"startTime": "2023-10-16T23:24:00+00:00",
"state": "ACTIVE",
"tags": {}
}
}

```

Untuk informasi selengkapnya, lihat [Perekaman Komposit \(Streaming Waktu Nyata\)](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetComposition](#) di Referensi AWS CLI Perintah.

## get-encoder-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-encoder-configuration`.

### AWS CLI

Untuk mendapatkan konfigurasi encoder komposisi

`get-encoder-configuration` Contoh berikut mendapatkan konfigurasi encoder komposisi yang ditentukan oleh ARN yang diberikan (Amazon Resource Name).

```

aws ivs-realtime get-encoder-configuration \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh"

```

**Output:**

```
{
  "encoderConfiguration": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh",
    "name": "test-ec",
    "tags": {},
    "video": {
      "bitrate": 3500000,
      "framerate": 30,
      "height": 1080,
      "width": 1920
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetEncoderConfiguration](#) di Referensi AWS CLI Perintah.

**get-participant**

Contoh kode berikut menunjukkan cara menggunakan `get-participant`.

**AWS CLI**

Untuk mendapatkan peserta panggung

`get-participant` Contoh berikut mendapatkan peserta tahap untuk ID peserta dan ID sesi tertentu di ARN tahap tertentu (Nama Sumber Daya Amazon).

```
aws ivs-realtime get-participant \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g \
  --participant-id abCDEf12GHIj
```

**Output:**

```
{
```

```
"participant": {
  "browserName", "Google Chrome",
  "browserVersion", "116",
  "firstJoinTime": "2023-04-26T20:30:34+00:00",
  "ispName", "Comcast",
  "osName", "Microsoft Windows 10 Pro",
  "osVersion", "10.0.19044",
  "participantId": "abCDEf12GHIj",
  "published": true,
  "sdkVersion", "",
  "state": "DISCONNECTED",
  "userId": ""
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetParticipant](#) di Referensi AWS CLI Perintah.

## get-stage-session

Contoh kode berikut menunjukkan cara menggunakan `get-stage-session`.

### AWS CLI

Untuk mendapatkan sesi panggung

`get-stage-session` Contoh berikut mendapatkan sesi tahap untuk ID sesi tertentu dari ARN tahap tertentu (Amazon Resource Name).

```
aws ivs-realtime get-stage-session \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g
```

Output:

```
{
  "stageSession": {
    "endTime": "2023-04-26T20:36:29+00:00",
    "sessionId": "st-a1b2c3d4e5f6g",
    "startTime": "2023-04-26T20:30:29.602000+00:00"
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetStageSession](#) di Referensi AWS CLI Perintah.

## get-stage

Contoh kode berikut menunjukkan cara menggunakan `get-stage`.

### AWS CLI

Untuk mendapatkan informasi konfigurasi panggung

`get-stage` Contoh berikut mendapatkan konfigurasi tahap untuk ARN tahap tertentu (Amazon Resource Name).

```
aws ivs-realtime get-stage \  
  --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Output:

```
{  
  "stage": {  
    "activeSessionId": "st-a1b2c3d4e5f6g",  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",  
    "name": "test",  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetStage](#) di Referensi AWS CLI Perintah.

## get-storage-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-storage-configuration`.

## AWS CLI

Untuk mendapatkan konfigurasi penyimpanan komposisi

`get-storage-configuration` Contoh berikut mendapatkan konfigurasi penyimpanan komposisi yang ditentukan oleh ARN yang diberikan (Amazon Resource Name).

```
aws ivs-realtime get-storage-configuration \  
  --name arn "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/  
abcdABCDefgh"
```

Output:

```
{  
  "storageConfiguration": {  
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/  
abcdABCDefgh",  
    "name": "test-sc",  
    "s3": {  
      "bucketName": "test-bucket-name"  
    },  
    "tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [GetStorageConfiguration](#) di Referensi AWS CLI Perintah.

## **list-compositions**

Contoh kode berikut menunjukkan cara menggunakan `list-compositions`.

## AWS CLI

Untuk mendapatkan daftar komposisi

Berikut ini `list-compositions` mencantumkan semua komposisi untuk AWS akun Anda, di AWS wilayah tempat permintaan API diproses.

```
aws ivs-realtime list-compositions
```

## Output:

```
{
  "compositions": [
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/
abcdABCDefgh",
      "destinations": [
        {
          "id": "AabBCcdDEefF",
          "startTime": "2023-10-16T23:25:23+00:00",
          "state": "ACTIVE"
        }
      ],
      "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/
defgABCDabcd",
      "startTime": "2023-10-16T23:25:21+00:00",
      "state": "ACTIVE",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/
ABcdabCDefgh",
      "destinations": [
        {
          "endTime": "2023-10-16T23:25:00.786512+00:00",
          "id": "aABbcCDdeEFf",
          "startTime": "2023-10-16T23:24:01+00:00",
          "state": "STOPPED"
        },
        {
          "endTime": "2023-10-16T23:25:00.786512+00:00",
          "id": "deEFfaABbcCD",
          "startTime": "2023-10-16T23:24:01+00:00",
          "state": "STOPPED"
        }
      ],
      "endTime": "2023-10-16T23:25:00+00:00",
      "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/
efghabcdABCD",
      "startTime": "2023-10-16T23:24:00+00:00",
      "state": "STOPPED",
      "tags": {}
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListCompositions](#) di Referensi AWS CLI Perintah.

## list-encoder-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-encoder-configurations`.

### AWS CLI

Untuk membuat daftar konfigurasi encoder komposisi

Berikut ini `list-encoder-configurations` mencantumkan semua konfigurasi encoder komposisi untuk AWS akun Anda, di AWS wilayah tempat permintaan API diproses.

```
aws ivs-realtime list-encoder-configurations
```

Output:

```
{
  "encoderConfigurations": [
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/abcdABCDefgh",
      "name": "test-ec-1",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABCefgEFGabc",
      "name": "test-ec-2",
      "tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListEncoderConfigurations](#) di Referensi AWS CLI Perintah.

## list-participant-events

Contoh kode berikut menunjukkan cara menggunakan `list-participant-events`.

### AWS CLI

Untuk mendapatkan daftar acara peserta panggung

`list-participant-events` Contoh berikut mencantumkan semua peristiwa peserta untuk ID peserta tertentu dan ID sesi dari ARN tahap tertentu (Nama Sumber Daya Amazon).

```
aws ivs-realtime list-participant-events \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g \
  --participant-id abCDEf12GHIj
```

### Output:

```
{
  "events": [
    {
      "eventTime": "2023-04-26T20:36:28+00:00",
      "name": "LEFT",
      "participantId": "abCDEf12GHIj"
    },
    {
      "eventTime": "2023-04-26T20:36:28+00:00",
      "name": "PUBLISH_STOPPED",
      "participantId": "abCDEf12GHIj"
    },
    {
      "eventTime": "2023-04-26T20:30:34+00:00",
      "name": "JOINED",
      "participantId": "abCDEf12GHIj"
    },
    {
      "eventTime": "2023-04-26T20:30:34+00:00",
      "name": "PUBLISH_STARTED",
      "participantId": "abCDEf12GHIj"
    }
  ]
}
```



```
]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListParticipantEvents](#) di Referensi AWS CLI Perintah.

## list-participants

Contoh kode berikut menunjukkan cara menggunakan `list-participants`.

### AWS CLI

Untuk mendapatkan daftar peserta panggung

`list-participants` Contoh berikut mencantumkan semua peserta untuk ID sesi tertentu dari ARN tahap tertentu (Nama Sumber Daya Amazon).

```
aws ivs-realtime list-participants \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g
```

Output:

```
{
  "participants": [
    {
      "firstJoinTime": "2023-04-26T20:30:34+00:00",
      "participantId": "abCDEf12GHIj"
      "published": true,
      "state": "DISCONNECTED",
      "userId": ""
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListParticipants](#) di Referensi AWS CLI Perintah.

## list-stage-sessions

Contoh kode berikut menunjukkan cara menggunakan `list-stage-sessions`.

### AWS CLI

Untuk mendapatkan daftar sesi panggung

`list-stage-sessions` Contoh berikut mencantumkan semua sesi untuk ARN tahap tertentu (Nama Sumber Daya Amazon).

```
aws ivs-realtime list-stage-sessions \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Output:

```
{
  "stageSessions": [
    {
      "endTime": "2023-04-26T20:36:29+00:00",
      "sessionId": "st-a1b2c3d4e5f6g",
      "startTime": "2023-04-26T20:30:29.602000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListStageSessions](#) di Referensi AWS CLI Perintah.

## list-stages

Contoh kode berikut menunjukkan cara menggunakan `list-stages`.

### AWS CLI

Untuk mendapatkan informasi ringkasan tentang semua tahapan

`list-stages` Contoh berikut mencantumkan semua tahapan untuk AWS akun Anda, di AWS wilayah tempat permintaan API diproses.

```
aws ivs-realtime list-stages
```

Output:

```
{
  "stages": [
    {
      "activeSessionId": "st-a1b2c3d4e5f6g",
      "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
      "name": "stage1",
      "tags": {}
    },
    {
      "activeSessionId": "st-a123bcd456efg",
      "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcd1234ABCD",
      "name": "stage2",
      "tags": {}
    },
    {
      "activeSessionId": "st-abcDEF1234ghi",
      "arn": "arn:aws:ivs:us-west-2:123456789012:stage/ABCD1234efgh",
      "name": "stage3",
      "tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListStages](#) di Referensi AWS CLI Perintah.

## list-storage-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-storage-configurations`.

### AWS CLI

Untuk membuat daftar konfigurasi penyimpanan komposisi

Berikut ini `list-storage-configurations` mencantumkan semua konfigurasi penyimpanan komposisi untuk AWS akun Anda, di AWS wilayah tempat permintaan API diproses.

```
aws ivs-realtime list-storage-configurations
```

Output:

```
{
  "storageConfigurations": [
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/abcdABCDefgh",
      "name": "test-sc-1",
      "s3": {
        "bucketName": "test-bucket-1-name"
      },
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/ABCefgEFGabc",
      "name": "test-sc-2",
      "s3": {
        "bucketName": "test-bucket-2-name"
      },
      "tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [ListStorageConfigurations](#) di Referensi AWS CLI Perintah.

## start-composition

Contoh kode berikut menunjukkan cara menggunakan `start-composition`.

### AWS CLI

Contoh 1: Untuk memulai komposisi dengan pengaturan tata letak default

`start-composition` Contoh berikut memulai komposisi untuk tahap tertentu yang akan dialirkan ke lokasi yang ditentukan.

```
aws ivs-realtime start-composition \
  --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCdabcd \
  --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
  "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
  {"s3":{"encoderConfigurationArns":["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
  "storageConfigurationArn":"arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE"}}]'
```

### Output:

```
{
  "composition": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
    "destinations": [
      {
        "configuration": {
          "channel": {
            "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
            "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
          },
          "name": ""
        },
        "id": "AabBCcdDEefF",
        "state": "STARTING"
      },
      {
        "configuration": {
          "name": "",
          "s3": {
            "encoderConfigurationArns": [
              "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
            ],
            "recordingConfiguration": {
              "format": "HLS"
            },
            "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
          }
        }
      }
    ]
  }
}
```

```

        }
      },
      "detail": {
        "s3": {
          "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
        }
      },
      "id": "GHFabcgefABC",
      "state": "STARTING"
    }
  ],
  "layout": {
    "grid": {
      "featuredParticipantAttribute": ""
      "gridGap": 2,
      "omitStoppedVideo": false,
      "videoAspectRatio": "VIDEO",
      "videoFillMode": ""
    }
  },
  "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
  "startTime": "2023-10-16T23:24:00+00:00",
  "state": "STARTING",
  "tags": {}
}
}

```

Untuk informasi selengkapnya, lihat [Perekaman Komposit \(Streaming Waktu Nyata\)](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

Contoh 2: Untuk memulai komposisi dengan tata letak PiP

`start-composition` Contoh berikut memulai komposisi untuk tahap tertentu yang akan dialirkan ke lokasi yang ditentukan menggunakan tata letak PiP.

```

aws ivs-realtime start-composition \
  --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd \
  --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
    "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \

```

```
{
  "s3": {
    "encoderConfigurationArns": [
      "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
    ],
    "storageConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/FefABabCDcdE"
  }
}' \
--layout pip='{featuredParticipantAttribute="abcdefg}"'
```

### Output:

```
{
  "composition": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs",
    "destinations": [
      {
        "configuration": {
          "channel": {
            "channelArn": "arn:aws:ivs:ap-northeast-1:123456789012:channel/abcABCdefDEg",
            "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
          },
          "name": ""
        },
        "id": "AabBCcdDEefF",
        "state": "STARTING"
      },
      {
        "configuration": {
          "name": "",
          "s3": {
            "encoderConfigurationArns": [
              "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
            ],
            "recordingConfiguration": {
              "format": "HLS"
            },
            "storageConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/FefABabCDcdE"
          }
        },
        "detail": {
          "s3": {
```

```

        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
        }
    },
    "id": "GHFabcgefABC",
    "state": "STARTING"
}
],
"layout": {
    "pip": {
        "featuredParticipantAttribute": "abcdefg",
        "gridGap": 0,
        "omitStoppedVideo": false,
        "pipBehavior": "STATIC",
        "pipOffset": 0,
        "pipParticipantAttribute": "",
        "pipPosition": "BOTTOM_RIGHT",
        "videoFillMode": "COVER"
    }
},
"stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
"startTime": "2023-10-16T23:24:00+00:00",
"state": "STARTING",
"tags": {}
}
}

```

Untuk informasi selengkapnya, lihat [Perekaman Komposit \(Streaming Waktu Nyata\)](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [StartComposition](#) di Referensi AWS CLI Perintah.

## stop-composition

Contoh kode berikut menunjukkan cara menggunakan `stop-composition`.

### AWS CLI

Untuk menghentikan komposisi

Berikut ini `stop-composition` menghentikan komposisi yang ditentukan oleh ARN yang diberikan (Nama Sumber Daya Amazon).



```
aws ivs-realtime stop-composition \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [StopComposition](#) di Referensi AWS CLI Perintah.

## update-stage

Contoh kode berikut menunjukkan cara menggunakan `update-stage`.

### AWS CLI

Untuk memperbarui konfigurasi tahap

`update-stage` Contoh berikut memperbarui tahap untuk ARN tahap tertentu untuk memperbarui nama panggung.

```
aws ivs-realtime update-stage \  
  --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \  
  --name stage1a
```

Output:

```
{  
  "stage": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",  
    "name": "stage1a"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan Beberapa Host di Amazon IVS Stream](#) di Panduan Pengguna Layanan Video Interaktif Amazon.

- Untuk detail API, lihat [UpdateStage](#) di Referensi AWS CLI Perintah.

## Amazon Kendra contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon Kendra. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-data-source**

Contoh kode berikut menunjukkan cara menggunakan `create-data-source`.

AWS CLI

Untuk membuat konektor sumber data Amazon Kendra

Berikut ini `create-data-source` membuat dan mengkonfigurasi konektor sumber data Amazon Kendra. Anda dapat menggunakan `describe-data-source` untuk melihat status konektor sumber data, dan membaca pesan kesalahan apa pun jika status menunjukkan konektor sumber data “GAGAL” untuk sepenuhnya dibuat.

```
aws kendra create-data-source \  
  --name "example data source 1" \  
  --description "Example data source 1 for example index 1 contains the first set  
of example documents" \  
  --tags '{"Key": "test resources", "Value": "kendra"}, {"Key": "test resources",  
"Value": "aws"}' \  
  --role-arn "arn:aws:iam::my-account-id:role/  
KendraRoleForS3TemplateConfigDataSource" \  
  --index-id exampleindex1 \  

```

```

--language-code "es" \
--schedule "0 0 18 ? * TUE,MON,WED,THU,FRI,SAT *" \
--configuration '{"TemplateConfiguration": {"Template": file://
s3schemaconfig.json}}' \
--type "TEMPLATE" \
--custom-document-enrichment-configuration '{"PostExtractionHookConfiguration":
{"LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
"S3Bucket": "s3://my-s3-bucket/scanned-image-text-example-docs"}, "RoleArn":
"arn:aws:iam:my-account-id:role/KendraRoleForCDE"}' \
--vpc-configuration '{"SecurityGroupIds": ["sg-1234567890abcdef0"], "SubnetIds":
["subnet-1c234", "subnet-2b134"]}'

```

Output:

```

{
  "Id": "exampledatasource1"
}

```

Untuk informasi selengkapnya, lihat [Memulai indeks Amazon Kendra dan konektor sumber data di Panduan](#) Pengembang Amazon Kendra.

- Untuk detail API, lihat [CreateDataSource](#) di Referensi AWS CLI Perintah.

## create-index

Contoh kode berikut menunjukkan cara menggunakan `create-index`.

### AWS CLI

Untuk membuat indeks Amazon Kendra

Berikut ini `create-index` membuat dan mengkonfigurasi indeks Amazon Kendra. Anda dapat menggunakan `describe-index` untuk melihat status indeks, dan membaca pesan kesalahan apa pun jika status menunjukkan indeks “GAGAL” untuk sepenuhnya dibuat.

```

aws kendra create-index \
--name "example index 1" \
--description "Example index 1 contains the first set of example documents" \
--tags '{"Key": "test resources", "Value": "kendra"}, {"Key": "test resources",
"Value": "aws"}' \
--role-arn "arn:aws:iam::my-account-id:role/KendraRoleForExampleIndex" \
--edition "DEVELOPER_EDITION" \
--server-side-encryption-configuration '{"KmsKeyId": "my-kms-key-id"}' \

```

```
--user-context-policy "USER_TOKEN" \  
--user-token-configurations '{"JsonTokenTypeConfiguration":  
{"GroupAttributeField": "groupNameField", "UserNameAttributeField":  
"userNameField"}}'
```

Output:

```
{  
  "Id": index1  
}
```

Untuk informasi selengkapnya, lihat [Memulai indeks Amazon Kendra dan konektor sumber data di Panduan](#) Pengembang Amazon Kendra.

- Untuk detail API, lihat [CreateIndex](#) di Referensi AWS CLI Perintah.

## describe-data-source

Contoh kode berikut menunjukkan cara menggunakan `describe-data-source`.

### AWS CLI

Untuk mendapatkan informasi tentang konektor sumber data Amazon Kendra

Berikut ini `describe-data-source` mendapatkan informasi tentang konektor source data Amazon Kendra. Anda dapat melihat konfigurasi konektor sumber data, dan membaca pesan kesalahan apa pun jika status menunjukkan konektor sumber data “GAGAL” untuk sepenuhnya dibuat.

```
aws kendra describe-data-source \  
--id exampledatasource1 \  
--index-id exampleindex1
```

Output:

```
{  
  "Configuration": {  
    "TemplateConfiguration": {  
      "Template": {  
        "connectionConfiguration": {  
          "repositoryEndpointMetadata": {  
            "BucketName": "my-bucket"          }  
        }  
      }  
    }  
  }  
}
```

```
    }
  },
  "repositoryConfigurations": {
    "document":{
      "fieldMappings": [
        {
          "indexFieldName": "_document_title",
          "indexFieldType": "STRING",
          "dataSourceFieldName": "title"
        },
        {
          "indexFieldName": "_last_updated_at",
          "indexFieldType": "DATE",
          "dataSourceFieldName": "modified_date"
        }
      ]
    }
  },
  "additionalProperties": {
    "inclusionPatterns": [
      "*.txt",
      "*.doc",
      "*.docx"
    ],
    "exclusionPatterns": [
      "*.json"
    ],
    "inclusionPrefixes": [
      "PublicExampleDocsFolder"
    ],
    "exclusionPrefixes": [
      "PrivateDocsFolder/private"
    ],
    "aclConfigurationFilePath": "ExampleDocsFolder/AclConfig.json",
    "metadataFilesPrefix": "metadata"
  },
  "syncMode": "FULL_CRAWL",
  "type": "S3",
  "version": "1.0.0"
}
},
"CreatedAt": 2024-02-25T13:30:10+00:00,
"CustomDocumentEnrichmentConfiguration": {
```

```

    "PostExtractionHookConfiguration": {
      "LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
      "S3Bucket": "s3://my-s3-bucket/scanned-image-text-example-docs/function"
    },
    "RoleArn": "arn:aws:iam:my-account-id:role/KendraRoleForCDE"
  }
  "Description": "Example data source 1 for example index 1 contains the first set
of example documents",
  "Id": "exampledatasource1",
  "IndexId": "exampleindex1",
  "LanguageCode": "en",
  "Name": "example data source 1",
  "RoleArn": "arn:aws:iam::my-account-id:role/
KendraRoleForS3TemplateConfigDataSource",
  "Schedule": "0 0 18 ? * TUE,MON,WED,THU,FRI,SAT *",
  "Status": "ACTIVE",
  "Type": "TEMPLATE",
  "UpdatedAt": 1709163615,
  "VpcConfiguration": {
    "SecurityGroupIds": ["sg-1234567890abcdef0"],
    "SubnetIds": ["subnet-1c234", "subnet-2b134"]
  }
}

```

Untuk informasi selengkapnya, lihat [Memulai indeks Amazon Kendra dan konektor sumber data di Panduan](#) Pengembang Amazon Kendra.

- Untuk detail API, lihat [DescribeDataSource](#) di Referensi AWS CLI Perintah.

## describe-index

Contoh kode berikut menunjukkan cara menggunakan `describe-index`.

### AWS CLI

Untuk mendapatkan informasi tentang indeks Amazon Kendra

Berikut ini `describe-index` mendapatkan informasi tentang indeks Amazon Kendra. Anda dapat melihat konfigurasi indeks, dan membaca pesan kesalahan apa pun jika status menunjukkan indeks “GAGAL” untuk sepenuhnya dibuat.

```
aws kendra describe-index \
```

```
--id exampleindex1
```

**Output:**

```
{
  "CapacityUnits": {
    "QueryCapacityUnits": 0,
    "StorageCapacityUnits": 0
  },
  "CreatedAt": 2024-02-25T12:30:10+00:00,
  "Description": "Example index 1 contains the first set of example documents",
  "DocumentMetadataConfigurations": [
    {
      "Name": "_document_title",
      "Relevance": {
        "Importance": 8
      },
      "Search": {
        "Displayable": true,
        "Facetable": false,
        "Searchable": true,
        "Sortable": false
      },
      "Type": "STRING_VALUE"
    },
    {
      "Name": "_document_body",
      "Relevance": {
        "Importance": 5
      },
      "Search": {
        "Displayable": true,
        "Facetable": false,
        "Searchable": true,
        "Sortable": false
      },
      "Type": "STRING_VALUE"
    },
    {
      "Name": "_last_updated_at",
      "Relevance": {
        "Importance": 6,
        "Duration": "2628000s",

```

```
        "Freshness": true
    },
    "Search": {
        "Displayable": true,
        "Facetable": false,
        "Searchable": true,
        "Sortable": true
    },
    "Type": "DATE_VALUE"
},
{
    "Name": "department_custom_field",
    "Relevance": {
        "Importance": 7,
        "ValueImportanceMap": {
            "Human Resources" : 4,
            "Marketing and Sales" : 2,
            "Research and innvoation" : 3,
            "Admin" : 1
        }
    },
    "Search": {
        "Displayable": true,
        "Facetable": true,
        "Searchable": true,
        "Sortable": true
    },
    "Type": "STRING_VALUE"
}
],
"Edition": "DEVELOPER_EDITION",
"Id": "index1",
"IndexStatistics": {
    "FaqStatistics": {
        "IndexedQuestionAnswersCount": 10
    },
    "TextDocumentStatistics": {
        "IndexedTextBytes": 1073741824,
        "IndexedTextDocumentsCount": 1200
    }
},
"Name": "example index 1",
"RoleArn": "arn:aws:iam:my-account-id:role/KendraRoleForExampleIndex",
"ServerSideEncryptionConfiguration": {
```



```

    "KmsKeyId": "my-kms-key-id"
  },
  "Status": "ACTIVE",
  "UpdatedAt": 1709163615,
  "UserContextPolicy": "USER_TOKEN",
  "UserTokenConfigurations": [
    {
      "JsonTokenTypeConfiguration": {
        "GroupAttributeField": "groupNameField",
        "UserNameAttributeField": "userNameField"
      }
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Memulai indeks Amazon Kendra dan konektor sumber data di Panduan](#) Pengembang Amazon Kendra.

- Untuk detail API, lihat [DescribeIndex](#) di Referensi AWS CLI Perintah.

## update-data-source

Contoh kode berikut menunjukkan cara menggunakan `update-data-source`.

### AWS CLI

Untuk memperbarui konektor sumber data Amazon Kendra

Berikut ini `update-data-source` memperbarui konfigurasi konektor sumber data Amazon Kendra. Jika tindakan berhasil, layanan mengirim kembali tidak ada output, kode status HTTP 200, atau kode pengembalian AWS CLI 0. Anda dapat menggunakan `describe-data-source` untuk melihat konfigurasi dan status konektor sumber data.

```

aws kendra update-data-source \
  --id exampledatasource1 \
  --index-id exampleindex1 \
  --name "new name for example data source 1" \
  --description "new description for example data source 1" \
  --role-arn arn:aws:iam::my-account-id:role/KendraNewRoleForExampleDataSource \
  --configuration '{"TemplateConfiguration": {"Template": file://
s3schemanewconfig.json}}' \
  --custom-document-enrichment-configuration '{"PostExtractionHookConfiguration":
{"LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",

```

```
"S3Bucket": "s3://my-s3-bucket/scanned-image-text-example-docs"}, "RoleArn":
"arn:aws:iam:my-account-id:role/KendraNewRoleForCDE"}' \
--language-code "es" \
--schedule "0 0 18 ? * MON,WED,FRI *" \
--vpc-configuration '{"SecurityGroupIds": ["sg-1234567890abcdef0"], "SubnetIds":
["subnet-1c234","subnet-2b134"]}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai indeks Amazon Kendra dan konektor sumber data di Panduan](#) Pengembang Amazon Kendra.

- Untuk detail API, lihat [UpdateDataSource](#) di Referensi AWS CLI Perintah.

## update-index

Contoh kode berikut menunjukkan cara menggunakan `update-index`.

### AWS CLI

Untuk memperbarui indeks Amazon Kendra

Berikut ini `update-index` memperbarui konfigurasi indeks Amazon Kendra. Jika tindakan berhasil, layanan mengirim kembali tidak ada output, kode status HTTP 200, atau kode pengembalian AWS CLI 0. Anda dapat menggunakan `describe-index` untuk melihat konfigurasi dan status indeks.

```
aws kendra update-index \
--id enterpriseindex1 \
--name "new name for Enterprise Edition index 1" \
--description "new description for Enterprise Edition index 1" \
--role-arn arn:aws:iam::my-account-id:role/KendraNewRoleForEnterpriseIndex \
--capacity-units '{"QueryCapacityUnits": 2, "StorageCapacityUnits": 1}' \
--document-metadata-configuration-updates '{"Name": "_document_title",
"Relevance": {"Importance": 6}}, {"Name": "_last_updated_at", "Relevance":
{"Importance": 8}}' \
--user-context-policy "USER_TOKEN" \
--user-token-configurations '{"JsonTokenTypeConfiguration":
{"GroupAttributeField": "groupNameField", "UserNameAttributeField":
"userNameField"}'}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai indeks Amazon Kendra dan konektor sumber data di Panduan](#) Pengembang Amazon Kendra.

- Untuk detail API, lihat [UpdateIndex](#) di Referensi AWS CLI Perintah.

## Contoh Kinesis menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface Kinesis with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags-to-stream**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-stream`.

AWS CLI

Untuk menambahkan tag ke aliran data

`add-tags-to-stream` Contoh berikut memberikan tag dengan kunci `samplekey` dan nilai `example` untuk aliran tertentu.

```
aws kinesis add-tags-to-stream \  
  --stream-name samplestream \  
  --tags samplekey=example
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Streaming Anda di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [AddTagsToStream](#) di Referensi AWS CLI Perintah.

## **create-stream**

Contoh kode berikut menunjukkan cara menggunakan `create-stream`.

### AWS CLI

Untuk membuat aliran data

`create-stream` Contoh berikut membuat aliran data bernama `samplestream` dengan 3 pecahan.

```
aws kinesis create-stream \  
  --stream-name samplestream \  
  --shard-count 3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Stream](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [CreateStream](#) di Referensi AWS CLI Perintah.

## **decrease-stream-retention-period**

Contoh kode berikut menunjukkan cara menggunakan `decrease-stream-retention-period`.

### AWS CLI

Untuk mengurangi periode retensi aliran data

`decrease-stream-retention-period` Contoh berikut mengurangi periode retensi (lamanya waktu catatan data dapat diakses setelah ditambahkan ke aliran) dari aliran bernama `samplestream` hingga 48 jam.

```
aws kinesis decrease-stream-retention-period \  
  --stream-name samplestream \  
  --retention-period-hours 48
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengubah Periode Retensi Data](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [DecreaseStreamRetentionPeriod](#) di Referensi AWS CLI Perintah.

## **delete-stream**

Contoh kode berikut menunjukkan cara menggunakan `delete-stream`.

### AWS CLI

Untuk menghapus aliran data

`delete-stream` Contoh berikut menghapus aliran data yang ditentukan.

```
aws kinesis delete-stream \  
  --stream-name samplestream
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Stream](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [DeleteStream](#) di Referensi AWS CLI Perintah.

## **deregister-stream-consumer**

Contoh kode berikut menunjukkan cara menggunakan `deregister-stream-consumer`.

### AWS CLI

Untuk membatalkan pendaftaran konsumen aliran data

`deregister-stream-consumer` Contoh berikut membatalkan pendaftaran konsumen tertentu dari aliran data yang ditentukan.

```
aws kinesis deregister-stream-consumer \  
  --stream-arn arn:aws:kinesis:us-west-2:123456789012:stream/samplestream \  
  --consumer-name KinesisConsumerApplication
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengembangkan Konsumen dengan Penggemar yang Ditingkatkan Menggunakan API Aliran Data Kinesis di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [DeregisterStreamConsumer](#) di Referensi AWS CLI Perintah.

## **describe-limits**

Contoh kode berikut menunjukkan cara menggunakan `describe-limits`.

### AWS CLI

Untuk menggambarkan batas pecahan

`describe-limits` Contoh berikut menampilkan batas pecahan dan penggunaan untuk AWS akun saat ini.

```
aws kinesis describe-limits
```

Output:

```
{
  "ShardLimit": 500,
  "OpenShardCount": 29
}
```

Untuk informasi selengkapnya, lihat [Menyusun Ulang Stream](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [DescribeLimits](#) di Referensi AWS CLI Perintah.

## **describe-stream-consumer**

Contoh kode berikut menunjukkan cara menggunakan `describe-stream-consumer`.

### AWS CLI

Untuk menggambarkan konsumen aliran data

`describe-stream-consumer` Contoh berikut mengembalikan deskripsi konsumen tertentu, terdaftar dengan aliran data yang ditentukan.

```
aws kinesis describe-stream-consumer \  
  --stream-arn arn:aws:kinesis:us-west-2:012345678912:stream/samplestream \  
  --consumer-name KinesisConsumerApplication
```

Output:

```
{  
  "ConsumerDescription": {  
    "ConsumerName": "KinesisConsumerApplication",  
    "ConsumerARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream/  
consumer/KinesisConsumerApplication:1572383852",  
    "ConsumerStatus": "ACTIVE",  
    "ConsumerCreationTimestamp": 1572383852.0,  
    "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membaca Data dari Amazon Kinesis Data Streams](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [DescribeStreamConsumer](#) di Referensi AWS CLI Perintah.

## describe-stream-summary

Contoh kode berikut menunjukkan cara menggunakan `describe-stream-summary`.

AWS CLI

Untuk menggambarkan ringkasan aliran data

`describe-stream-summary` Contoh berikut memberikan deskripsi yang diringkas (tanpa daftar pecahan) dari aliran data yang ditentukan.

```
aws kinesis describe-stream-summary \  
  --stream-name samplestream
```

Output:

```
{  
  "StreamDescriptionSummary": {  
    "StreamName": "samplestream",  
    "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream",
```

```

    "StreamStatus": "ACTIVE",
    "RetentionPeriodHours": 48,
    "StreamCreationTimestamp": 1572297168.0,
    "EnhancedMonitoring": [
      {
        "ShardLevelMetrics": []
      }
    ],
    "EncryptionType": "NONE",
    "OpenShardCount": 3,
    "ConsumerCount": 0
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Streaming di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [DescribeStreamSummary](#) di Referensi AWS CLI Perintah.

## describe-stream

Contoh kode berikut menunjukkan cara menggunakan `describe-stream`.

### AWS CLI

Untuk menggambarkan aliran data

`describe-stream` Contoh berikut mengembalikan rincian aliran data yang ditentukan.

```

aws kinesis describe-stream \
  --stream-name samplestream

```

Output:

```

{
  "StreamDescription": {
    "Shards": [
      {
        "ShardId": "shardId-000000000000",
        "HashKeyRange": {
          "StartingHashKey": "0",
          "EndingHashKey": "113427455640312821154458202477256070484"
        }
      }
    ]
  }
}

```



```
        "SequenceNumberRange": {
            "StartingSequenceNumber":
"49600871682957036442365024926191073437251060580128653314"
        }
    },
    {
        "ShardId": "shardId-000000000001",
        "HashKeyRange": {
            "StartingHashKey": "113427455640312821154458202477256070485",
            "EndingHashKey": "226854911280625642308916404954512140969"
        },
        "SequenceNumberRange": {
            "StartingSequenceNumber":
"4960087168297933718756355549332609155523708941634633746"
        }
    },
    {
        "ShardId": "shardId-000000000002",
        "HashKeyRange": {
            "StartingHashKey": "226854911280625642308916404954512140970",
            "EndingHashKey": "340282366920938463463374607431768211455"
        },
        "SequenceNumberRange": {
            "StartingSequenceNumber":
"49600871683001637932762086172474144873796357303140614178"
        }
    }
],
"StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream",
"StreamName": "samplestream",
"StreamStatus": "ACTIVE",
"RetentionPeriodHours": 24,
"EnhancedMonitoring": [
    {
        "ShardLevelMetrics": []
    }
],
"EncryptionType": "NONE",
"KeyId": null,
"StreamCreationTimestamp": 1572297168.0
}
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Streaming di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [DescribeStream](#) di Referensi AWS CLI Perintah.

## disable-enhanced-monitoring

Contoh kode berikut menunjukkan cara menggunakan `disable-enhanced-monitoring`.

### AWS CLI

Untuk menonaktifkan pemantauan yang disempurnakan untuk metrik tingkat shard

`disable-enhanced-monitoring` Contoh berikut menonaktifkan pemantauan aliran data Kinesis yang disempurnakan untuk metrik tingkat shard.

```
aws kinesis disable-enhanced-monitoring \  
  --stream-name samplestream --shard-level-metrics ALL
```

Output:

```
{  
  "StreamName": "samplestream",  
  "CurrentShardLevelMetrics": [  
    "IncomingBytes",  
    "OutgoingRecords",  
    "IteratorAgeMilliseconds",  
    "IncomingRecords",  
    "ReadProvisionedThroughputExceeded",  
    "WriteProvisionedThroughputExceeded",  
    "OutgoingBytes"  
  ],  
  "DesiredShardLevelMetrics": []  
}
```

Untuk informasi [selengkapnya, lihat Memantau Aliran di Amazon Kinesis Data Streams](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [DisableEnhancedMonitoring](#) di Referensi AWS CLI Perintah.

## enable-enhanced-monitoring

Contoh kode berikut menunjukkan cara menggunakan `enable-enhanced-monitoring`.

### AWS CLI

Untuk mengaktifkan pemantauan yang disempurnakan untuk metrik tingkat shard

`enable-enhanced-monitoring` Contoh berikut memungkinkan pemantauan aliran data Kinesis yang disempurnakan untuk metrik tingkat shard.

```
aws kinesis enable-enhanced-monitoring \  
  --stream-name samplestream \  
  --shard-level-metrics ALL
```

Output:

```
{  
  "StreamName": "samplestream",  
  "CurrentShardLevelMetrics": [],  
  "DesiredShardLevelMetrics": [  
    "IncomingBytes",  
    "OutgoingRecords",  
    "IteratorAgeMilliseconds",  
    "IncomingRecords",  
    "ReadProvisionedThroughputExceeded",  
    "WriteProvisionedThroughputExceeded",  
    "OutgoingBytes"  
  ]  
}
```

Untuk informasi [selengkapnya, lihat Memantau Aliran di Amazon Kinesis Data Streams](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [EnableEnhancedMonitoring](#) di Referensi AWS CLI Perintah.

## get-records

Contoh kode berikut menunjukkan cara menggunakan `get-records`.

### AWS CLI

Untuk mendapatkan catatan dari pecahan

`get-records` Contoh berikut mendapatkan catatan data dari pecahan Kinesis aliran data menggunakan iterator shard tertentu.

```
aws kinesis get-records \  
  --shard-iterator AAAAAAAAAAF7/0mWD7IuHj1yGv/  
TKuNgx2ukD5xipCY4cy4gU96orWwZwcSXh3K9tAmGYe0ZyLZrvzze0FVf9iN99hUPw/w/  
b0YWYeefNvnf1DYt5XpDJghLKr3DzgzcnkTmMymDP3R+3wRKeuEw6/kdxY2yKJH0veaiekaVc4N2VwK/  
GvaGP2Hh9Fg7N++q0Adg6fIDQPt4p8RpavDbk+A4sL9SWG1
```

Output:

```
{  
  "Records": [],  
  "MillisBehindLatest": 80742000  
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan Konsumen Menggunakan API Kinesis Data Streams AWS dengan SDK for Java](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [GetRecords](#) di Referensi AWS CLI Perintah.

## **get-shard-iterator**

Contoh kode berikut menunjukkan cara menggunakan `get-shard-iterator`.

### AWS CLI

Untuk mendapatkan iterator shard

`get-shard-iterator` Contoh berikut menggunakan tipe iterator `AT_SEQUENCE_NUMBER` shard dan menghasilkan iterator shard untuk mulai membaca catatan data persis dari posisi yang dilambangkan dengan nomor urut yang ditentukan.

```
aws kinesis get-shard-iterator \  
  --stream-name samplestream \  
  --shard-id shardId-000000000001 \  
  --shard-iterator-type LATEST
```

Output:

```
{
```

```
"ShardIterator": "AAAAAAAAAAFEvJjIYI+3jw/4aqqH9FifJ+n48XWTh/
IFIsbILP6o5eDueD39NXNBfpZ10WL5K6ADXk8w+5H+Qhd9cFA9k268CPXCz/kebq1TGYI7Vy
+1UkA9BuN3xvATxMBGxRY3zYK05gqgvaIRn9408SqeEqwhigwZxNwxID3Ej7YYYcxQi8Q/fIrCjGAy/
n2r5Z9G864YpWdfN9upNNQAR/ii0Wks"
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan Konsumen Menggunakan API Kinesis Data Streams AWS dengan SDK for Java](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [GetShardIterator](#) di Referensi AWS CLI Perintah.

## increase-stream-retention-period

Contoh kode berikut menunjukkan cara menggunakan `increase-stream-retention-period`.

### AWS CLI

Untuk meningkatkan periode retensi aliran data

`increase-stream-retention-period` Contoh berikut meningkatkan periode retensi (lamanya waktu catatan data dapat diakses setelah ditambahkan ke aliran) dari aliran yang ditentukan menjadi 168 jam.

```
aws kinesis increase-stream-retention-period \
  --stream-name samplestream \
  --retention-period-hours 168
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengubah Periode Retensi Data](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [IncreaseStreamRetentionPeriod](#) di Referensi AWS CLI Perintah.

## list-shards

Contoh kode berikut menunjukkan cara menggunakan `list-shards`.

### AWS CLI

Untuk membuat daftar pecahan dalam aliran data

`list-shards` Contoh berikut mencantumkan semua pecahan dalam aliran yang ditentukan dimulai dengan pecahan yang IDnya segera mengikuti yang ditentukan `exclusive-start-shard-id` dari `shardId-000000000000`

```
aws kinesis list-shards \  
  --stream-name samplestream \  
  --exclusive-start-shard-id shardId-000000000000
```

Output:

```
{  
  "Shards": [  
    {  
      "ShardId": "shardId-000000000001",  
      "HashKeyRange": {  
        "StartingHashKey": "113427455640312821154458202477256070485",  
        "EndingHashKey": "226854911280625642308916404954512140969"  
      },  
      "SequenceNumberRange": {  
        "StartingSequenceNumber":  
"49600871682979337187563555549332609155523708941634633746"  
      }  
    },  
    {  
      "ShardId": "shardId-000000000002",  
      "HashKeyRange": {  
        "StartingHashKey": "226854911280625642308916404954512140970",  
        "EndingHashKey": "340282366920938463463374607431768211455"  
      },  
      "SequenceNumberRange": {  
        "StartingSequenceNumber":  
"49600871683001637932762086172474144873796357303140614178"  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar Pecahan](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [ListShards](#) di Referensi AWS CLI Perintah.

## list-streams

Contoh kode berikut menunjukkan cara menggunakan `list-streams`.

### AWS CLI

Untuk membuat daftar aliran data

`list-streams` Contoh berikut mencantumkan semua aliran data aktif di akun dan wilayah saat ini.

```
aws kinesisis list-streams
```

Output:

```
{
  "StreamNames": [
    "samplestream",
    "samplestream1"
  ]
}
```

Untuk informasi selengkapnya, lihat [Daftar Aliran](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [ListStreams](#) di Referensi AWS CLI Perintah.

## list-tags-for-stream

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-stream`.

### AWS CLI

Untuk mencantumkan tag untuk aliran data

`list-tags-for-stream` Contoh berikut mencantumkan tag yang dilampirkan ke aliran data yang ditentukan.

```
aws kinesisis list-tags-for-stream \
  --stream-name samplestream
```

Output:

```
{
  "Tags": [
    {
      "Key": "samplekey",
      "Value": "example"
    }
  ],
  "HasMoreTags": false
}
```

Untuk informasi selengkapnya, lihat [Menandai Streaming Anda di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [ListTagsForStream](#) di Referensi AWS CLI Perintah.

## **merge-shards**

Contoh kode berikut menunjukkan cara menggunakan `merge-shards`.

### AWS CLI

Untuk menggabungkan pecahan

`merge-shards` Contoh berikut menggabungkan dua pecahan yang berdekatan dengan ID `ShardId-000000000000` dan `ShardId-0000000001` dalam aliran data yang ditentukan dan menggabungkannya menjadi pecahan tunggal.

```
aws kinesis merge-shards \
  --stream-name samplestream \
  --shard-to-merge shardId-000000000000 \
  --adjacent-shard-to-merge shardId-000000000001
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggabungkan Dua Pecahan](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [MergeShards](#) di Referensi AWS CLI Perintah.

## **put-record**

Contoh kode berikut menunjukkan cara menggunakan `put-record`.



## AWS CLI

Untuk menulis catatan ke dalam aliran data

`put-record` Contoh berikut menulis catatan data tunggal ke dalam aliran data yang ditentukan menggunakan kunci partisi yang ditentukan.

```
aws kinesis put-record \  
  --stream-name samplestream \  
  --data sampledatarecord \  
  --partition-key samplepartitionkey
```

Output:

```
{  
  "ShardId": "shardId-000000000009",  
  "SequenceNumber": "49600902273357540915989931256901506243878407835297513618",  
  "EncryptionType": "KMS"  
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan Produsen Menggunakan API Amazon Kinesis Data Streams AWS dengan SDK for Java](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [PutRecord](#) di Referensi AWS CLI Perintah.

## `put-records`

Contoh kode berikut menunjukkan cara menggunakan `put-records`.

## AWS CLI

Untuk menulis beberapa catatan ke dalam aliran data

`put-records` Contoh berikut menulis catatan data menggunakan kunci partisi tertentu dan catatan data lain menggunakan kunci partisi yang berbeda dalam satu panggilan.

```
aws kinesis put-records \  
  --stream-name samplestream \  
  --records Data=blob1,PartitionKey=partitionkey1  
  Data=blob2,PartitionKey=partitionkey2
```

**Output:**

```
{
  "FailedRecordCount": 0,
  "Records": [
    {
      "SequenceNumber":
"49600883331171471519674795588238531498465399900093808706",
      "ShardId": "shardId-000000000004"
    },
    {
      "SequenceNumber":
"49600902273357540915989931256902715169698037101720764562",
      "ShardId": "shardId-000000000009"
    }
  ],
  "EncryptionType": "KMS"
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan Produsen Menggunakan API Amazon Kinesis Data Streams AWS dengan SDK for Java](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [PutRecords](#) di Referensi AWS CLI Perintah.

**register-stream-consumer**

Contoh kode berikut menunjukkan cara menggunakan `register-stream-consumer`.

**AWS CLI**

Untuk mendaftarkan konsumen aliran data

`register-stream-consumer` Contoh berikut mendaftarkan konsumen yang dipanggil `KinesisConsumerApplication` dengan aliran data yang ditentukan.

```
aws kinesis register-stream-consumer \
  --stream-arn arn:aws:kinesis:us-west-2:012345678912:stream/samplestream \
  --consumer-name KinesisConsumerApplication
```

**Output:**

```
{
  "Consumer": {
    "ConsumerName": "KinesisConsumerApplication",
    "ConsumerARN": "arn:aws:kinesis:us-west-2: 123456789012:stream/samplestream/
consumer/KinesisConsumerApplication:1572383852",
    "ConsumerStatus": "CREATING",
    "ConsumerCreationTimestamp": 1572383852.0
  }
}
```

Untuk informasi selengkapnya, lihat [Mengembangkan Konsumen dengan Penggemar yang Ditingkatkan Menggunakan API Aliran Data Kinesis di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [RegisterStreamConsumer](#) di Referensi AWS CLI Perintah.

## **remove-tags-from-stream**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-stream`.

### AWS CLI

Untuk menghapus tag dari aliran data

`remove-tags-from-stream` Contoh berikut menghapus tag dengan kunci yang ditentukan dari aliran data yang ditentukan.

```
aws kinesis remove-tags-from-stream \
  --stream-name samplestream \
  --tag-keys samplekey
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Streaming Anda di Panduan Pengembang Amazon Kinesis Data Streams](#).

- Untuk detail API, lihat [RemoveTagsFromStream](#) di Referensi AWS CLI Perintah.

## **split-shard**

Contoh kode berikut menunjukkan cara menggunakan `split-shard`.

## AWS CLI

Untuk membagi pecahan

`split-shard` Contoh berikut membagi pecahan yang ditentukan menjadi dua pecahan baru menggunakan kunci hash awal baru 10.

```
aws kinesis split-shard \  
  --stream-name samplestream \  
  --shard-to-split shardId-000000000000 \  
  --new-starting-hash-key 10
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memisahkan Shard di Panduan](#) Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [SplitShard](#) di Referensi AWS CLI Perintah.

## **start-stream-encryption**

Contoh kode berikut menunjukkan cara menggunakan `start-stream-encryption`.

## AWS CLI

Untuk mengaktifkan enkripsi aliran data

`start-stream-encryption` Contoh berikut memungkinkan enkripsi sisi server untuk aliran tertentu, menggunakan kunci KMS yang ditentukan AWS .

```
aws kinesis start-stream-encryption \  
  --encryption-type KMS \  
  --key-id arn:aws:kms:us-west-2:012345678912:key/a3c4a7cd-728b-45dd-  
b334-4d3eb496e452 \  
  --stream-name samplestream
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perlindungan Data di Amazon Kinesis Data](#) Streams di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [StartStreamEncryption](#) di Referensi AWS CLI Perintah.

## stop-stream-encryption

Contoh kode berikut menunjukkan cara menggunakan `stop-stream-encryption`.

### AWS CLI

Untuk menonaktifkan enkripsi aliran data

`stop-stream-encryption` Contoh berikut menonaktifkan enkripsi sisi server untuk aliran tertentu, menggunakan kunci KMS yang ditentukan. AWS

```
aws kinesis start-stream-encryption \  
  --encryption-type KMS \  
  --key-id arn:aws:kms:us-west-2:012345678912:key/a3c4a7cd-728b-45dd-  
b334-4d3eb496e452 \  
  --stream-name samplestream
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perlindungan Data di Amazon Kinesis Data Streams](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [StopStreamEncryption](#) di Referensi AWS CLI Perintah.

## update-shard-count

Contoh kode berikut menunjukkan cara menggunakan `update-shard-count`.

### AWS CLI

Untuk memperbarui jumlah pecahan dalam aliran data

`update-shard-count` Contoh berikut memperbarui jumlah pecahan dari aliran data yang ditentukan ke 6. Contoh ini menggunakan penskalaan seragam, yang menciptakan pecahan dengan ukuran yang sama.

```
aws kinesis update-shard-count \  
  --stream-name samplestream \  
  --shard-count 6
```

```
--scaling-type UNIFORM_SCALING \  
--target-shard-count 6
```

Output:

```
{  
  "StreamName": "samplestream",  
  "CurrentShardCount": 3,  
  "TargetShardCount": 6  
}
```

Untuk informasi selengkapnya, lihat [Menyusun Ulang Stream](#) di Panduan Pengembang Amazon Kinesis Data Streams.

- Untuk detail API, lihat [UpdateShardCount](#) di Referensi AWS CLI Perintah.

## AWS KMS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS KMS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **cancel-key-deletion**

Contoh kode berikut menunjukkan cara menggunakan `cancel-key-deletion`.

## AWS CLI

Untuk membatalkan penghapusan terjadwal kunci KMS yang dikelola pelanggan

`cancel-key-deletion` Contoh berikut membatalkan penghapusan terjadwal kunci KMS yang dikelola pelanggan.

```
aws kms cancel-key-deletion \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{  
  "KeyId": "arn:aws:kms:us-  
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

Ketika `cancel-key-deletion` perintah berhasil, penghapusan terjadwal dibatalkan. Namun, status kunci dari kunci KMS adalah `Disabled`, jadi Anda tidak dapat menggunakan kunci KMS dalam operasi kriptografi. Untuk mengembalikan fungsinya, gunakan `enable-key` perintah.

Untuk informasi selengkapnya, lihat [Menjadwalkan dan membatalkan penghapusan kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [CancelKeyDeletion](#) di Referensi AWS CLI Perintah.

## **connect-custom-key-store**

Contoh kode berikut menunjukkan cara menggunakan `connect-custom-key-store`.

## AWS CLI

Untuk menghubungkan toko kunci kustom

`connect-custom-key-store` Contoh berikut menghubungkan kembali toko kunci kustom yang ditentukan. Anda dapat menggunakan perintah seperti ini untuk menghubungkan toko kunci khusus untuk pertama kalinya atau untuk menghubungkan kembali toko kunci yang terputus.

Anda dapat menggunakan perintah ini untuk menghubungkan toko kunci AWS CloudHSM atau toko kunci eksternal.

```
aws kms connect-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0
```

Perintah ini tidak memberikan output apa pun. Untuk memverifikasi bahwa perintah itu efektif, gunakan `describe-custom-key-stores` perintah.

Untuk informasi tentang menghubungkan penyimpanan kunci AWS CloudHSM, [lihat Menyambungkan dan memutuskan penyimpanan kunci AWS CloudHSM di Panduan Pengembang Layanan Manajemen Kunci.AWS](#)

Untuk informasi tentang menghubungkan penyimpanan kunci eksternal, lihat [Menghubungkan dan memutuskan penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ConnectCustomKeyStore](#) di Referensi AWS CLI Perintah.

## **create-alias**

Contoh kode berikut menunjukkan cara menggunakan `create-alias`.

### AWS CLI

Untuk membuat alias untuk kunci KMS

`create-alias` Perintah berikut membuat alias bernama `example-alias` untuk kunci KMS diidentifikasi oleh ID kunci. `1234abcd-12ab-34cd-56ef-1234567890ab`

Nama alias harus dimulai dengan `alias/`. Jangan gunakan nama alias yang dimulai dengan `alias/aws`; ini dicadangkan untuk digunakan oleh AWS.

```
aws kms create-alias \  
  --alias-name alias/example-alias \  
  --target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak mengembalikan output apa pun. Untuk melihat alias baru, gunakan `list-aliases` perintah.

Untuk informasi selengkapnya, lihat [Menggunakan alias](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.



- Untuk detail API, lihat [CreateAlias](#) di Referensi AWS CLI Perintah.

## create-custom-key-store

Contoh kode berikut menunjukkan cara menggunakan `create-custom-key-store`.

### AWS CLI

Contoh 1: Untuk membuat toko kunci AWS CloudHSM

`create-custom-key-store` Contoh berikut membuat penyimpanan kunci AWS CloudHSM yang didukung oleh klaster AWS CloudHSM menggunakan parameter yang diperlukan. Anda juga dapat menambahkan `custom-key-store-type` parameter with the default value: `AWS_CLOUDHSM`.

Untuk menentukan input file untuk `trust-anchor-certificate` perintah di AWS CLI, `file://` awalan diperlukan.

```
aws kms create-custom-key-store \  
  --custom-key-store-name ExampleCloudHSMKeyStore \  
  --cloud-hsm-cluster-id cluster-1a23b4cdefg \  
  --key-store-password kmsPswd \  
  --trust-anchor-certificate file://customerCA.crt
```

Output:

```
{  
  "CustomKeyId": cks-1234567890abcdef0  
}
```

Untuk informasi selengkapnya, lihat [Membuat penyimpanan kunci AWS CloudHSM](#) di Panduan AWS Pengembang Layanan Manajemen Kunci.

Contoh 2: Untuk membuat penyimpanan kunci eksternal dengan konektivitas titik akhir publik

`create-custom-key-store` Contoh berikut membuat penyimpanan kunci eksternal (XKS) yang berkomunikasi dengan AWS KMS melalui internet.

Dalam contoh ini, `XksProxyUriPath` menggunakan awalan opsional dari `example-prefix`

CATATAN: Jika Anda menggunakan AWS CLI versi 1.0, jalankan perintah berikut sebelum menentukan parameter dengan nilai HTTP atau HTTPS, seperti parameter. `XksProxyUriEndpoint`

```
aws configure set cli_follow_urlparam false
```

Jika tidak, AWS CLI versi 1.0 menggantikan nilai parameter dengan konten yang ditemukan di alamat URI tersebut.

```
aws kms create-custom-key-store \  
  --custom-key-store-name ExamplePublicEndpointXKS \  
  --custom-key-store-type EXTERNAL_KEY_STORE \  
  --xks-proxy-connectivity PUBLIC_ENDPOINT \  
  --xks-proxy-uri-endpoint "https://myproxy.xks.example.com" \  
  --xks-proxy-uri-path "/example-prefix/kms/xks/v1" \  
  --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,  
RawSecretAccessKey=DXjSUawne12fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Output:

```
{  
  "CustomKeyId": cks-2234567890abcdef0  
}
```

Untuk informasi selengkapnya, lihat [Membuat penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 3: Untuk membuat penyimpanan kunci eksternal dengan konektivitas layanan titik akhir VPC

`create-custom-key-store` Contoh berikut membuat penyimpanan kunci eksternal (XKS) yang menggunakan layanan endpoint Amazon VPC untuk berkomunikasi dengan KMS. AWS

CATATAN: Jika Anda menggunakan AWS CLI versi 1.0, jalankan perintah berikut sebelum menentukan parameter dengan nilai HTTP atau HTTPS, seperti parameter. `XksProxyUriEndpoint`

```
aws configure set cli_follow_urlparam false
```

Jika tidak, AWS CLI versi 1.0 menggantikan nilai parameter dengan konten yang ditemukan di alamat URI tersebut.

```
aws kms create-custom-key-store \  
  --custom-key-store-name ExampleVPCEndpointXKS \  
  --custom-key-store-type EXTERNAL_KEY_STORE \  
  --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \  
  --xks-proxy-uri-endpoint "https://myproxy-private.xks.example.com" \  
  --xks-proxy-uri-path "/kms/xks/v1" \  
  --xks-proxy-vpc-endpoint-service-name "com.amazonaws.vpce.us-east-1.vpce-svc-  
example1" \  
  --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,  
RawSecretAccessKey=DXjSUawne12fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

### Output:

```
{  
  "CustomKeyId": cks-3234567890abcdef0  
}
```

Untuk informasi selengkapnya, lihat [Membuat penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [CreateCustomKeyStore](#) di Referensi AWS CLI Perintah.

## create-grant

Contoh kode berikut menunjukkan cara menggunakan `create-grant`.

### AWS CLI

Untuk membuat hibah

`create-grant` Contoh berikut membuat hibah yang memungkinkan `exampleUser` pengguna untuk menggunakan `decrypt` perintah pada kunci KMS `1234abcd-12ab-34cd-56ef-1234567890ab` contoh. Kepala sekolah yang pensiun adalah `adminRole` perannya. Hibah menggunakan batasan `EncryptionContextSubset` hibah untuk mengizinkan izin ini hanya jika konteks enkripsi dalam `decrypt` permintaan menyertakan pasangan nilai `"Department": "IT"` kunci.

```
aws kms create-grant \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --grantee-principal arn:aws:iam::123456789012:user/exampleUser \  
  --encryption-context-subset '{"Department": "IT"}'
```

```
--operations Decrypt \  
--constraints EncryptionContextSubset={Department=IT} \  
--retiring-principal arn:aws:iam::123456789012:role/adminRole
```

### Output:

```
{  
  "GrantId": "1a2b3c4d2f5e69f440bae30eaec9570bb1fb7358824f9ddfa1aa5a0dab1a59b2",  
  "GrantToken": "<grant token here>"  
}
```

Untuk melihat informasi terperinci tentang hibah, gunakan `list-grants` perintah.

Untuk informasi selengkapnya, lihat [Hibah di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Utama.

- Untuk detail API, lihat [CreateGrant](#) di Referensi AWS CLI Perintah.

## create-key

Contoh kode berikut menunjukkan cara menggunakan `create-key`.

### AWS CLI

Contoh 1: Untuk membuat kunci KMS yang dikelola pelanggan di AWS KMS

`create-key` Contoh berikut membuat kunci KMS enkripsi simetris.

Untuk membuat kunci KMS dasar, kunci enkripsi simetris, Anda tidak perlu menentukan parameter apa pun. Nilai default untuk parameter tersebut membuat kunci enkripsi simetris.

Karena perintah ini tidak menentukan kebijakan kunci, kunci KMS mendapatkan [kebijakan kunci default untuk kunci](#) KMS yang dibuat secara terprogram. Untuk melihat kebijakan kunci, gunakan `get-key-policy` perintah. Untuk mengubah kebijakan kunci, gunakan `put-key-policy` perintah.

```
aws kms create-key
```

`create-key` Perintah mengembalikan metadata kunci, termasuk ID kunci dan ARN dari kunci KMS baru. Anda dapat menggunakan nilai-nilai ini untuk mengidentifikasi kunci KMS dalam

operasi AWS KMS lainnya. Outputnya tidak termasuk tag. Untuk melihat tag untuk kunci KMS, gunakan tombol. `list-resource-tags` command

Output:

```
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": "2017-07-05T14:04:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "Description": "",
    "Enabled": true,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "AWS_KMS"
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ]
  }
}
```

Catatan: `create-key` Perintah tidak memungkinkan Anda menentukan alias, Untuk membuat alias untuk kunci KMS baru, gunakan perintah. `create-alias`

Untuk informasi selengkapnya, lihat [Membuat AWS kunci](#) di Panduan Pengembang Layanan Manajemen Kunci.

Contoh 2: Untuk membuat kunci KMS RSA asimetris untuk enkripsi dan dekripsi

`create-key` Contoh berikut membuat kunci KMS yang berisi asimetris RSA key pair untuk enkripsi dan dekripsi.

```
aws kms create-key \
  --key-spec RSA_4096 \
  --key-usage ENCRYPT_DECRYPT
```

**Output:**

```
{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2021-04-05T14:04:55-07:00",
    "CustomerMasterKeySpec": "RSA_4096",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "RSAES_OAEP_SHA_1",
      "RSAES_OAEP_SHA_256"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "RSA_4096",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "AWS_KMS"
  }
}
```

Untuk informasi selengkapnya, lihat [Kunci asimetris di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 3: Untuk membuat kunci KMS kurva elips asimetris untuk penandatanganan dan verifikasi

Untuk membuat kunci KMS asimetris yang berisi key pair asymmetric elliptic curve (ECC) untuk penandatanganan dan verifikasi. --key-usageParameter diperlukan meskipun SIGN\_VERIFY merupakan satu-satunya nilai yang valid untuk kunci ECC KMS.

```
aws kms create-key \
  --key-spec ECC_NIST_P521 \
  --key-usage SIGN_VERIFY
```

**Output:**

```
{
```

```

    "KeyMetadata": {
      "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "AWSAccountId": "111122223333",
      "CreationDate": "2019-12-02T07:48:55-07:00",
      "CustomerMasterKeySpec": "ECC_NIST_P521",
      "Description": "",
      "Enabled": true,
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "KeyManager": "CUSTOMER",
      "KeySpec": "ECC_NIST_P521",
      "KeyState": "Enabled",
      "KeyUsage": "SIGN_VERIFY",
      "MultiRegion": false,
      "Origin": "AWS_KMS",
      "SigningAlgorithms": [
        "ECDSA_SHA_512"
      ]
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Kunci asimetris di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

#### Contoh 4: Untuk membuat kunci HMAC KMS

`create-key` Contoh berikut membuat kunci KMS HMAC 384-bit. `GENERATE_VERIFY_MAC` nilai untuk `--key-usage` parameter diperlukan meskipun itu satu-satunya nilai yang valid untuk kunci HMAC KMS.

```

aws kms create-key \
  --key-spec HMAC_384 \
  --key-usage GENERATE_VERIFY_MAC

```

#### Output:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2022-04-05T14:04:55-07:00",

```

```

    "CustomerMasterKeySpec": "HMAC_384",
    "Description": "",
    "Enabled": true,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "HMAC_384",
    "KeyState": "Enabled",
    "KeyUsage": "GENERATE_VERIFY_MAC",
    "MacAlgorithms": [
      "HMAC_SHA_384"
    ],
    "MultiRegion": false,
    "Origin": "AWS_KMS"
  }
}

```

Untuk informasi selengkapnya, lihat [kunci HMAC di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 4: Untuk membuat kunci KMS primer Multi-wilayah

`create-key` Contoh berikut membuat kunci enkripsi simetris primer Multi-region. Karena nilai default untuk semua parameter membuat kunci enkripsi simetris, hanya `--multi-region` parameter yang diperlukan untuk kunci KMS ini. Di AWS CLI, untuk menunjukkan bahwa parameter Boolean benar, cukup tentukan nama parameternya.

```

aws kms create-key \
  --multi-region

```

Output:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-west-2:111122223333:key/mrk-1234abcd12ab34cd56ef12345678990ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2021-09-02T016:15:21-09:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ]
  }
}

```



```

    ],
    "KeyId": "mrk-1234abcd12ab34cd56ef12345678990ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": true,
    "MultiRegionConfiguration": {
      "MultiRegionKeyType": "PRIMARY",
      "PrimaryKey": {
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef12345678990ab",
        "Region": "us-west-2"
      },
      "ReplicaKeys": []
    },
    "Origin": "AWS_KMS"
  }
}

```

Untuk informasi selengkapnya, lihat [Kunci asimetris di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 5: Untuk membuat kunci KMS untuk bahan kunci impor

`create-key` berikut membuat membuat kunci KMS tanpa bahan kunci. Ketika operasi selesai, Anda dapat mengimpor materi kunci Anda sendiri ke dalam kunci KMS. Untuk membuat kunci KMS ini, atur `--origin` parameter ke `EXTERNAL`.

```

aws kms create-key \
  --origin EXTERNAL

```

Output:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2019-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "Description": "",

```

```

    "Enabled": false,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "PendingImport",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "EXTERNAL"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengimpor materi kunci dalam kunci AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 6: Untuk membuat kunci KMS di toko kunci AWS CloudHSM

`create-key` Contoh berikut membuat membuat kunci KMS di toko kunci AWS CloudHSM tertentu. Operasi membuat kunci KMS dan metadatanya di AWS KMS dan membuat materi kunci di kluster CloudHSM yang terkait dengan penyimpanan kunci AWS kustom. Parameter `--custom-key-store-id` dan `--origin` diperlukan.

```

aws kms create-key \
  --origin AWS_CLOUDHSM \
  --custom-key-store-id cks-1234567890abcdef0

```

Output:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CloudHsmClusterId": "cluster-1a23b4cdefg",
    "CreationDate": "2019-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "CustomKeyId": "cks-1234567890abcdef0",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [

```

```

        "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "AWS_CLOUDHSM"
}
}

```

Untuk informasi selengkapnya, lihat [AWS Penyimpanan kunci CloudHSM](#) di Panduan AWS Pengembang Layanan Manajemen Kunci.

Contoh 7: Untuk membuat kunci KMS di toko kunci eksternal

`create-key` berikut membuat membuat kunci KMS di toko kunci eksternal yang ditentukan. `--xks-key-id` Parameter `--custom-key-store-id` `--origin`, dan diperlukan dalam perintah ini.

`--xks-key-id` Parameter menentukan ID kunci enkripsi simetris yang ada di pengelola kunci eksternal Anda. Kunci ini berfungsi sebagai bahan kunci eksternal untuk kunci KMS. Nilai `--origin` parameter `EXTERNAL_KEY_STORE` `custom-key-store-id` harus. Parameter harus mengidentifikasi penyimpanan kunci eksternal yang terhubung ke proxy penyimpanan kunci eksternal.

```

aws kms create-key \
  --origin EXTERNAL_KEY_STORE \
  --custom-key-store-id cks-9876543210fedcba9 \
  --xks-key-id bb8562717f809024

```

Output:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2022-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",

```

```

    "CustomKeyStoreId": "cks-9876543210fedcba9",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "EXTERNAL_KEY_STORE",
    "XksKeyConfiguration": {
      "Id": "bb8562717f809024"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [CreateKey](#) di Referensi AWS CLI Perintah.

## decrypt

Contoh kode berikut menunjukkan cara menggunakan `decrypt`.

### AWS CLI

Contoh 1: Untuk mendekripsi pesan terenkripsi dengan kunci KMS simetris (Linux dan macOS)

Contoh `decrypt` perintah berikut menunjukkan cara yang disarankan untuk mendekripsi data dengan CLI. AWS Versi ini menunjukkan cara mendekripsi data di bawah kunci KMS simetris.

Berikan ciphertext dalam file. Dalam nilai `--ciphertext-blob` parameter, gunakan `fileb://` awalan, yang memberitahu CLI untuk membaca data dari file biner. Jika file tidak ada di direktori saat ini, ketik path lengkap ke file. Untuk informasi lebih lanjut tentang membaca nilai parameter AWS CLI dari file, lihat Memuat parameter AWS CLI dari file < <https://docs.aws.amazon.com/cli/latest/userguide/cli-usage-parameters-file.html> > di Panduan Pengguna Antarmuka Baris AWS Perintah dan Praktik Terbaik untuk Parameter File Lokal < <https://aws.amazon.com/blogs/developer/best-practices-for-local-file-parameters/> > di Blog Alat Baris Perintah. Tentukan

kunci KMS untuk mendekripsi ciphertext. Parameter tidak diperlukan AWS saat mendekripsi dengan kunci KMS simetris. `--key-id` AWS KMS bisa mendapatkan ID kunci dari kunci KMS yang digunakan untuk mengenkripsi data dari metadata dalam ciphertext. Tetapi selalu merupakan praktik terbaik untuk menentukan kunci KMS yang Anda gunakan. Praktek ini memastikan bahwa Anda menggunakan kunci KMS yang Anda inginkan, dan mencegah Anda dari secara tidak sengaja mendekripsi ciphertext menggunakan kunci KMS yang tidak Anda percayai. Minta output plaintext sebagai nilai teks. Parameter `--query` memberitahu CLI untuk mendapatkan hanya nilai bidang dari output. Plaintext `--output` Parameter mengembalikan output sebagai `text.base64-decode` plaintext dan menyimpannya dalam file. Contoh berikut pipa (`|`) nilai Plaintext parameter ke utilitas Base64, yang menerjemahkan itu. Kemudian, itu mengalihkan (`>`) output yang diterjemahkan ke file. `ExamplePlaintext`

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms decrypt \  
  --ciphertext-blob fileb://ExampleEncryptedFile \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --output text \  
  --query Plaintext | base64 \  
  --decode > ExamplePlaintextFile
```

Perintah ini tidak menghasilkan output. Output dari `decrypt` perintah adalah base64-decoded dan disimpan dalam file.

Untuk informasi selengkapnya, lihat [Mendekripsi](#) di Referensi API Layanan Manajemen AWS Kunci.

Contoh 2: Untuk mendekripsi pesan terenkripsi dengan tombol KMS simetris (prompt perintah Windows)

Contoh berikut adalah sama dengan yang sebelumnya kecuali bahwa ia menggunakan `certutil` utilitas untuk Base64-decode data plaintext. Prosedur ini membutuhkan dua perintah, seperti yang ditunjukkan pada contoh berikut.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms decrypt ^
```

```
--ciphertext-blob fileb://ExampleEncryptedFile ^
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab ^
--output text ^
--query Plaintext > ExamplePlaintextFile.base64
```

Jalankan perintah certutil.

```
certutil -decode ExamplePlaintextFile.base64 ExamplePlaintextFile
```

Output:

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Untuk informasi selengkapnya, lihat [Mendekripsi](#) di Referensi API Layanan Manajemen AWS Kunci.

Contoh 3: Untuk mendekripsi pesan terenkripsi dengan kunci KMS asimetris (Linux dan macOS)

Contoh decrypt perintah berikut menunjukkan cara mendekripsi data yang dienkripsi di bawah kunci KMS asimetris RSA.

Saat menggunakan kunci KMS asimetris, `encryption-algorithm` parameter, yang menentukan algoritma yang digunakan untuk mengenkripsi plaintext, diperlukan.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms decrypt \
  --ciphertext-blob fileb://ExampleEncryptedFile \
  --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
  --encryption-algorithm RSAES_OAEP_SHA_256 \
  --output text \
  --query Plaintext | base64 \
  --decode > ExamplePlaintextFile
```

Perintah ini tidak menghasilkan output. Output dari decrypt perintah adalah base64-decoded dan disimpan dalam file.

Untuk informasi selengkapnya, lihat [Kunci asimetris di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [Mendekripsi](#) di Referensi AWS CLI Perintah.

## **delete-alias**

Contoh kode berikut menunjukkan cara menggunakan `delete-alias`.

### AWS CLI

Untuk menghapus alias AWS KMS

`delete-alias` Contoh berikut menghapus `alias/example-alias` alias. Nama alias harus dimulai dengan `alias/`.

```
aws kms delete-alias \  
  --alias-name alias/example-alias
```

Perintah ini tidak menghasilkan output. Untuk menemukan alias, gunakan `list-aliases` perintah.

Untuk informasi selengkapnya, lihat [Menghapus alias](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [DeleteAlias](#) di Referensi AWS CLI Perintah.

## **delete-custom-key-store**

Contoh kode berikut menunjukkan cara menggunakan `delete-custom-key-store`.

### AWS CLI

Untuk menghapus toko kunci kustom

`delete-custom-key-store` Contoh berikut menghapus toko kunci kustom yang ditentukan.

Menghapus penyimpanan kunci AWS CloudHSM tidak berpengaruh pada kluster CloudHSM terkait. Menghapus penyimpanan kunci eksternal tidak berpengaruh pada proxy penyimpanan kunci eksternal terkait, pengelola kunci eksternal, atau kunci eksternal.

CATATAN: Sebelum Anda dapat menghapus toko kunci khusus, Anda harus menjadwalkan penghapusan semua kunci KMS di toko kunci khusus dan kemudian menunggu kunci KMS tersebut dihapus. Kemudian, Anda harus memutuskan sambungan toko kunci khusus. Untuk bantuan menemukan kunci KMS di toko kunci kustom Anda, lihat [Menghapus penyimpanan kunci \(API\) CloudHSM](#) di Panduan AWS Pengembang Layanan Manajemen Kunci. AWS

```
delete-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0
```

Perintah ini tidak memberikan output apa pun. Untuk memverifikasi bahwa penyimpanan kunci khusus dihapus, gunakan `describe-custom-key-stores` perintah.

Untuk informasi tentang menghapus penyimpanan kunci AWS CloudHSM, lihat [Menghapus penyimpanan kunci AWS CloudHSM di Panduan Pengembang Layanan](#) Manajemen Kunci.AWS

Untuk informasi tentang menghapus penyimpanan kunci eksternal, lihat [Menghapus penyimpanan kunci eksternal di Panduan](#) Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [DeleteCustomKeyStore](#) di Referensi AWS CLI Perintah.

## **delete-imported-key-material**

Contoh kode berikut menunjukkan cara menggunakan `delete-imported-key-material`.

### AWS CLI

Untuk menghapus materi kunci yang diimpor dari kunci KMS

`delete-imported-key-material` Contoh berikut menghapus materi kunci yang telah diimpor ke kunci KMS.

```
aws kms delete-imported-key-material \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output. Untuk memverifikasi bahwa materi kunci dihapus, gunakan `describe-key` perintah untuk mencari status kunci `PendingImport` atau `PendingDeletion`.

Untuk informasi selengkapnya, lihat [Menghapus materi kunci yang diimpor](https://docs.aws.amazon.com/kms/latest/developerguide/importing-keys-delete-key-material.html) < <https://docs.aws.amazon.com/kms/latest/developerguide/importing-keys-delete-key-material.html> > di Panduan Pengembang Layanan Manajemen Kunci.AWS



- Untuk detail API, lihat [DeleteImportedKeyMaterial](#) di Referensi AWS CLI Perintah.

## describe-custom-key-stores

Contoh kode berikut menunjukkan cara menggunakan `describe-custom-key-stores`.

### AWS CLI

Contoh 1: Untuk mendapatkan detail tentang toko kunci AWS CloudHSM

`describe-custom-key-store` Contoh berikut menampilkan rincian tentang penyimpanan kunci AWS CloudHSM yang ditentukan. Perintahnya sama untuk semua jenis penyimpanan kunci khusus, tetapi outputnya berbeda dengan jenis penyimpanan kunci dan, untuk penyimpanan kunci eksternal, opsi konektivitasnya.

Secara default, perintah ini menampilkan informasi tentang semua toko kunci khusus di akun dan Wilayah. Untuk menampilkan informasi tentang penyimpanan kunci khusus tertentu, gunakan `custom-key-store-id` parameter `custom-key-store-name` atau.

```
aws kms describe-custom-key-stores \
  --custom-key-store-name ExampleCloudHSMKeyStore
```

Output dari perintah ini mencakup rincian berguna tentang penyimpanan kunci AWS CloudHSM termasuk status koneksinya (`ConnectionState`). Jika status koneksi `FAILED`, output mencakup `ConnectionErrorCode` bidang yang menjelaskan masalah.

Output:

```
{
  "CustomKeyStores": [
    {
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "ConnectionState": "CONNECTED",
      "CreationDate": "2022-04-05T14:04:55-07:00",
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleExternalKeyStore",
      "TrustAnchorCertificate": "<certificate appears here>"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat penyimpanan kunci AWS CloudHSM](#) di Panduan AWS Pengembang Layanan Manajemen Kunci.

Contoh 2: Untuk mendapatkan detail tentang penyimpanan kunci eksternal dengan konektivitas titik akhir publik

`describe-custom-key-store` Contoh berikut menampilkan rincian tentang penyimpanan kunci eksternal yang ditentukan. Perintahnya sama untuk semua jenis penyimpanan kunci khusus, tetapi outputnya berbeda dengan jenis penyimpanan kunci dan, untuk penyimpanan kunci eksternal, opsi konektivitasnya.

Secara default, perintah ini menampilkan informasi tentang semua toko kunci khusus di akun dan Wilayah. Untuk menampilkan informasi tentang penyimpanan kunci khusus tertentu, gunakan `custom-key-store-id` parameter `custom-key-store-name` atau.

```
aws kms describe-custom-key-stores \  
  --custom-key-store-id cks-9876543210fedcba9
```

Output dari perintah ini mencakup rincian berguna tentang penyimpanan kunci eksternal termasuk status koneksi (`ConnectionState`). Jika status koneksi `FAILED`, output mencakup `ConnectionErrorCode` bidang yang menjelaskan masalah.

Output:

```
{  
  "CustomKeyStores": [  
    {  
      "CustomKeyStoreId": "cks-9876543210fedcba9",  
      "CustomKeyStoreName": "ExampleXKS",  
      "ConnectionState": "CONNECTED",  
      "CreationDate": "2022-12-02T07:48:55-07:00",  
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",  
      "XksProxyConfiguration": {  
        "AccessKeyId": "ABCDE12345670EXAMPLE",  
        "Connectivity": "PUBLIC_ENDPOINT",  
        "UriEndpoint": "https://myproxy.xks.example.com",  
        "UriPath": "/example-prefix/kms/xks/v1"  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 3: Untuk mendapatkan detail tentang penyimpanan kunci eksternal dengan konektivitas layanan titik akhir VPC

`describe-custom-key-store` Contoh berikut menampilkan rincian tentang penyimpanan kunci eksternal yang ditentukan. Perintahnya sama untuk semua jenis penyimpanan kunci khusus, tetapi outputnya berbeda dengan jenis penyimpanan kunci dan, untuk penyimpanan kunci eksternal, opsi konektivitasnya.

Secara default, perintah ini menampilkan informasi tentang semua toko kunci khusus di akun dan Wilayah. Untuk menampilkan informasi tentang penyimpanan kunci khusus tertentu, gunakan `custom-key-store-id` parameter `custom-key-store-name` atau.

```
aws kms describe-custom-key-stores \
  --custom-key-store-id cks-2234567890abcdef0
```

Output dari perintah ini mencakup rincian berguna tentang penyimpanan kunci eksternal termasuk status koneksi (`ConnectionState`). Jika status koneksi `FAILED`, output mencakup `ConnectionErrorCode` bidang yang menjelaskan masalah.

Output:

```
{
  "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-3234567890abcdef0",
      "CustomKeyStoreName": "ExampleVPCExternalKeyStore",
      "ConnectionState": "CONNECTED",
      "CreationDate": "2022-12-22T07:48:55-07:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE12345670EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://myproxy-private.xks.example.com",
        "UriPath": "/kms/xks/v1",
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-
example1"
      }
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Melihat penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [DescribeCustomKeyStores](#) di Referensi AWS CLI Perintah.

## describe-key

Contoh kode berikut menunjukkan cara menggunakan `describe-key`.

### AWS CLI

Contoh 1: Untuk menemukan informasi rinci tentang kunci KMS

`describe-key` Contoh berikut mendapatkan informasi rinci tentang kunci AWS terkelola untuk Amazon S3 di akun contoh dan Wilayah. Anda dapat menggunakan perintah ini untuk menemukan detail tentang kunci AWS terkelola dan kunci yang dikelola pelanggan.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan nilai nama alias, tetapi Anda dapat menggunakan ID kunci, ARN kunci, nama alias, atau alias ARN dalam perintah ini.

```
aws kms describe-key \  
  --key-id alias/aws/s3
```

Output:

```
{  
  "KeyMetadata": {  
    "AWSAccountId": "846764612917",  
    "KeyId": "b8a9477d-836c-491f-857e-07937918959b",  
    "Arn": "arn:aws:kms:us-west-2:846764612917:key/  
b8a9477d-836c-491f-857e-07937918959b",  
    "CreationDate": 2017-06-30T21:44:32.140000+00:00,  
    "Enabled": true,  
    "Description": "Default KMS key that protects my S3 objects when no other  
key is defined",  
    "KeyUsage": "ENCRYPT_DECRYPT",  
    "KeyState": "Enabled",  
    "Origin": "AWS_KMS",  
    "KeyManager": "AWS",
```

```
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat AWS kunci](#) di Panduan Pengembang Layanan Manajemen Kunci.

Contoh 2: Untuk mendapatkan detail tentang kunci KMS asimetris RSA

`describe-key` Contoh berikut mendapatkan informasi rinci tentang kunci KMS RSA asimetris yang digunakan untuk penandatanganan dan verifikasi.

```
aws kms describe-key \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": "2019-12-02T19:47:14.861000+00:00",
    "CustomerMasterKeySpec": "RSA_2048",
    "Enabled": false,
    "Description": "",
    "KeyState": "Disabled",
    "Origin": "AWS_KMS",
    "MultiRegion": false,
    "KeyManager": "CUSTOMER",
    "KeySpec": "RSA_2048",
    "KeyUsage": "SIGN_VERIFY",
    "SigningAlgorithms": [
      "RSASSA_PKCS1_V1_5_SHA_256",
      "RSASSA_PKCS1_V1_5_SHA_384",
      "RSASSA_PKCS1_V1_5_SHA_512",
      "RSASSA_PSS_SHA_256",
      "RSASSA_PSS_SHA_384",
      "RSASSA_PSS_SHA_512"
    ]
  }
}
```

```

    ]
  }
}

```

### Contoh 3: Untuk mendapatkan detail tentang kunci replika Multi-wilayah

`describe-key` Contoh berikut mendapatkan metadata untuk kunci replika Multi-wilayah. Kunci Multi-region ini adalah kunci enkripsi simetris. Output dari `describe-key` perintah untuk setiap kunci Multi-region mengembalikan informasi tentang kunci utama dan semua replika nya.

```

aws kms describe-key \
  --key-id arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab

```

### Output:

```

{
  "KeyMetadata": {
    "MultiRegion": true,
    "AWSAccountId": "111122223333",
    "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
    "CreationDate": "2021-06-28T21:09:16.114000+00:00",
    "Description": "",
    "Enabled": true,
    "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "Origin": "AWS_KMS",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "MultiRegionConfiguration": {
      "MultiRegionKeyType": "PRIMARY",
      "PrimaryKey": {
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "Region": "us-west-2"
      },
      "ReplicaKeys": [
        {

```

```

        "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "Region": "eu-west-1"
    },
    {
        "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "Region": "ap-northeast-1"
    },
    {
        "Arn": "arn:aws:kms:sa-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "Region": "sa-east-1"
    }
]
}
}
}

```

Contoh 4: Untuk mendapatkan detail tentang kunci HMAC KMS

`describe-key` Contoh berikut mendapatkan informasi rinci tentang kunci HMAC KMS.

```

aws kms describe-key \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab

```

Output:

```

{
  "KeyMetadata": {
    "AWSAccountId": "123456789012",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Arn": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": "2022-04-03T22:23:10.194000+00:00",
    "Enabled": true,
    "Description": "Test key",
    "KeyUsage": "GENERATE_VERIFY_MAC",
    "KeyState": "Enabled",
    "Origin": "AWS_KMS",
    "KeyManager": "CUSTOMER",
    "CustomerMasterKeySpec": "HMAC_256",
    "MacAlgorithms": [

```

```
        "HMAC_SHA_256"  
    ],  
    "MultiRegion": false  
  }  
}
```

- Untuk detail API, lihat [DescribeKey](#) di Referensi AWS CLI Perintah.

## disable-key-rotation

Contoh kode berikut menunjukkan cara menggunakan `disable-key-rotation`.

### AWS CLI

Untuk menonaktifkan rotasi otomatis tombol KMS

`disable-key-rotation` Contoh berikut menonaktifkan rotasi otomatis kunci KMS yang dikelola pelanggan. Untuk mengaktifkan kembali rotasi otomatis, gunakan `enable-key-rotation` perintah.

```
aws kms disable-key-rotation \  
  --key-id arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output. Untuk memverifikasi bahwa rotasi otomatis dinonaktifkan untuk kunci KMS, gunakan `get-key-rotation-status` perintah.

Untuk informasi selengkapnya, lihat [Memutar kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [DisableKeyRotation](#) di Referensi AWS CLI Perintah.

## disable-key

Contoh kode berikut menunjukkan cara menggunakan `disable-key`.

### AWS CLI

Untuk menonaktifkan sementara kunci KMS

Contoh berikut menggunakan `disable-key` perintah untuk menonaktifkan kunci KMS yang dikelola pelanggan. Untuk mengaktifkan kembali kunci KMS, gunakan perintah. `enable-key`



```
aws kms disable-key \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan dan Menonaktifkan Kunci di Panduan Pengembang Layanan Manajemen AWS Kunci](#).

- Untuk detail API, lihat [DisableKey](#) di Referensi AWS CLI Perintah.

## **disconnect-custom-key-store**

Contoh kode berikut menunjukkan cara menggunakan `disconnect-custom-key-store`.

### AWS CLI

Untuk memutuskan sambungan toko kunci kustom

`disconnect-custom-key-store` Contoh berikut memutus penyimpanan kunci kustom dari kluster AWS CloudHSM. Anda mungkin memutuskan penyimpanan kunci untuk memecahkan masalah, memperbarui pengaturannya, atau untuk mencegah kunci KMS di keystore digunakan dalam operasi kriptografi.

Perintah ini sama untuk semua toko kunci khusus, termasuk toko kunci AWS CloudHSM dan toko kunci eksternal.

Sebelum menjalankan perintah ini, ganti contoh ID penyimpanan kunci kustom dengan yang valid.

```
$ aws kms disconnect-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0
```

Perintah ini tidak menghasilkan keluaran. verifikasi bahwa perintah itu efektif, gunakan perintah `describe-custom-key-stores`

Untuk informasi selengkapnya tentang memutuskan sambungan AWS penyimpanan kunci CloudHSM, [lihat Menyambungkan dan memutuskan penyimpanan kunci AWS CloudHSM di Panduan Pengembang Layanan Manajemen Kunci.AWS](#)

Untuk informasi selengkapnya tentang memutuskan penyimpanan kunci eksternal, lihat [Menghubungkan dan memutuskan penyimpanan kunci eksternal di Panduan Pengembang Layanan Manajemen AWS Kunci](#).

- Untuk detail API, lihat [DisconnectCustomKeyStore](#) di Referensi AWS CLI Perintah.

## enable-key-rotation

Contoh kode berikut menunjukkan cara menggunakan `enable-key-rotation`.

### AWS CLI

Untuk mengaktifkan rotasi otomatis tombol KMS

`enable-key-rotation` Contoh berikut memungkinkan rotasi otomatis kunci KMS yang dikelola pelanggan dengan periode rotasi 180 hari. Kunci KMS akan diputar satu tahun (perkiraan 365 hari) dari tanggal perintah ini selesai dan setiap tahun setelahnya.

`--key-id` Parameter mengidentifikasi kunci KMS. Contoh ini menggunakan nilai ARN kunci, tetapi Anda dapat menggunakan ID kunci atau ARN dari kunci `--rotation-period-in-days` KMS. Parameter menentukan jumlah hari antara setiap tanggal rotasi. Tentukan nilai antara 90 dan 2560 hari. Jika tidak ada nilai yang ditentukan, nilai defaultnya adalah 365 hari.

```
aws kms enable-key-rotation \
  --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \
  --rotation-period-in-days 180
```

Perintah ini tidak menghasilkan output. Untuk memverifikasi bahwa kunci KMS diaktifkan, gunakan `get-key-rotation-status` perintah.

Untuk informasi selengkapnya, lihat [Memutar kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [EnableKeyRotation](#) di Referensi AWS CLI Perintah.

## enable-key

Contoh kode berikut menunjukkan cara menggunakan `enable-key`.

### AWS CLI

Untuk mengaktifkan kunci KMS

`enable-key` Contoh berikut memungkinkan kunci yang dikelola pelanggan. Anda dapat menggunakan perintah seperti ini untuk mengaktifkan kunci KMS yang Anda nonaktifkan

sementara dengan menggunakan `disable-key` perintah. Anda juga dapat menggunakannya untuk mengaktifkan kunci KMS yang dinonaktifkan karena dijadwalkan untuk dihapus dan penghapusan dibatalkan.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan nilai ID kunci, tetapi Anda dapat menggunakan ID kunci atau nilai ARN kunci dalam perintah ini.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan yang valid.

```
aws kms enable-key \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output. Untuk memverifikasi bahwa kunci KMS diaktifkan, gunakan `describe-key` perintah. Lihat nilai `KeyState` dan `Enabled` bidang dalam `describe-key` output.

Untuk informasi selengkapnya, lihat [Mengaktifkan dan Menonaktifkan Kunci di Panduan Pengembang Layanan Manajemen AWS Kunci](#).

- Untuk detail API, lihat [EnableKey](#) di Referensi AWS CLI Perintah.

## encrypt

Contoh kode berikut menunjukkan cara menggunakan `encrypt`.

### AWS CLI

Contoh 1: Untuk mengenkripsi isi file di Linux atau macOS

`encrypt` Perintah berikut menunjukkan cara yang disarankan untuk mengenkripsi data dengan CLI AWS .

```
aws kms encrypt \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --plaintext fileb://ExamplePlaintextFile \  
  --output text \  
  --query CiphertextBlob | base64 \  
  --decode > ExampleEncryptedFile
```

Perintah tersebut melakukan beberapa hal:

Menggunakan `--plaintext` parameter untuk menunjukkan data untuk mengenkripsi. Nilai parameter ini harus dikodekan Base64. Nilai `plaintext` parameter harus dikodekan base64, atau Anda harus menggunakan awalan `fileb://`, yang memberitahu AWS CLI untuk membaca data biner dari file. Jika file tidak ada di direktori saat ini, ketik path lengkap ke file. Misalnya: `fileb:///var/tmp/ExamplePlaintextFile` atau `fileb://C:\Temp\ExamplePlaintextFile`. [Untuk informasi lebih lanjut tentang membaca nilai parameter AWS CLI dari file, lihat Memuat Parameter dari File di Panduan Pengguna Antarmuka Baris AWS Perintah dan Praktik Terbaik untuk Parameter File Lokal di Blog Alat Baris AWS Perintah.](#) Menggunakan `--output` dan `--query` parameter untuk mengontrol output perintah. Parameter ini mengekstrak data terenkripsi, yang disebut `ciphertext`, dari output perintah. [Untuk informasi lebih lanjut tentang mengontrol output, lihat Mengontrol Perintah Output](#) dalam Panduan Pengguna Antarmuka Baris AWS Perintah. Menggunakan `base64` utilitas untuk memecahkan kode output yang diekstraksi ke dalam data biner. `Ciphertext` yang dikembalikan oleh perintah yang berhasil adalah teks yang dikodekan base64. `encrypt` Anda harus memecahkan kode teks ini sebelum Anda dapat menggunakan AWS CLI untuk mendekripsi itu. Menyimpan `ciphertext` biner ke file. Bagian terakhir dari perintah `> ExampleEncryptedFile` () menyimpan `ciphertext` biner ke file untuk membuat dekripsi lebih mudah. Untuk contoh perintah yang menggunakan AWS CLI untuk mendekripsi data, lihat contoh dekripsi.

Contoh 2: Menggunakan AWS CLI untuk mengenkripsi data pada Windows

Contoh ini sama dengan yang sebelumnya, kecuali bahwa ia menggunakan `certutil` alat sebagai gantinya `base64`. Prosedur ini membutuhkan dua perintah, seperti yang ditunjukkan pada contoh berikut.

```
aws kms encrypt \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --plaintext fileb:///ExamplePlaintextFile \
  --output text \
  --query CiphertextBlob > C:\Temp\ExampleEncryptedFile.base64

certutil -decode C:\Temp\ExampleEncryptedFile.base64 C:\Temp\ExampleEncryptedFile
```

Contoh 3: Mengenkripsi dengan tombol KMS asimetris

`encrypt` Perintah berikut menunjukkan cara mengenkripsi `plaintext` dengan kunci KMS asimetris. parameter `--encryption-algorithm` diperlukan. Seperti pada semua perintah `encrypt` CLI, `plaintext` parameter harus dikodekan base64, atau Anda harus menggunakan `fileb://` awalan, yang memberi tahu AWS CLI untuk membaca data biner dari file.

```
aws kms encrypt \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --encryption-algorithm RSAES_OAEP_SHA_256 \  
  --plaintext file://ExamplePlaintextFile \  
  --output text \  
  --query CiphertextBlob | base64 \  
  --decode > ExampleEncryptedFile
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [Enkripsi](#) di Referensi AWS CLI Perintah.

## generate-data-key-pair-without-plaintext

Contoh kode berikut menunjukkan cara menggunakan `generate-data-key-pair-without-plaintext`.

### AWS CLI

Untuk menghasilkan ECC NIST P384 asymmetric data key pair

`generate-data-key-pair-without-plaintext` Contoh berikut meminta key pair ECC NIST P384 untuk digunakan di luar. AWS

Perintah mengembalikan kunci publik plaintext dan salinan kunci pribadi yang dienkripsi di bawah kunci KMS yang ditentukan. Itu tidak mengembalikan kunci pribadi plaintext. Anda dapat dengan aman menyimpan kunci pribadi terenkripsi dengan data terenkripsi, dan memanggil AWS KMS untuk mendekripsi kunci pribadi saat Anda perlu menggunakannya.

Untuk meminta ECC NIST P384 asymmetric data key pair, gunakan parameter `key-pair-spec` dengan nilai. `ECC_NIST_P384`

Kunci KMS yang Anda tentukan harus berupa kunci KMS enkripsi simetris, yaitu kunci KMS dengan nilai. `KeySpec SYMMETRIC_DEFAULT`

**CATATAN:** Nilai-nilai dalam output dari contoh ini terpotong untuk tampilan.

```
aws kms generate-data-key-pair-without-plaintext \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --key-pair-spec ECC_NIST_P384
```

**Output:**

```
{
  "PrivateKeyCiphertextBlob": "AQIDAHi6LtupRpdK12aJTzkK6Fbh0tQkM1QJJH3PdtHvS/y
+hAFFxmiD134doUDzMGmfCEtcAAAHaTCCB2UGCSqGSIb3DQEHBqCCB1...",
  "PublicKey":
  "MIIBojANBgkqhkiG9w0BAQEFAAOCAy8AMIIBigKCAYEA3A3eGMyPrivSn7+Ld1JE1oUoQV5HpEuHAVbd0yND
+NmYDH/mL10SIEuLrcdZ5hrMH4pk83r401...",
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "KeyPairSpec": "ECC_NIST_P384"
}
```

The `PublicKey` dan dikembalikan dalam `PrivateKeyCiphertextBlob` format yang dikodekan base64.

Untuk informasi selengkapnya, lihat [Pasangan kunci data](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GenerateDataKeyPairWithoutPlaintext](#) di Referensi AWS CLI Perintah.

**generate-data-key-pair**

Contoh kode berikut menunjukkan cara menggunakan `generate-data-key-pair`.

**AWS CLI**

Untuk menghasilkan sebuah 2048-bit RSA asymmetric data key pair

`generate-data-key-pair` Contoh berikut meminta 2048-bit RSA asymmetric data key pair untuk digunakan di luar. AWS Perintah mengembalikan kunci publik plaintext dan kunci pribadi plaintext untuk segera digunakan dan dihapus, dan salinan kunci pribadi yang dienkripsi di bawah kunci KMS yang ditentukan. Anda dapat dengan aman menyimpan kunci pribadi terenkripsi dengan data terenkripsi.

Untuk meminta 2048-bit RSA asymmetric data key pair, gunakan parameter `key-pair-spec` dengan nilai. `RSA_2048`

Kunci KMS yang Anda tentukan harus berupa kunci KMS enkripsi simetris, yaitu kunci KMS dengan nilai. `KeySpec SYMMETRIC_DEFAULT`

**CATATAN:** Nilai-nilai dalam output dari contoh ini terpotong untuk tampilan.

```
aws kms generate-data-key-pair \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --key-pair-spec RSA_2048
```

### Output:

```
{
  "PrivateKeyCiphertextBlob": "AQIDAHi6LtupRpdK12aJTzkk6Fbh0tQkMlQJJH3PdtHvS/y
+hAFFxmiD134doUDzMGmfCEtcAAAHaTCCB2UGCSqGSIB3DQEHBqCCB1...",
  "PrivateKeyPlaintext": "MIIG/
QIBADANBgqhkiG9w0BAQEFAASCBUcwggbjAgEAAoIBgQDcDd4YzI
+u9Kfv4t2UkTWhShBXkekS4cBVt07I0P42ZgMf+YvU5IgS4ut...",
  "PublicKey":
  "MIIB0jANBgqhkiG9w0BAQEFAA0CAY8AMIIBigKCAYEA3A3eGMyPrvSn7+Ld1JE1oUoQV5HpEuHAVbd0yND
+NmYDH/mL10SIEuLrzdZ5hrMH4pk83r401...",
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "KeyPairSpec": "RSA_2048"
}
```

The `PublicKey`, `PrivateKeyPlaintext`, dan dikembalikan dalam `PrivateKeyCiphertextBlob` format yang dikodekan base64.

Untuk informasi selengkapnya, lihat [Pasangan kunci data](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GenerateDataKeyPair](#) di Referensi AWS CLI Perintah.

## generate-data-key-without-plaintext

Contoh kode berikut menunjukkan cara menggunakan `generate-data-key-without-plaintext`.

### AWS CLI

Untuk menghasilkan kunci data simetris 256-bit tanpa kunci plaintext

`generate-data-key-without-plaintext` Contoh berikut meminta salinan terenkripsi dari kunci data simetris 256-bit untuk digunakan di luar. AWS Anda dapat memanggil AWS KMS untuk mendekripsi kunci data saat Anda siap menggunakannya.

Untuk meminta kunci data 256-bit, gunakan `key-spec` parameter dengan nilai `AES_256`. Untuk meminta kunci data 128-bit, gunakan `key-spec` parameter dengan nilai `AES_128`. Untuk semua panjang kunci data lainnya, gunakan `number-of-bytes` parameter.

Kunci KMS yang Anda tentukan harus berupa kunci KMS enkripsi simetris, yaitu kunci KMS dengan nilai spesifikasi kunci `SYMMETRIC_DEFAULT`.

```
aws kms generate-data-key-without-plaintext \
  --key-id "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab" \
  --key-spec AES_256
```

Output:

```
{
  "CiphertextBlob":
  "AQEDAHjRYf5WytIc0C857tFSnBaPn2F8DgfmThbJlGfR8P3WlwAAAH4wfAYJKoZlIhvcNAQcGoG8wbQIBADBoBgkqhki
  "KeyId": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

`CiphertextBlob` (Kunci data terenkripsi) dikembalikan dalam format yang dikodekan base64.

Untuk informasi selengkapnya, lihat [Kunci data](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GenerateDataKeyWithoutPlaintext](#) di Referensi AWS CLI Perintah.

## generate-data-key

Contoh kode berikut menunjukkan cara menggunakan `generate-data-key`.

### AWS CLI

Contoh 1: Untuk menghasilkan kunci data simetris 256-bit

`generate-data-key` Contoh berikut meminta kunci data simetris 256-bit untuk digunakan di luar. AWS Perintah mengembalikan kunci data plaintext untuk segera digunakan dan dihapus, dan salinan kunci data yang dienkripsi di bawah kunci KMS yang ditentukan. Anda dapat dengan aman menyimpan kunci data terenkripsi dengan data terenkripsi.



Untuk meminta kunci data 256-bit, gunakan `key-spec` parameter dengan nilai. `AES_256` Untuk meminta kunci data 128-bit, gunakan `key-spec` parameter dengan nilai. `AES_128` Untuk semua panjang kunci data lainnya, gunakan `number-of-bytes` parameter.

Kunci KMS yang Anda tentukan harus berupa kunci KMS enkripsi simetris, yaitu kunci KMS dengan nilai spesifikasi kunci `SYMMETRIC_DEFAULT`.

```
aws kms generate-data-key \
  --key-id alias/ExampleAlias \
  --key-spec AES_256
```

Output:

```
{
  "Plaintext": "VdzKNHGzUAzJeRBVY+uUmofUGGiDzyB3+i9fVkh3piw=",
  "KeyId": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "CiphertextBlob":
  "AQEDAHAjRYf5WytIc0C857tFSnBaPn2F8DgfmThbJlGfR8P3WlwAAAH4wfAYJKoZIhvcNAQcGoG8wbQIBADBoBgkqhkg
+YdhV8MirkBQPeac0ReRVNDt9qlEAt+SHgIRF8P0H+7U="
}
```

`Plaintext` (Kunci data teks biasa) dan `CiphertextBlob` (kunci data terenripsi) dikembalikan dalam format yang disandikan base64.

Untuk informasi selengkapnya, lihat Kunci data < <https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys> di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 2: Untuk menghasilkan kunci data simetris 512-bit

`generate-data-key` Contoh berikut meminta kunci data simetris 512-bit untuk enkripsi dan dekripsi. Perintah mengembalikan kunci data plaintext untuk segera digunakan dan dihapus, dan salinan kunci data yang dienkripsi di bawah kunci KMS yang ditentukan. Anda dapat dengan aman menyimpan kunci data terenripsi dengan data terenripsi.

Untuk meminta panjang kunci selain 128 atau 256 bit, gunakan `number-of-bytes` parameter. Untuk meminta kunci data 512-bit, contoh berikut menggunakan `number-of-bytes` parameter dengan nilai 64 (byte).

Kunci KMS yang Anda tentukan harus berupa kunci KMS enkripsi simetris, yaitu kunci KMS dengan nilai spesifikasi kunci `SYMMETRIC_DEFAULT`.

CATATAN: Nilai-nilai dalam output dari contoh ini terpotong untuk tampilan.

```
aws kms generate-data-key \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --number-of-bytes 64
```

Output:

```
{  
  "CiphertextBlob": "AQIBAHi6LtupRpdK12aJTzkk6Fbh0tQkMlQJJH3PdtHvS/y+hAEnX/  
QQNmMwDfg2koιNMEc8AAACaDCCAmQGCSqGSiB3DQEBqCCA1UwggJRAgEAMIICSgYJKoZ...",  
  "Plaintext": "ty8Lr0Bk60F07M2Bwt6qbFdNB  
+G00ZLtf5MSEb4a13R2UKWG0p06njAwy2n72VRm2m7z/  
Pm9Wpbvttz6a4lSo9hgPvKhZ5y6RTm40ovEXiVfBveyX3DQxDzRSwbKDPk/...",  
  "KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

Plaintext (Kunci data teks biasa) dan CiphertextBlob (kunci data terenkripsi) dikembalikan dalam format yang disandikan base64.

Untuk informasi selengkapnya, lihat Kunci data < <https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys> di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GenerateDataKey](#) di Referensi AWS CLI Perintah.

## generate-random

Contoh kode berikut menunjukkan cara menggunakan `generate-random`.

### AWS CLI

Contoh 1: Untuk menghasilkan string byte acak 256-bit (Linux atau macOS)

`generate-random` Contoh berikut menghasilkan 256-bit (32-byte), string byte acak yang dikodekan base64. Contoh menerjemahkan string byte dan menyimpannya dalam file acak.

Ketika Anda menjalankan perintah ini, Anda harus menggunakan `number-of-bytes` parameter untuk menentukan panjang nilai acak dalam byte.

Anda tidak menentukan kunci KMS ketika Anda menjalankan perintah ini. String byte acak tidak terkait dengan kunci KMS apa pun.

Secara default, AWS KMS menghasilkan nomor acak. Namun, jika Anda menentukan penyimpanan kunci kustom < <https://docs.aws.amazon.com/kms/latest/developerguide/custom-key-store-overview.html> >, string byte acak dihasilkan di klaster CloudHSM AWS yang terkait dengan penyimpanan kunci khusus.

Contoh ini menggunakan parameter dan nilai berikut:

Ini menggunakan `--number-of-bytes` parameter yang diperlukan dengan nilai 32 untuk meminta string 32-byte (256-bit). Ini menggunakan `--output` parameter dengan nilai `text` mengarahkan AWS CLI untuk mengembalikan output sebagai teks, bukan JSON. Ini menggunakan `Plaintext` untuk mengekstrak nilai `Plaintext` properti dari response. `|` output dari perintah `--query` parameter ke `base64` utilitas, yang menerjemahkan output yang diekstraksi. Ini menggunakan operator pengalihan (`>`) untuk menyimpan string byte yang diterjemahkan ke file. Ini menggunakan operator pengalihan (`>`) `ExampleRandom` untuk menyimpan ciphertext biner ke file.

```
aws kms generate-random \  
  --number-of-bytes 32 \  
  --output text \  
  --query Plaintext | base64 --decode > ExampleRandom
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [GenerateRandom](#) di Referensi API Layanan Manajemen AWS Kunci.

Contoh 2: Untuk menghasilkan nomor acak 256-bit (Windows Command Prompt)

Contoh berikut menggunakan `generate-random` perintah untuk menghasilkan string byte acak 256-bit (32-byte) yang dikodekan base64. Contoh menerjemahkan string byte dan menyimpannya dalam file acak. Contoh ini sama dengan contoh sebelumnya, kecuali bahwa ia menggunakan `certutil` utilitas di Windows untuk base64-decode string byte acak sebelum menyimpannya dalam file.

Pertama, buat string byte acak yang dikodekan base64 dan simpan dalam file sementara,.

`ExampleRandom.base64`

```
aws kms generate-random \  
  --number-of-bytes 32 \  
  --output text \  
  --query Plaintext > ExampleRandom.base64
```

Karena output dari `generate-random` perintah disimpan dalam file, contoh ini tidak menghasilkan output.

Sekarang gunakan `certutil -decode` perintah untuk memecahkan kode string byte yang dikodekan base64 dalam file. `ExampleRandom.base64` Kemudian, ia menyimpan string byte yang diterjemahkan dalam file. `ExampleRandom`

```
certutil -decode ExampleRandom.base64 ExampleRandom
```

Output:

```
Input Length = 18  
Output Length = 12  
CertUtil: -decode command completed successfully.
```

Untuk informasi selengkapnya, lihat [GenerateRandom](#) di Referensi API Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GenerateRandom](#) di Referensi AWS CLI Perintah.

## get-key-policy

Contoh kode berikut menunjukkan cara menggunakan `get-key-policy`.

### AWS CLI

Untuk menyalin kebijakan kunci dari satu kunci KMS ke kunci KMS lainnya

`get-key-policy` Contoh berikut mendapatkan kebijakan kunci dari satu kunci KMS dan menyimpannya dalam file teks. Kemudian, itu menggantikan kebijakan kunci KMS yang berbeda menggunakan file teks sebagai input kebijakan.

Karena `--policy` parameter `put-key-policy` membutuhkan string, Anda harus menggunakan `--output text` opsi untuk mengembalikan output sebagai string teks bukan JSON.

```
aws kms get-key-policy \  
  --policy-name default \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --query Policy \  
  --output text > policy.txt  
  
aws kms put-key-policy \  
  --policy-name default \  
  --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \  
  --policy file://policy.txt
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [PutKeyPolicy](#) di Referensi API AWS KMS.

- Untuk detail API, lihat [GetKeyPolicy](#) di Referensi AWS CLI Perintah.

## get-key-rotation-status

Contoh kode berikut menunjukkan cara menggunakan `get-key-rotation-status`.

### AWS CLI

Untuk mengambil status rotasi untuk kunci KMS.

`get-key-rotation-status` Contoh berikut mengembalikan informasi tentang status rotasi kunci KMS yang ditentukan, termasuk apakah rotasi otomatis diaktifkan, periode rotasi, dan tanggal rotasi terjadwal berikutnya. Anda dapat menggunakan perintah ini pada kunci KMS yang dikelola pelanggan dan kunci KMS yang AWS dikelola. Namun, semua kunci KMS yang AWS dikelola secara otomatis diputar setiap tahun.

```
aws kms get-key-rotation-status \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{  
  "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
```

```
"KeyRotationEnabled": true,  
"NextRotationDate": "2024-02-14T18:14:33.587000+00:00",  
"RotationPeriodInDays": 365  
}
```

Untuk informasi selengkapnya, lihat [Memutar kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GetKeyRotationStatus](#) di Referensi AWS CLI Perintah.

## get-parameters-for-import

Contoh kode berikut menunjukkan cara menggunakan `get-parameters-for-import`.

### AWS CLI

Untuk mendapatkan item yang diperlukan untuk mengimpor materi kunci ke kunci KMS

`get-parameters-for-import` Contoh berikut mendapatkan kunci publik dan token impor yang Anda butuhkan untuk mengimpor materi kunci ke kunci KMS. Saat Anda menggunakan `import-key-material` perintah, pastikan untuk menggunakan token impor dan materi kunci yang dienkripsi oleh kunci publik yang dikembalikan dalam perintah yang sama `get-parameters-for-import`. Juga, algoritma pembungkus yang Anda tentukan dalam perintah ini harus menjadi salah satu yang Anda gunakan untuk mengenkripsi materi kunci dengan kunci publik.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan ID kunci, tetapi Anda dapat menggunakan ID kunci atau kunci ARN dalam perintah ini.

```
aws kms get-parameters-for-import \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --wrapping-algorithm RSAES_OAEP_SHA_256 \  
  --wrapping-key-spec RSA_2048
```

Output:

```
{  
  "KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
  "PublicKey": "<public key base64 encoded data>",  
  "ImportToken": "<import token base64 encoded data>",  
  "ParametersValidTo": 1593893322.32
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengunduh kunci publik dan token impor](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GetParametersForImport](#) di Referensi AWS CLI Perintah.

## get-public-key

Contoh kode berikut menunjukkan cara menggunakan `get-public-key`.

### AWS CLI

Contoh 1: Untuk mengunduh kunci publik dari kunci KMS asimetris

`get-public-key` Contoh berikut mengunduh kunci publik dari kunci KMS asimetris.

Selain mengembalikan kunci publik, output mencakup informasi yang Anda butuhkan untuk menggunakan kunci publik dengan aman di luar AWS KMS, termasuk penggunaan kunci dan algoritma enkripsi yang didukung.

```
aws kms get-public-key \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "PublicKey": "jANBgqhkkiG9w0BAQEFAAOCAg8AMIICCgKCAgEA15epvg1/
QtJhxSi2g9SDEVg8QV/...",
  "CustomerMasterKeySpec": "RSA_4096",
  "KeyUsage": "ENCRYPT_DECRYPT",
  "EncryptionAlgorithms": [
    "RSAES_OAEP_SHA_1",
    "RSAES_OAEP_SHA_256"
  ]
}
```

Untuk informasi selengkapnya tentang penggunaan kunci KMS asimetris di AWS KMS, lihat [Menggunakan Kunci Simetris dan Asimetris di Referensi API](#) Layanan Manajemen AWS Kunci.

Contoh 2: Untuk mengonversi kunci publik ke format DER (Linux dan macOS)

`get-public-key` Contoh berikut mengunduh kunci publik dari kunci KMS asimetris dan menyimpannya dalam file DER.

Saat Anda menggunakan `get-public-key` perintah di AWS CLI, ia mengembalikan kunci publik X.509 yang dikodekan DER yang dikodekan oleh Base64. Contoh ini mendapatkan nilai `PublicKey` properti sebagai teks. Ini Base64-mendekode `PublicKey` dan menyimpannya dalam file `public_key.der` outputParameter mengembalikan output sebagai teks, bukan JSON. `--queryParameter` hanya mendapatkan `PublicKey` properti, bukan properti yang Anda butuhkan untuk menggunakan kunci publik dengan aman di luar AWS KMS.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms get-public-key \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --output text \  
  --query PublicKey | base64 --decode > public_key.der
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang penggunaan kunci KMS asimetris di AWS KMS, lihat [Menggunakan Kunci Simetris dan Asimetris di Referensi API](#) Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [GetPublicKey](#) di Referensi AWS CLI Perintah.

## **import-key-material**

Contoh kode berikut menunjukkan cara menggunakan `import-key-material`.

### AWS CLI

Untuk mengimpor materi kunci ke kunci KMS

`import-key-material` Contoh berikut mengunggah materi kunci ke kunci KMS yang dibuat tanpa materi kunci. Status kunci dari kunci KMS harus `PendingImport`.

Perintah ini menggunakan materi kunci yang Anda enkripsi dengan kunci publik yang dikembalikan `get-parameters-for-import` perintah. Ini juga menggunakan token impor dari `get-parameters-for-import` perintah yang sama.



`expiration-model`Parameter menunjukkan bahwa bahan utama secara otomatis kedaluwarsa pada tanggal dan waktu yang ditentukan oleh `valid-to` parameter. Ketika materi kunci kedaluwarsa, AWS KMS menghapus materi kunci, status kunci kunci KMS berubah menjadi `Pending import` dan kunci KMS menjadi tidak dapat digunakan. Untuk mengembalikan kunci KMS, Anda harus mengimpor ulang materi kunci yang sama. Untuk menggunakan bahan kunci yang berbeda, Anda harus membuat kunci KMS baru.

Sebelum menjalankan perintah ini, ganti contoh ID kunci dengan ID kunci yang valid atau ARN kunci dari akun Anda AWS .

```
aws kms import-key-material \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --encrypted-key-material fileb://EncryptedKeyMaterial.bin \
  --import-token fileb://ImportToken.bin \
  --expiration-model KEY_MATERIAL_EXPIRES \
  --valid-to 2021-09-21T19:00:00Z
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang mengimpor materi utama, lihat [Mengimpor Materi Utama](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ImportKeyMaterial](#) di Referensi AWS CLI Perintah.

## list-aliases

Contoh kode berikut menunjukkan cara menggunakan `list-aliases`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua alias di AWS akun dan Wilayah

Contoh berikut menggunakan `list-aliases` perintah untuk daftar semua alias di Wilayah default AWS akun. Outputnya mencakup alias yang terkait dengan kunci KMS AWS terkelola dan kunci KMS yang dikelola pelanggan.

```
aws kms list-aliases
```

Output:

```
{
  "Aliases": [
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/testKey",
      "AliasName": "alias/testKey",
      "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/FinanceDept",
      "AliasName": "alias/FinanceDept",
      "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
    },
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/dynamodb",
      "AliasName": "alias/aws/dynamodb",
      "TargetKeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    },
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/ebs",
      "AliasName": "alias/aws/ebs",
      "TargetKeyId": "0987ab65-43cd-21ef-09ab-87654321cdef"
    },
    ...
  ]
}
```

Contoh 2: Untuk daftar semua alias untuk kunci KMS tertentu

Contoh berikut menggunakan `list-aliases` perintah dan `key-id` parameternya untuk daftar semua alias yang terkait dengan kunci KMS tertentu.

Setiap alias dikaitkan dengan hanya satu kunci KMS, tetapi kunci KMS dapat memiliki beberapa alias. Perintah ini sangat berguna karena konsol AWS KMS hanya mencantumkan satu alias untuk setiap tombol KMS. Untuk menemukan semua alias untuk kunci KMS, Anda harus menggunakan perintah. `list-aliases`

Contoh ini menggunakan ID kunci kunci KMS untuk `--key-id` parameter, tetapi Anda dapat menggunakan ID kunci, kunci ARN, nama alias, atau alias ARN dalam perintah ini.

```
aws kms list-aliases --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
  "Aliases": [
    {
      "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/oregon-test-key",
      "AliasName": "alias/oregon-test-key"
    },
    {
      "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/project121-test",
      "AliasName": "alias/project121-test"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Alias](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ListAliases](#) di Referensi AWS CLI Perintah.

## list-grants

Contoh kode berikut menunjukkan cara menggunakan `list-grants`.

### AWS CLI

Untuk melihat hibah pada kunci AWS KMS

`list-grants` Contoh berikut menampilkan semua hibah pada kunci KMS AWS terkelola yang ditentukan untuk Amazon DynamoDB di akun Anda. Hibah ini memungkinkan DynamoDB untuk menggunakan kunci KMS atas nama Anda untuk mengenkripsi tabel DynamoDB sebelum menulisnya ke disk. Anda dapat menggunakan perintah seperti ini untuk melihat hibah pada kunci KMS yang AWS dikelola dan kunci KMS yang dikelola pelanggan di AWS akun dan Wilayah.

Perintah ini menggunakan `key-id` parameter dengan ID kunci untuk mengidentifikasi kunci KMS. Anda dapat menggunakan ID kunci atau kunci ARN untuk mengidentifikasi kunci KMS. Untuk mendapatkan ID kunci atau kunci ARN dari kunci KMS yang AWS dikelola, gunakan perintah `or`.

`list-keys list-aliases`

```
aws kms list-grants \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output menunjukkan bahwa hibah memberikan izin Amazon DynamoDB untuk menggunakan kunci KMS untuk operasi kriptografi, dan memberikan izin untuk melihat detail tentang kunci KMS () dan untuk menghentikan hibah DescribeKey (). RetireGrant Batasan EncryptionContextSubset membatasi izin ini untuk permintaan yang menyertakan pasangan konteks enkripsi yang ditentukan. Akibatnya, izin dalam hibah hanya efektif pada akun tertentu dan tabel DynamoDB.

```
{
  "Grants": [
    {
      "Constraints": {
        "EncryptionContextSubset": {
          "aws:dynamodb:subscriberId": "123456789012",
          "aws:dynamodb:tableName": "Services"
        }
      },
      "IssuingAccount": "arn:aws:iam::123456789012:root",
      "Name": "8276b9a6-6cf0-46f1-b2f0-7993a7f8c89a",
      "Operations": [
        "Decrypt",
        "Encrypt",
        "GenerateDataKey",
        "ReEncryptFrom",
        "ReEncryptTo",
        "RetireGrant",
        "DescribeKey"
      ],
      "GrantId":
        "1667b97d27cf748cf05b487217dd4179526c949d14fb3903858e25193253fe59",
      "KeyId": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "RetiringPrincipal": "dynamodb.us-west-2.amazonaws.com",
      "GranteePrincipal": "dynamodb.us-west-2.amazonaws.com",
      "CreationDate": "2021-05-13T18:32:45.144000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Hibah di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Utama.

- Untuk detail API, lihat [ListGrants](#) di Referensi AWS CLI Perintah.

## list-key-policies

Contoh kode berikut menunjukkan cara menggunakan `list-key-policies`.

### AWS CLI

Untuk mendapatkan nama kebijakan kunci untuk kunci KMS

`list-key-policies` Contoh berikut mendapatkan nama-nama kebijakan utama untuk kunci yang dikelola pelanggan di akun contoh dan Wilayah. Anda dapat menggunakan perintah ini untuk menemukan nama kebijakan kunci untuk kunci AWS terkelola dan kunci yang dikelola pelanggan.

Karena satu-satunya nama kebijakan kunci yang valid adalah `default`, perintah ini tidak berguna.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan nilai ID kunci, tetapi Anda dapat menggunakan ID kunci atau kunci ARN dalam perintah ini.

```
aws kms list-key-policies \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
  "PolicyNames": [
    "default"
  ]
}
```

Untuk informasi selengkapnya tentang kebijakan kunci AWS KMS, lihat [Menggunakan Kebijakan Utama di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ListKeyPolicies](#) di Referensi AWS CLI Perintah.

## list-key-rotations

Contoh kode berikut menunjukkan cara menggunakan `list-key-rotations`.

### AWS CLI

Untuk mengambil informasi tentang semua rotasi material kunci yang telah selesai

`list-key-rotations` Contoh berikut mencantumkan informasi tentang semua rotasi material kunci yang diselesaikan untuk kunci KMS yang ditentukan.

```
aws kms list-key-rotations \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
  "Rotations": [
    {
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "RotationDate": "2024-03-02T10:11:36.564000+00:00",
      "RotationType": "AUTOMATIC"
    },
    {
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "RotationDate": "2024-04-05T15:14:47.757000+00:00",
      "RotationType": "ON_DEMAND"
    }
  ],
  "Truncated": false
}
```

Untuk informasi selengkapnya, lihat [Memutar kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ListKeyRotations](#) di Referensi AWS CLI Perintah.

## list-keys

Contoh kode berikut menunjukkan cara menggunakan `list-keys`.

### AWS CLI

Untuk mendapatkan kunci KMS di akun dan Wilayah

`list-keys` Contoh berikut mendapatkan kunci KMS di akun dan Region. Perintah ini mengembalikan kunci AWS terkelola dan kunci yang dikelola pelanggan.

```
aws kms list-keys
```

**Output:**

```
{
  "Keys": [
    {
      "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321",
      "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
    },
    {
      "KeyArn": "arn:aws:kms:us-
east-2:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
      "KeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat AWS Kunci](#) di Panduan Pengembang Layanan Manajemen Kunci.

- Untuk detail API, lihat [ListKeys](#) di Referensi AWS CLI Perintah.

**list-resource-tags**

Contoh kode berikut menunjukkan cara menggunakan `list-resource-tags`.

**AWS CLI**

Untuk mendapatkan tag pada kunci KMS

`list-resource-tags` Contoh berikut mendapatkan tag untuk kunci KMS. Untuk menambah atau mengganti tag sumber daya pada tombol KMS, gunakan `tag-resource` perintah. Output menunjukkan bahwa kunci KMS ini memiliki dua tag sumber daya, yang masing-masing memiliki kunci dan nilai.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan nilai ID kunci, tetapi Anda dapat menggunakan ID kunci atau kunci ARN dalam perintah ini.

```
aws kms list-resource-tags \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
  "Tags": [
    {
      "TagKey": "Dept",
      "TagValue": "IT"
    },
    {
      "TagKey": "Purpose",
      "TagValue": "Test"
    }
  ],
  "Truncated": false
}
```

Untuk informasi selengkapnya tentang penggunaan tag di AWS KMS, lihat [Menandai kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ListResourceTags](#) di Referensi AWS CLI Perintah.

## **list-retirable-grants**

Contoh kode berikut menunjukkan cara menggunakan `list-retirable-grants`.

### AWS CLI

Untuk melihat hibah bahwa kepala sekolah dapat pensiun

`list-retirable-grants` Contoh berikut menampilkan semua hibah yang ExampleAdmin pengguna dapat pensiun pada kunci KMS di AWS akun dan Wilayah. Anda dapat menggunakan perintah seperti ini untuk melihat hibah bahwa setiap prinsipal akun dapat pensiun pada kunci KMS di AWS akun dan Wilayah.

Nilai `retiring-principal` parameter yang diperlukan harus berupa Nama Sumber Daya Amazon (ARN) dari akun, pengguna, atau peran.



Anda tidak dapat menentukan layanan untuk nilai `retiring-principal` dalam perintah ini, meskipun layanan dapat menjadi prinsipal pensiun. Untuk menemukan hibah di mana layanan tertentu adalah kepala sekolah pensiun, gunakan perintah `list-grants`

Output menunjukkan bahwa `ExampleAdmin` pengguna memiliki izin untuk menghentikan hibah pada dua kunci KMS yang berbeda di akun dan wilayah. Selain kepala sekolah yang pensiun, akun memiliki izin untuk mempensiunkan hibah apa pun di akun.

```
aws kms list-retirable-grants \
  --retiring-principal arn:aws:iam::111122223333:user/ExampleAdmin
```

Output:

```
{
  "Grants": [
    {
      "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "GrantId":
"156b69c63cb154aa21f59929ff19760717be8d9d82b99df53e18b94a15a5e88e",
      "Name": "",
      "CreationDate": 2021-01-14T20:17:36.419000+00:00,
      "GranteePrincipal": "arn:aws:iam::111122223333:user/ExampleUser",
      "RetiringPrincipal": "arn:aws:iam::111122223333:user/ExampleAdmin",
      "IssuingAccount": "arn:aws:iam::111122223333:root",
      "Operations": [
        "Encrypt"
      ],
      "Constraints": {
        "EncryptionContextSubset": {
          "Department": "IT"
        }
      }
    },
    {
      "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321",
      "GrantId":
"8c94d1f12f5e69f440bae30eaec9570bb1fb7358824f9ddfa1aa5a0dab1a59b2",
      "Name": "",
      "CreationDate": "2021-02-02T19:49:49.638000+00:00",
      "GranteePrincipal": "arn:aws:iam::111122223333:role/ExampleRole",
```

```

    "RetiringPrincipal": "arn:aws:iam::111122223333:user/ExampleAdmin",
    "IssuingAccount": "arn:aws:iam::111122223333:root",
    "Operations": [
      "Decrypt"
    ],
    "Constraints": {
      "EncryptionContextSubset": {
        "Department": "IT"
      }
    }
  },
  "Truncated": false
}

```

Untuk informasi selengkapnya, lihat [Hibah di AWS KMS](#) di Panduan Pengembang Layanan Manajemen AWS Utama.

- Untuk detail API, lihat [ListRetirableGrants](#) di Referensi AWS CLI Perintah.

## put-key-policy

Contoh kode berikut menunjukkan cara menggunakan `put-key-policy`.

### AWS CLI

Untuk mengubah kebijakan kunci untuk kunci KMS

`put-key-policy` Contoh berikut mengubah kebijakan kunci untuk kunci yang dikelola pelanggan.

Untuk memulai, buat kebijakan kunci dan simpan dalam file JSON lokal. Dalam contoh ini, file tersebut adalah `key_policy.json`. Anda juga dapat menentukan kebijakan kunci sebagai nilai string `policy` parameter.

Pernyataan pertama dalam kebijakan kunci ini memberikan izin AWS akun untuk menggunakan kebijakan IAM untuk mengontrol akses ke kunci KMS. Pernyataan kedua memberikan izin `test-user` pengguna untuk menjalankan `list-keys` perintah `describe-key` dan pada kunci KMS.

Isi dari `key_policy.json`:

```

{
  "Version" : "2012-10-17",

```

```

    "Id" : "key-default-1",
    "Statement" : [
      {
        "Sid" : "Enable IAM User Permissions",
        "Effect" : "Allow",
        "Principal" : {
          "AWS" : "arn:aws:iam::111122223333:root"
        },
        "Action" : "kms:*",
        "Resource" : "*"
      },
      {
        "Sid" : "Allow Use of Key",
        "Effect" : "Allow",
        "Principal" : {
          "AWS" : "arn:aws:iam::111122223333:user/test-user"
        },
        "Action" : [
          "kms:DescribeKey",
          "kms:ListKeys"
        ],
        "Resource" : "*"
      }
    ]
  }
}

```

Untuk mengidentifikasi kunci KMS, contoh ini menggunakan ID kunci, tetapi Anda juga dapat menggunakan kunci ARN. Untuk menentukan kebijakan kunci, perintah menggunakan `policy` parameter. Untuk menunjukkan bahwa kebijakan tersebut ada dalam file, kebijakan tersebut menggunakan `file://` awalan yang diperlukan. Awalan ini diperlukan untuk mengidentifikasi file pada semua sistem operasi yang didukung. Akhirnya, perintah menggunakan `policy-name` parameter dengan nilai `default`. Jika tidak ada nama kebijakan yang ditentukan, nilai defaultnya adalah `default`. Satu-satunya nilai yang valid adalah `default`.

```

aws kms put-key-policy \
  --policy-name default \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --policy file://key_policy.json

```

Perintah ini tidak menghasilkan output apa pun. Untuk memverifikasi bahwa perintah itu efektif, gunakan `get-key-policy` perintah. Contoh perintah berikut mendapatkan kebijakan kunci

untuk kunci KMS yang sama. `outputParameter` dengan nilai `text` mengembalikan format teks yang mudah dibaca.

```
aws kms get-key-policy \  
  --policy-name default \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --output text
```

Output:

```
{  
  "Version" : "2012-10-17",  
  "Id" : "key-default-1",  
  "Statement" : [  
    {  
      "Sid" : "Enable IAM User Permissions",  
      "Effect" : "Allow",  
      "Principal" : {  
        "AWS" : "arn:aws:iam::111122223333:root"  
      },  
      "Action" : "kms:*",  
      "Resource" : "*"   
    },  
    {  
      "Sid" : "Allow Use of Key",  
      "Effect" : "Allow",  
      "Principal" : {  
        "AWS" : "arn:aws:iam::111122223333:user/test-user"  
      },  
      "Action" : [ "kms:Describe", "kms:List" ],  
      "Resource" : "*"   
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengubah Kebijakan Utama](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [PutKeyPolicy](#) di Referensi AWS CLI Perintah.

## re-encrypt

Contoh kode berikut menunjukkan cara menggunakan `re-encrypt`.

### AWS CLI

Contoh 1: Untuk mengenkripsi ulang pesan terenkripsi di bawah kunci KMS simetris yang berbeda (Linux dan macOS).

Contoh `re-encrypt` perintah berikut menunjukkan cara yang disarankan untuk mengenkripsi ulang data dengan CLI. AWS

Berikan `ciphertext` dalam file. Dalam nilai `--ciphertext-blob` parameter, gunakan `fileb://` awalan, yang memberitahu CLI untuk membaca data dari file biner. Jika file tidak ada di direktori saat ini, ketik path lengkap ke file. Untuk informasi lebih lanjut tentang membaca nilai parameter AWS CLI dari file, lihat Memuat parameter AWS CLI dari file < <https://docs.aws.amazon.com/cli/latest/userguide/cli-usage-parameters-file.html> > di Panduan Pengguna Antarmuka Baris AWS Perintah dan Praktik Terbaik untuk Parameter File Lokal < <https://aws.amazon.com/blogs/developer/best-practices-for-local-file-parameters/> > di Blog Alat Baris Perintah. Tentukan kunci KMS sumber, yang mendekripsi `ciphertext`. Parameter tidak AWS diperlukan saat mendekripsi dengan simetris `--source-key-id` kunci enkripsi KMS. AWS KMS bisa mendapatkan kunci KMS yang digunakan untuk mengenkripsi data dari metadata di gumpalan `ciphertext`. Tetapi selalu merupakan praktik terbaik untuk menentukan kunci KMS yang Anda gunakan. Praktik ini memastikan bahwa Anda menggunakan kunci KMS yang Anda inginkan, dan mencegah Anda mendekripsi `ciphertext` secara tidak sengaja menggunakan kunci KMS yang tidak Anda percayai. Tentukan kunci KMS tujuan, yang mengenkripsi ulang data. Parameter selalu diperlukan. `--destination-key-id` Contoh ini menggunakan ARN kunci, tetapi Anda dapat menggunakan pengidentifikasi kunci yang valid. Minta output teks biasa sebagai nilai teks. Parameter `--query` memberitahu CLI untuk mendapatkan hanya nilai bidang dari output. `PlainText` `--output` Parameter mengembalikan output sebagai `text.base64-decode plaintext` dan menyimpannya dalam file. Contoh berikut pipa (`|`) nilai `PlainText` parameter ke utilitas Base64, yang menerjemahkan itu. Kemudian, itu mengalihkan (`>`) output yang diterjemahkan ke file. `ExamplePlainText`

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan pengidentifikasi kunci yang valid dari AWS akun Anda.

```
aws kms re-encrypt \  
  --ciphertext-blob fileb://ExampleEncryptedFile \  
  --source-key-id   
  --destination-key-id   
  --query   
  --output   
  ExamplePlainText
```

```
--source-key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
--destination-key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \  
--query CiphertextBlob \  
--output text | base64 --decode > ExampleReEncryptedFile
```

Perintah ini tidak menghasilkan output. Output dari `re-encrypt` perintah adalah base64-decoded dan disimpan dalam file.

Untuk informasi selengkapnya, lihat `ReEncrypt` < [https://docs.aws.amazon.com/kms/latest/APIReference/API\\_ReEncrypt.html](https://docs.aws.amazon.com/kms/latest/APIReference/API_ReEncrypt.html) di Referensi API Layanan Manajemen AWS Kunci.

Contoh 2: Untuk mengenkripsi ulang pesan terenkripsi di bawah tombol KMS simetris yang berbeda (prompt perintah Windows).

Contoh `re-encrypt` perintah berikut adalah sama dengan yang sebelumnya kecuali bahwa ia menggunakan `certutil` utilitas untuk Base64-decode data plaintext. Prosedur ini membutuhkan dua perintah, seperti yang ditunjukkan pada contoh berikut.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms re-encrypt ^  
  --ciphertext-blob fileb://ExampleEncryptedFile ^  
  --source-key-id 1234abcd-12ab-34cd-56ef-1234567890ab ^  
  --destination-key-id 0987dcba-09fe-87dc-65ba-ab0987654321 ^  
  --query CiphertextBlob ^  
  --output text > ExampleReEncryptedFile.base64
```

Kemudian gunakan `certutil` utilitas

```
certutil -decode ExamplePlaintextFile.base64 ExamplePlaintextFile
```

Output:

```
Input Length = 18  
Output Length = 12  
CertUtil: -decode command completed successfully.
```

Untuk informasi selengkapnya, lihat `ReEncrypt` < [https://docs.aws.amazon.com/kms/latest/APIReference/API\\_ReEncrypt.html](https://docs.aws.amazon.com/kms/latest/APIReference/API_ReEncrypt.html) di Referensi API Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ReEncrypt](#) di Referensi AWS CLI Perintah.

## **retire-grant**

Contoh kode berikut menunjukkan cara menggunakan `retire-grant`.

### AWS CLI

Untuk pensiun hibah pada kunci master pelanggan

`retire-grant` Contoh berikut menghapus hibah dari kunci KMS.

Contoh perintah berikut menentukan `grant-id` dan `key-id` parameter. Nilai `key-id` parameter harus menjadi kunci ARN dari kunci KMS.

```
aws kms retire-grant \
  --grant-id 1234a2345b8a4e350500d432bccf8ecd6506710e1391880c4f7f7140160c9af3 \
  --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output. Untuk mengonfirmasi bahwa hibah telah pensiun, gunakan `list-grants` perintah.

Untuk informasi selengkapnya, lihat [Pensiun dan pencabutan hibah di Panduan Pengembang Layanan Manajemen AWS Utama](#).

- Untuk detail API, lihat [RetireGrant](#) di Referensi AWS CLI Perintah.

## **revoke-grant**

Contoh kode berikut menunjukkan cara menggunakan `revoke-grant`.

### AWS CLI

Untuk mencabut hibah pada kunci master pelanggan

`revoke-grant` Contoh berikut menghapus hibah dari kunci KMS. Contoh perintah berikut menentukan `grant-id` dan `key-id` parameter. Nilai `key-id` parameter dapat berupa ID kunci atau kunci ARN dari kunci KMS.

```
aws kms revoke-grant \
  --grant-id 1234a2345b8a4e350500d432bccf8ecd6506710e1391880c4f7f7140160c9af3 \
```

```
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output. Untuk mengonfirmasi bahwa hibah telah dicabut, gunakan `list-grants` perintah.

Untuk informasi selengkapnya, lihat [Pensiun dan pencabutan hibah di Panduan Pengembang Layanan Manajemen AWS Utama](#).

- Untuk detail API, lihat [RevokeGrant](#) di Referensi AWS CLI Perintah.

## rotate-key-on-demand

Contoh kode berikut menunjukkan cara menggunakan `rotate-key-on-demand`.

### AWS CLI

Untuk melakukan rotasi on-demand dari kunci KMS

`rotate-key-on-demand` Contoh berikut segera memulai rotasi bahan kunci untuk kunci KMS yang ditentukan.

```
aws kms rotate-key-on-demand \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{  
  "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

Untuk informasi selengkapnya, lihat [Cara melakukan rotasi kunci sesuai permintaan](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [RotateKeyOnDemand](#) di Referensi AWS CLI Perintah.

## schedule-key-deletion

Contoh kode berikut menunjukkan cara menggunakan `schedule-key-deletion`.

### AWS CLI

Untuk menjadwalkan penghapusan kunci KMS yang dikelola pelanggan.



`schedule-key-deletion` Contoh berikut menjadwalkan kunci KMS terkelola pelanggan yang ditentukan untuk dihapus dalam 15 hari.

`--key-id` Parameter mengidentifikasi kunci KMS. Contoh ini menggunakan nilai ARN kunci, tetapi Anda dapat menggunakan ID kunci atau ARN dari kunci `--pending-window-in-days` KMS. Parameter menentukan panjang masa tunggu 7-30 hari. Secara default, masa tunggu adalah 30 hari. Contoh ini menentukan nilai 15, yang memberitahu AWS untuk secara permanen menghapus kunci KMS 15 hari setelah perintah selesai.

```
aws kms schedule-key-deletion \  
  --key-id arn:aws:kms:us-  
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab \  
  --pending-window-in-days 15
```

Respons termasuk ARN kunci, status kunci, periode tunggu (`PendingWindowInDays`), dan tanggal penghapusan dalam waktu Unix. Untuk melihat tanggal penghapusan dalam waktu setempat, gunakan konsol AWS KMS. Kunci KMS dalam status `PendingDeletion` kunci tidak dapat digunakan dalam operasi kriptografi.

```
{  
  "KeyId": "arn:aws:kms:us-  
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
  "DeletionDate": "2022-06-18T23:43:51.272000+00:00",  
  "KeyState": "PendingDeletion",  
  "PendingWindowInDays": 15  
}
```

Untuk informasi selengkapnya, lihat [Menghapus kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [ScheduleKeyDeletion](#) di Referensi AWS CLI Perintah.

## sign

Contoh kode berikut menunjukkan cara menggunakan `sign`.

### AWS CLI

Contoh 1: Untuk menghasilkan tanda tangan digital untuk pesan

`sign` Contoh berikut menghasilkan tanda tangan kriptografi untuk pesan singkat. Output dari perintah mencakup `Signature` bidang yang dikodekan basis-64 yang dapat Anda verifikasi dengan menggunakan perintah `verify`

Anda harus menentukan pesan yang akan ditandatangani dan algoritma penandatanganan yang didukung oleh kunci KMS asimetris Anda. Untuk mendapatkan algoritma penandatanganan untuk kunci KMS Anda, gunakan perintah `describe-key`

Di AWS CLI 2.0, nilai `message` parameter harus dikodekan Base64. Atau, Anda dapat menyimpan pesan dalam file dan menggunakan `fileb://` awalan, yang memberitahu AWS CLI untuk membaca data biner dari file.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda. ID kunci harus mewakili kunci KMS asimetris dengan penggunaan kunci `SIGN_VERIFY`.

```
msg=(echo 'Hello World' | base64)

aws kms sign \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --message fileb://UnsignedMessage \
  --message-type RAW \
  --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256
```

Output:

```
{
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "Signature": "ABCDEFhpyVYyTxbafE74ccSvEJLJr3zuoV1Hfymz4qv+
fxmxNLA7SE1SiF8lHw80fKZZ3bJ...",
  "SigningAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"
}
```

Untuk informasi selengkapnya tentang penggunaan kunci KMS asimetris di AWS KMS, lihat [Kunci asimetris di AWS KMS di Panduan Pengembang](#) Layanan Manajemen AWS Kunci.

Contoh 2: Untuk menyimpan tanda tangan digital dalam file (Linux dan macOS)

`sign` Contoh berikut menghasilkan tanda tangan kriptografi untuk pesan singkat yang disimpan dalam file lokal. Perintah juga mendapatkan `Signature` properti dari respons, Base64-

mendekodekannya dan menyimpannya dalam file. `ExampleSignature` Anda dapat menggunakan file tanda tangan dalam `verify` perintah yang memverifikasi tanda tangan.

`sign` Perintah ini memerlukan pesan yang disandikan Base64 dan algoritma penandatanganan yang didukung oleh kunci KMS asimetris Anda. Untuk mendapatkan algoritma penandatanganan yang didukung oleh kunci KMS Anda, gunakan perintah `describe-key`

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda. ID kunci harus mewakili kunci KMS asimetris dengan penggunaan kunci `SIGN_VERIFY`.

```
echo 'hello world' | base64 > EncodedMessage

aws kms sign \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --message fileb://EncodedMessage \
  --message-type RAW \
  --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256 \
  --output text \
  --query Signature | base64 --decode > ExampleSignature
```

Perintah ini tidak menghasilkan output. Contoh ini mengekstrak `Signature` properti output dan menyimpannya dalam file.

Untuk informasi selengkapnya tentang penggunaan kunci KMS asimetris di AWS KMS, lihat [Kunci asimetris di AWS KMS di Panduan Pengembang](#) Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat Referensi AWS CLI Perintah [Masuk](#).

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke kunci KMS

`tag-resource` Contoh berikut menambahkan `"Purpose": "Test"` dan `"Dept": "IT"` tag ke kunci KMS yang dikelola pelanggan. Anda dapat menggunakan tag seperti ini untuk memberi label kunci KMS dan membuat kategori kunci KMS untuk izin dan audit.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan nilai ID kunci, tetapi Anda dapat menggunakan ID kunci atau kunci ARN dalam perintah ini.

```
aws kms tag-resource \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --tags TagKey='Purpose',TagValue='Test' TagKey='Dept',TagValue='IT'
```

Perintah ini tidak menghasilkan output. Untuk melihat tag pada tombol AWS KMS KMS, gunakan perintah. `list-resource-tags`

Untuk informasi selengkapnya tentang penggunaan tag di AWS KMS, lihat [Menandai kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari kunci KMS

`untag-resource` Contoh berikut menghapus tag dengan "Purpose" kunci dari kunci KMS yang dikelola pelanggan.

Untuk menentukan kunci KMS, gunakan `key-id` parameter. Contoh ini menggunakan nilai ID kunci, tetapi Anda dapat menggunakan ID kunci atau kunci ARN dalam perintah ini. Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms untag-resource \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --tag-key 'Purpose'
```

Perintah ini tidak menghasilkan output. Untuk melihat tag pada tombol AWS KMS KMS, gunakan perintah. `list-resource-tags`

Untuk informasi selengkapnya tentang penggunaan tag di AWS KMS, lihat [Menandai kunci](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-alias

Contoh kode berikut menunjukkan cara menggunakan `update-alias`.

### AWS CLI

Untuk mengaitkan alias dengan kunci KMS yang berbeda

`update-alias` Contoh berikut mengaitkan alias `alias/test-key` dengan kunci KMS yang berbeda.

`--alias-name` Parameter menentukan alias. Nilai nama alias harus dimulai dengan `alias/`. `--target-key-id` Parameter menentukan kunci KMS untuk mengaitkan dengan alias. Anda tidak perlu menentukan kunci KMS saat ini untuk alias.

```
aws kms update-alias \  
  --alias-name alias/test-key \  
  --target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Perintah ini tidak menghasilkan output. Untuk menemukan alias, gunakan `list-aliases` perintah.

Untuk informasi selengkapnya, lihat [Memperbarui alias](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [UpdateAlias](#) di Referensi AWS CLI Perintah.

## update-custom-key-store

Contoh kode berikut menunjukkan cara menggunakan `update-custom-key-store`.

### AWS CLI

Contoh 1: Untuk mengedit nama ramah dari toko kunci kustom

`update-custom-key-store` Contoh berikut mengubah nama toko kunci kustom. Contoh ini berfungsi untuk penyimpanan kunci AWS CloudHSM atau penyimpanan kunci eksternal.

Gunakan `custom-key-store-id` untuk mengidentifikasi toko kunci. Gunakan `new-custom-key-store-name` parameter untuk menentukan nama ramah baru.

Untuk memperbarui nama ramah dari toko kunci AWS CloudHSM, Anda harus terlebih dahulu memutuskan penyimpanan kunci, seperti dengan menggunakan perintah `disconnect-custom-key-store`. Anda dapat memperbarui nama ramah dari toko kunci eksternal saat terhubung atau terputus. Untuk menemukan status koneksi toko kunci kustom Anda, gunakan `describe-custom-key-store` perintah.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --new-custom-key-store-name ExampleKeyStore
```

Perintah ini tidak mengembalikan data apa pun. Untuk memverifikasi bahwa perintah berfungsi, gunakan `describe-custom-key-stores` perintah.

Untuk informasi selengkapnya tentang memperbarui penyimpanan kunci AWS CloudHSM, lihat [Mengedit setelah penyimpanan kunci AWS CloudHSM di Panduan Pengembang Layanan Manajemen Kunci.AWS](#)

Untuk informasi selengkapnya tentang memperbarui penyimpanan kunci eksternal, lihat [Mengedit properti penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 2: Untuk mengedit kata sandi `kmsuser` dari toko kunci AWS CloudHSM

`update-custom-key-store` Contoh berikut memperbarui nilai `kmsuser` kata sandi ke kata sandi saat ini untuk `kmsuser` di klaster CloudHSM yang terkait dengan penyimpanan kunci yang ditentukan. Perintah ini tidak mengubah `kmsuser` kata sandi klaster. Itu hanya memberi tahu AWS KMS kata sandi saat ini. Jika KMS tidak memiliki `kmsuser` kata sandi saat ini, KMS tidak dapat terhubung ke toko kunci AWS CloudHSM.

**CATATAN:** Sebelum memperbarui toko kunci AWS CloudHSM, Anda harus memutusnya. Gunakan perintah `disconnect-custom-key-store`. Setelah perintah selesai, Anda dapat menghubungkan kembali toko kunci CloudHSM AWS . Gunakan perintah `connect-custom-key-store`.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --key-store-password ExamplePassword
```

Perintah ini tidak memberikan output apa pun. Untuk memverifikasi bahwa perubahan itu efektif, gunakan `describe-custom-key-stores` perintah.

Untuk informasi selengkapnya tentang memperbarui penyimpanan kunci AWS CloudHSM, lihat [Mengedit setelah penyimpanan kunci AWS CloudHSM di Panduan Pengembang Layanan Manajemen Kunci.AWS](#)

Contoh 3: Untuk mengedit klaster AWS CloudHSM dari toko kunci CloudHSM AWS

Contoh berikut mengubah klaster AWS CloudHSM yang dikaitkan dengan penyimpanan kunci AWS CloudHSM ke cluster terkait, seperti cadangan yang berbeda dari cluster yang sama.

CATATAN: Sebelum memperbarui toko kunci AWS CloudHSM, Anda harus memutusnya. Gunakan perintah `disconnect-custom-key-store`. Setelah perintah selesai, Anda dapat menghubungkan kembali toko kunci CloudHSM AWS . Gunakan perintah `connect-custom-key-store`.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --cloud-hsm-cluster-id cluster-1a23b4cdefg
```

Perintah ini tidak memberikan output apa pun. Untuk memverifikasi bahwa perubahan itu efektif, gunakan `describe-custom-key-stores` perintah.

Untuk informasi selengkapnya tentang memperbarui penyimpanan kunci AWS CloudHSM, lihat [Mengedit setelah penyimpanan kunci AWS CloudHSM di Panduan Pengembang Layanan Manajemen Kunci.AWS](#)

Contoh 4: Untuk mengedit kredensi otentikasi proxy dari penyimpanan kunci eksternal

Contoh berikut memperbarui kredensi otentikasi proxy untuk penyimpanan kunci eksternal Anda. Anda harus menentukan keduanya `raw-secret-access-key` dan `access-key-id`, bahkan jika Anda hanya mengubah salah satu nilai. Anda dapat menggunakan fitur ini untuk memperbaiki kredensi yang tidak valid atau untuk mengubah kredensialnya saat proxy penyimpanan kunci eksternal memutarnya.

Tetapkan kredensi otentikasi proxy untuk AWS KMS di toko kunci eksternal Anda. Kemudian gunakan perintah ini untuk memberikan kredensi ke AWS KMS. AWS KMS menggunakan kredensi ini untuk menandatangani permintaannya ke proxy penyimpanan kunci eksternal Anda.

Anda dapat memperbarui kredensi otentikasi proxy saat penyimpanan kunci eksternal terhubung atau terputus. Untuk menemukan status koneksi toko kunci kustom Anda, gunakan `describe-custom-key-store` perintah.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,  
  RawSecretAccessKey=DXjSUawne12fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Perintah ini tidak memberikan output apa pun. Untuk memverifikasi bahwa perubahan itu efektif, gunakan `describe-custom-key-stores` perintah.

Untuk informasi selengkapnya tentang memperbarui penyimpanan kunci eksternal, lihat [Mengedit properti penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

Contoh 5: Untuk mengedit konektivitas proxy dari penyimpanan kunci eksternal

Contoh berikut mengubah opsi konektivitas proxy penyimpanan kunci eksternal dari konektivitas titik akhir publik ke konektivitas layanan titik akhir VPC. Selain mengubah `xks-proxy-connectivity` nilai, Anda harus mengubah `xks-proxy-uri-endpoint` nilai untuk mencerminkan nama DNS pribadi yang terkait dengan layanan titik akhir VPC. Anda juga harus menambahkan `xks-proxy-vpc-endpoint-service-name` nilai.

**CATATAN:** Sebelum memperbarui konektivitas proxy dari toko eksternal, Anda harus memutusnya. Gunakan perintah `disconnect-custom-key-store`. Setelah perintah selesai, Anda dapat menghubungkan kembali penyimpanan kunci eksternal dengan menggunakan perintah `connect-custom-key-store`

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \  
  --xks-proxy-uri-endpoint "https://myproxy-private.xks.example.com" \  
  --xks-proxy-vpc-endpoint-service-name "com.amazonaws.vpce.us-east-1.vpce-svc-  
  example"
```

Perintah ini tidak memberikan output apa pun. Untuk memverifikasi bahwa perubahan itu efektif, gunakan `describe-custom-key-stores` perintah.

Untuk informasi selengkapnya tentang memperbarui penyimpanan kunci eksternal, lihat [Mengedit properti penyimpanan kunci eksternal](#) di Panduan Pengembang Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [UpdateCustomKeyStore](#) di Referensi AWS CLI Perintah.



## update-key-description

Contoh kode berikut menunjukkan cara menggunakan `update-key-description`.

### AWS CLI

Contoh 1: Untuk menambah atau mengubah deskripsi ke kunci KMS yang dikelola pelanggan

`update-key-description` Contoh berikut menambahkan deskripsi ke kunci KMS yang dikelola pelanggan. Anda dapat menggunakan perintah yang sama untuk mengubah deskripsi yang ada.

`--key-id` Parameter mengidentifikasi kunci KMS dalam perintah. Contoh ini menggunakan nilai ARN kunci, tetapi Anda dapat menggunakan ID kunci atau ARN kunci dari kunci KMS. Parameter menentukan deskripsi baru `--description`. Nilai parameter ini menggantikan deskripsi kunci KMS saat ini, jika ada.

```
aws kms update-key-description \
  --key-id arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab \
  --description "IT Department test key"
```

Perintah ini tidak menghasilkan output. Untuk melihat deskripsi kunci KMS, gunakan `describe-key` perintah.

Untuk informasi selengkapnya, lihat [UpdateKeyDescription](#) di Referensi API Layanan Manajemen AWS Kunci.

Contoh 2: Untuk menghapus deskripsi kunci KMS yang dikelola pelanggan

`update-key-description` Contoh berikut menghapus deskripsi ke kunci KMS yang dikelola pelanggan.

`--key-id` Parameter mengidentifikasi kunci KMS dalam perintah. Contoh ini menggunakan nilai ID kunci, tetapi Anda dapat menggunakan ID kunci atau ARN kunci dari kunci KMS. Parameter dengan nilai string kosong (") menghapus deskripsi yang ada.

```
aws kms update-key-description \
  --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
  --description ''
```

Perintah ini tidak menghasilkan output. Untuk melihat deskripsi kunci KMS, gunakan perintah `deskripsi-kunci`.

Untuk informasi selengkapnya, lihat [UpdateKeyDescription](#) di Referensi API Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [UpdateKeyDescription](#) di Referensi AWS CLI Perintah.

## verify

Contoh kode berikut menunjukkan cara menggunakan `verify`.

### AWS CLI

Untuk memverifikasi tanda tangan digital

`verify` Contoh berikut memverifikasi tanda tangan kriptografi untuk pesan singkat yang disandikan Base64. ID kunci, pesan, jenis pesan, dan algoritma penandatanganan harus sama dengan yang digunakan untuk menandatangani pesan. Tanda tangan yang Anda tentukan tidak dapat dikodekan base64. Untuk bantuan mendekode tanda tangan yang dikembalikan `sign` perintah, lihat contoh `sign` perintah.

Output dari perintah termasuk `SignatureValid` bidang Boolean yang menunjukkan bahwa tanda tangan telah diverifikasi. Jika validasi tanda tangan gagal, `verify` perintah juga gagal.

Sebelum menjalankan perintah ini, ganti ID kunci contoh dengan ID kunci yang valid dari AWS akun Anda.

```
aws kms verify \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --message fileb://EncodedMessage \
  --message-type RAW \
  --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256 \
  --signature fileb://ExampleSignature
```

Output:

```
{
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "SignatureValid": true,
  "SigningAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"
}
```

Untuk informasi selengkapnya tentang penggunaan kunci KMS asimetris di AWS KMS, lihat [Menggunakan kunci asimetris di Panduan Pengembang](#) Layanan Manajemen AWS Kunci.

- Untuk detail API, lihat [Verifikasi](#) di Referensi AWS CLI Perintah.

## Contoh Lake Formation menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Lake Formation.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-lf-tags-to-resource**

Contoh kode berikut menunjukkan cara menggunakan `add-lf-tags-to-resource`.

AWS CLI

Untuk melampirkan satu atau beberapa LF-tag ke sumber daya yang ada

`add-lf-tags-to-resource` Contoh berikut melampirkan LF-tag yang diberikan ke sumber daya tabel.

```
aws lakeformation add-lf-tags-to-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_promotion"
    }
  },
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "analyst"
    ]
  }]
}
```

Output:

```
{
  "Failures": []
}
```

Untuk informasi selengkapnya, lihat [Menetapkan LF-tag ke sumber daya Katalog Data di Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [AddLfTagsToResource](#) di Referensi AWS CLI Perintah.

## batch-grant-permissions

Contoh kode berikut menunjukkan cara menggunakan `batch-grant-permissions`.

### AWS CLI

Untuk memberikan izin massal pada sumber daya kepada kepala sekolah

`batch-grant-permissions` Contoh berikut massal memberikan akses pada sumber daya tertentu ke kepala sekolah.

```
aws lakeformation batch-grant-permissions \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Entries": [{
    "Id": "1",
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_promotion"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": [
      "ALL"
    ]
  },
  {
    "Id": "2",
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_customer"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": [
      "ALL"
    ]
  },
}
```

```
{
  "Id": "3",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
business-analyst"
  },
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_promotion"
    }
  },
  "Permissions": [
    "ALL"
  ],
  "PermissionsWithGrantOption": [
    "ALL"
  ]
},
{
  "Id": "4",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
  },
  "Resource": {
    "DataCellsFilter": {
      "TableCatalogId": "123456789111",
      "DatabaseName": "tpc",
      "TableName": "dl_tpc_item",
      "Name": "developer_item"
    }
  },
  "Permissions": [
    "SELECT"
  ],
  "PermissionsWithGrantOption": []
}
]
```

Output:

```
{
  "Failures": []
}
```

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [BatchGrantPermissions](#) di Referensi AWS CLI Perintah.

## batch-revoke-permissions

Contoh kode berikut menunjukkan cara menggunakan `batch-revoke-permissions`.

### AWS CLI

Untuk mencabut izin secara massal pada sumber daya dari prinsipal

`batch-revoke-permissions` Contoh berikut massal mencabut akses pada sumber daya tertentu dari prinsipal.

```
aws lakeformation batch-revoke-permissions \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Entries": [{
    "Id": "1",
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_promotion"
      }
    },
    "Permissions": [
```

```

        "ALL"
      ],
      "PermissionsWithGrantOption": [
        "ALL"
      ]
    },
    {
      "Id": "2",
      "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
business-analyst"
      },
      "Resource": {
        "Table": {
          "CatalogId": "123456789111",
          "DatabaseName": "tpc",
          "Name": "dl_tpc_promotion"
        }
      },
      "Permissions": [
        "ALL"
      ],
      "PermissionsWithGrantOption": [
        "ALL"
      ]
    }
  ]
}

```

Output:

```

{
  "Failures": []
}

```

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [BatchRevokePermissions](#) di Referensi AWS CLI Perintah.

## cancel-transaction

Contoh kode berikut menunjukkan cara menggunakan `cancel-transaction`.



## AWS CLI

Untuk membatalkan transaksi

`cancel-transaction` Contoh berikut membatalkan transaksi.

```
aws lakeformation cancel-transaction \  
  --transaction-id='b014d972ca8347b89825e33c5774aec4'
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [CancelTransaction](#) di Referensi AWS CLI Perintah.

## `commit-transaction`

Contoh kode berikut menunjukkan cara menggunakan `commit-transaction`.

## AWS CLI

Untuk melakukan transaksi

`commit-transaction` Contoh berikut melakukan transaksi.

```
aws lakeformation commit-transaction \  
  --transaction-id='b014d972ca8347b89825e33c5774aec4'
```

Output:

```
{  
  "TransactionStatus": "committed"  
}
```

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [CommitTransaction](#) di Referensi AWS CLI Perintah.

## create-data-cells-filter

Contoh kode berikut menunjukkan cara menggunakan `create-data-cells-filter`.

### AWS CLI

Contoh 1: Untuk membuat filter sel data

`create-data-cells-filter` Contoh berikut membuat filter sel data untuk memungkinkan seseorang memberikan akses ke kolom tertentu berdasarkan kondisi baris.

```
aws lakeformation create-data-cells-filter \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "TableData": {  
    "ColumnNames": ["p_channel_details", "p_start_date_sk", "p_promo_name"],  
    "DatabaseName": "tpc",  
    "Name": "developer_promotion",  
    "RowFilter": {  
      "FilterExpression": "p_promo_name='ese'"  
    },  
    "TableCatalogId": "123456789111",  
    "TableName": "dl_tpc_promotion"  
  }  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pemfilteran data dan keamanan tingkat sel di Lake Formation di Panduan Pengembang AWS Lake Formation](#).

Contoh 2: Untuk membuat filter kolom

`create-data-cells-filter` Contoh berikut membuat filter data untuk memungkinkan seseorang untuk memberikan akses ke kolom tertentu.

```
aws lakeformation create-data-cells-filter \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "TableData": {
    "ColumnNames": ["p_channel_details", "p_start_date_sk", "p_promo_name"],
    "DatabaseName": "tpc",
    "Name": "developer_promotion_allrows",
    "RowFilter": {
      "AllRowsWildcard": {}
    },
    "TableCatalogId": "123456789111",
    "TableName": "dl_tpc_promotion"
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pemfilteran data dan keamanan tingkat sel di Lake Formation di Panduan Pengembang AWS Lake Formation](#).

Contoh 3: Untuk membuat filter data dengan kolom kecualikan

`create-data-cells-filter` Contoh berikut membuat filter data untuk memungkinkan seseorang untuk memberikan akses semua kecuali kolom yang disebutkan.

```
aws lakeformation create-data-cells-filter \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "TableData": {
    "ColumnWildcard": {
      "ExcludedColumnNames": ["p_channel_details", "p_start_date_sk"]
    },
    "DatabaseName": "tpc",
    "Name": "developer_promotion_excludecolumn",
    "RowFilter": {
      "AllRowsWildcard": {}
    },
    "TableCatalogId": "123456789111",
    "TableName": "dl_tpc_promotion"
  }
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pemfilteran data dan keamanan tingkat sel di Lake Formation di Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [CreateDataCellsFilter](#) di Referensi AWS CLI Perintah.

## **create-lf-tag**

Contoh kode berikut menunjukkan cara menggunakan `create-lf-tag`.

### AWS CLI

Untuk membuat LF-tag

`create-lf-tag` Contoh berikut membuat LF-tag dengan nama dan nilai yang ditentukan.

```
aws lakeformation create-lf-tag \  
  --catalog-id '123456789111' \  
  --tag-key 'usergroup' \  
  --tag-values '["developer","analyst","campaign"]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola LF-tag untuk kontrol akses metadata](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [CreateLfTag](#) di Referensi AWS CLI Perintah.

## **delete-data-cells-filter**

Contoh kode berikut menunjukkan cara menggunakan `delete-data-cells-filter`.

### AWS CLI

Untuk menghapus filter sel data

`delete-data-cells-filter` Contoh berikut menghapus filter sel data yang diberikan.

```
aws lakeformation delete-data-cells-filter \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "TableCatalogId": "123456789111",
  "DatabaseName": "tpc",
  "TableName": "dl_tpc_promotion",
  "Name": "developer_promotion"
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pemfilteran data dan keamanan tingkat sel di Lake Formation di Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [DeleteDataCellsFilter](#) di Referensi AWS CLI Perintah.

## **delete-lf-tag**

Contoh kode berikut menunjukkan cara menggunakan `delete-lf-tag`.

### AWS CLI

Untuk menghapus definisi LF-tag

`delete-lf-tag` Contoh berikut menghapus definisi LF-tag.

```
aws lakeformation delete-lf-tag \
  --catalog-id '123456789111' \
  --tag-key 'usergroup'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola LF-tag untuk kontrol akses metadata](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [DeleteLfTag](#) di Referensi AWS CLI Perintah.

## **delete-objects-on-cancel**

Contoh kode berikut menunjukkan cara menggunakan `delete-objects-on-cancel`.

### AWS CLI

Untuk menghapus objek saat transaksi dibatalkan

`delete-objects-on-cancel` Contoh berikut menghapus objek s3 yang terdaftar saat transaksi dibatalkan.

```
aws lakeformation delete-objects-on-cancel \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "012345678901",  
  "DatabaseName": "tpc",  
  "TableName": "dl_tpc_household_demographics_gov",  
  "TransactionId": "1234d972ca8347b89825e33c5774aec4",  
  "Objects": [{  
    "Uri": "s3://lf-data-lake-012345678901/target/  
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-  
ff26b17504414fe88b302cd795eabd00.parquet",  
    "ETag": "1234ab1fc50a316b149b4e1f21a73800"  
  }]  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [DeleteObjectsOnCancel](#) di Referensi AWS CLI Perintah.

## deregister-resource

Contoh kode berikut menunjukkan cara menggunakan `deregister-resource`.

### AWS CLI

Untuk membatalkan pendaftaran penyimpanan data lake

`deregister-resource` Contoh berikut membatalkan pendaftaran sumber daya seperti yang dikelola oleh Lake Formation.

```
aws lakeformation deregister-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123"
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan lokasi Amazon S3 ke danau data Anda di Panduan](#) Pengembang AWS Lake Formation.

- Untuk detail API, lihat [DeregisterResource](#) di Referensi AWS CLI Perintah.

## **describe-transaction**

Contoh kode berikut menunjukkan cara menggunakan `describe-transaction`.

### AWS CLI

Untuk mengambil detail transaksi

`describe-transaction` Contoh berikut mengembalikan rincian transaksi tunggal.

```
aws lakeformation describe-transaction \
  --transaction-id='8cb4b1a7cc8d486fbaca9a64e7d9f5ce'
```

Output:

```
{
  "TransactionDescription": {
    "TransactionId": "12345972ca8347b89825e33c5774aec4",
    "TransactionStatus": "committed",
    "TransactionStartTime": "2022-08-10T14:29:04.046000+00:00",
    "TransactionEndTime": "2022-08-10T14:29:09.681000+00:00"
  }
}
```

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [DescribeTransaction](#) di Referensi AWS CLI Perintah.

## extend-transaction

Contoh kode berikut menunjukkan cara menggunakan `extend-transaction`.

### AWS CLI

Untuk memperpanjang transaksi

`extend-transaction` Contoh berikut memperluas transaksi.

```
aws lakeformation extend-transaction \  
  --transaction-id='8cb4b1a7cc8d486fbaca9a64e7d9f5ce'
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [ExtendTransaction](#) di Referensi AWS CLI Perintah.

## get-data-lake-settings

Contoh kode berikut menunjukkan cara menggunakan `get-data-lake-settings`.

### AWS CLI

Untuk mengambil pengaturan AWS danau data yang dikelola Lake Formation

`get-data-lake-settings` Contoh berikut mengambil daftar administrator data lake dan pengaturan data lake lainnya.

```
aws lakeformation get-data-lake-settings \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "123456789111"  
}
```

Output:

```
{
```



```

    "DataLakeSettings": {
      "DataLakeAdmins": [{
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"
      }],
      "CreateDatabaseDefaultPermissions": [],
      "CreateTableDefaultPermissions": [
        {
          "Principal": {
            "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
          },
          "Permissions": [
            "ALL"
          ]
        }
      ],
      "TrustedResourceOwners": [],
      "AllowExternalDataFiltering": true,
      "ExternalDataFilteringAllowList": [{
        "DataLakePrincipalIdentifier": "123456789111"
      }],
      "AuthorizedSessionTagValueList": [
        "Amazon EMR"
      ]
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Mengubah setelan keamanan default untuk data lake Anda](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetDataLakeSettings](#) di Referensi AWS CLI Perintah.

## get-effective-permissions-for-path

Contoh kode berikut menunjukkan cara menggunakan `get-effective-permissions-for-path`.

### AWS CLI

Untuk mengambil izin pada sumber daya yang terletak di jalur tertentu

`get-effective-permissions-for-path` Contoh berikut mengembalikan izin Lake Formation untuk tabel atau sumber database tertentu yang terletak di jalur di Amazon S3.

```
aws lakeformation get-effective-permissions-for-path \
```

```
--cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111"
}
```

Output:

```
{
  "Permissions": [{
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
campaign-manager"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": []
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/EMR-
RuntimeRole"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": []
  },
  {
```

```
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:saml-
provider/oktaSAMLProvider:user/emr-developer"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "ALL",
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": []
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "ALL",
      "ALTER",
      "CREATE_TABLE",
      "DESCRIBE",
      "DROP"
    ],
    "PermissionsWithGrantOption": [
      "ALL",
      "ALTER",
      "CREATE_TABLE",
      "DESCRIBE",
      "DROP"
    ]
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/LF-
GlueServiceRole"
```

```

    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "CREATE_TABLE"
    ],
    "PermissionsWithGrantOption": []
  }
],
"NextToken":
"E5S1JDSTZ1eUp6SWpvaU9UQTN0RE0zTXpFeE5Ua3pJbjE5TENKbGVIQnBjbUYwYVc5dUlqcDdJbk5sWTI5dVpITW1P
}

```

Untuk informasi selengkapnya, lihat [Mengelola izin Lake Formation](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetEffectivePermissionsForPath](#) di Referensi AWS CLI Perintah.

## get-lf-tag

Contoh kode berikut menunjukkan cara menggunakan `get-lf-tag`.

### AWS CLI

Untuk mengambil definisi LF-tag

`get-lf-tag` Contoh berikut mengambil definisi LF-tag.

```

aws lakeformation get-lf-tag \
  --catalog-id '123456789111' \
  --tag-key 'usergroup'

```

Output:

```

{
  "CatalogId": "123456789111",
  "TagKey": "usergroup",
  "TagValues": [
    "analyst",

```

```
    "campaign",  
    "developer"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola LF-tag untuk kontrol akses metadata](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetLfTag](#) di Referensi AWS CLI Perintah.

## get-query-state

Contoh kode berikut menunjukkan cara menggunakan `get-query-state`.

### AWS CLI

Untuk mengambil status kueri yang dikirimkan

`get-query-state` Contoh berikut mengembalikan keadaan query sebelumnya dikirimkan.

```
aws lakeformation get-query-state \  
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Output:

```
{  
  "State": "FINISHED"  
}
```

Untuk informasi selengkapnya, lihat [Operasi data transaksional di Panduan](#) Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetQueryState](#) di Referensi AWS CLI Perintah.

## get-query-statistics

Contoh kode berikut menunjukkan cara menggunakan `get-query-statistics`.

### AWS CLI

Untuk mengambil statistik kueri

`get-query-statistics` Contoh berikut mengambil statistik pada perencanaan dan pelaksanaan query.

```
aws lakeformation get-query-statistics \  
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Output:

```
{  
  "ExecutionStatistics": {  
    "AverageExecutionTimeMillis": 0,  
    "DataScannedBytes": 0,  
    "WorkUnitsExecutedCount": 0  
  },  
  "PlanningStatistics": {  
    "EstimatedDataToScanBytes": 43235,  
    "PlanningTimeMillis": 2377,  
    "QueueTimeMillis": 440,  
    "WorkUnitsGeneratedCount": 1  
  },  
  "QuerySubmissionTime": "2022-08-11T02:14:38.641870+00:00"  
}
```

Untuk informasi selengkapnya, lihat [Operasi data transaksional di Panduan](#) Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetQueryStatistics](#) di Referensi AWS CLI Perintah.

## **get-resource-lf-tags**

Contoh kode berikut menunjukkan cara menggunakan `get-resource-lf-tags`.

AWS CLI

Untuk daftar LF-tag

`list-lf-tags` Contoh berikut mengembalikan daftar LF-tag yang pemohon memiliki izin untuk melihat.

```
aws lakeformation list-lf-tags \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "ResourceShareType": "ALL",
  "MaxResults": 2
}
```

Output:

```
{
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "category",
    "TagValues": [
      "private",
      "public"
    ]
  },
  {
    "CatalogId": "123456789111",
    "TagKey": "group",
    "TagValues": [
      "analyst",
      "campaign",
      "developer"
    ]
  }
],
  "NextToken": "kIiwiZXhwaXJhdGlvbiI6eyJzZWNVbmRzIjoxNjYwMDY4dCI6ZmFsc2V9"
}
```

Untuk informasi selengkapnya, lihat [Mengelola LF-tag untuk kontrol akses metadata](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetResourceLFTags](#) di Referensi AWS CLI Perintah.

## get-table-objects

Contoh kode berikut menunjukkan cara menggunakan `get-table-objects`.

### AWS CLI

Untuk daftar objek dari tabel yang diatur

`get-table-objects` Contoh berikut mengembalikan kumpulan objek Amazon S3 yang membentuk tabel diatur yang ditentukan.

```
aws lakeformation get-table-objects \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "012345678901",  
  "DatabaseName": "tpc",  
  "TableName": "dl_tpc_household_demographics_gov",  
  "QueryAsOfTime": "2022-08-10T15:00:00"  
}
```

Output:

```
{  
  "Objects": [{  
    "PartitionValues": [],  
    "Objects": [{  
      "Uri": "s3://lf-data-lake-012345678901/target/  
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-  
ff26b17504414fe88b302cd795eabd00.parquet",  
      "ETag": "12345b1fc50a316b149b4e1f21a73800",  
      "Size": 43235  
    }]  
  }]  
}
```

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [GetTableObjects](#) di Referensi AWS CLI Perintah.

## **get-work-unit-results**

Contoh kode berikut menunjukkan cara menggunakan `get-work-unit-results`.

### AWS CLI

Untuk mengambil unit kerja dari kueri yang diberikan



`get-work-unit-results` Contoh berikut mengembalikan unit kerja yang dihasilkan dari query.

```
aws lakeformation get-work-units \
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b' \
  --work-unit-id '0' \
  --work-unit-token 'B2fMSdmQXe9umX8Ux8XCo4=' outfile
```

Output:

```
outfile with Blob content.
```

Untuk informasi selengkapnya, lihat [Operasi data transaksional di Panduan](#) Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetWorkUnitResults](#) di Referensi AWS CLI Perintah.

## get-work-units

Contoh kode berikut menunjukkan cara menggunakan `get-work-units`.

### AWS CLI

Untuk mengambil unit kerja

`get-work-units` Contoh berikut mengambil unit kerja yang dihasilkan oleh `StartQueryPlanning` operasi.

```
aws lakeformation get-work-units \
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Output:

```
{
  "WorkUnitRanges": [{
    "WorkUnitIdMax": 0,
    "WorkUnitIdMin": 0,
    "WorkUnitToken":
      "1234eMAk4kL04umqEL4Z5WuxL04AXwABABVhd3MtY3J5cHRvLXB1YmxpYy1rZXkAREEwYm9QbkhINmFYTWphbmMxZW
      +f88jzGrYq22gE6jkQlp0B
      +0et2eqNUMfudAAAAfjB8BgkqhkiG9w0BBwagbzBtAgEAMGgGCSqGSIB3DQEHATAeBg1ghkgBZQMEAS4wEQMCOEWRda
```

```
wAAAAEAAAAAAAAAAAAAAAAAAEAAACX3/w5h75QAPomfKH+cyEKYU1yccUmB1
+VSojiG0tdsUk7vcjYXUUb0Ym3dvdqRqX2s4gROM0n
+Ij8R0/8jYmnHkpvYAFNVRPyETyIKg7k5Z9+5I1c2d3446Jw/moWGGxjH8AEG9h27ytm0hozxD0Ei/
F2ZoXz6wLGDfGUo/2WxCkY0hTyNaw6TM
+7drTM7yrW4iINVLUM0LX0xnFjIAhLhooWJek6vjQZUAZzB1AjBH8okRtYP8R7AY2W1s/
hqFBhG0V4l42AC0LxsuZbMQrE2SszWUZ0E9Uew7/n0cyX4CMQDR79INyv4ysMByW9kKGGKyba+cCNk1ExMR
+btBQBmMuB2fMSdmQXe9umX8Ux8XCo4="
  }],
  "QueryId": "1234273f-4a62-4cda-8d98-69615ee8be9b"
}
```

Untuk informasi selengkapnya, lihat [Operasi data transaksional di Panduan](#) Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GetWorkUnits](#) di Referensi AWS CLI Perintah.

## grant-permissions

Contoh kode berikut menunjukkan cara menggunakan `grant-permissions`.

### AWS CLI

Contoh 1: Untuk memberikan izin kepada prinsipal pada sumber daya menggunakan LF-tag

`grant-permissions` Contoh berikut memberikan SEMUA izin kepada prinsipal pada sumber daya database yang cocok dengan kebijakan LF-tag.

```
aws lakeformation grant-permissions \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"
  },
  "Resource": {
    "LFTagPolicy": {
      "CatalogId": "123456789111",
      "ResourceType": "DATABASE",
      "Expression": [{"
```

```

        "TagKey": "usergroup",
        "TagValues": [
            "analyst",
            "developer"
        ]
    }
}
    ],
    "Permissions": [
        "ALL"
    ],
    "PermissionsWithGrantOption": [
        "ALL"
    ]
}

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

Contoh 2: Untuk memberikan izin tingkat kolom kepada kepala sekolah

`grant-permissions` Contoh berikut memberikan izin untuk memilih kolom tertentu untuk kepala sekolah.

```

aws lakeformation grant-permissions \
  --cli-input-json file://input.json

```

Isi dari `input.json`:

```

{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
  "Resource": {
    "TableWithColumns": {
      "CatalogId": "123456789111",
      "ColumnNames": ["p_end_date_sk"],
      "DatabaseName": "tpc",
      "Name": "dl_tpc_promotion"
    }
  }
}

```

```
    }
  },
  "Permissions": [
    "SELECT"
  ],
  "PermissionsWithGrantOption": []
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

Contoh 3: Untuk memberikan izin tabel kepada kepala sekolah

`grant-permissions` Contoh berikut memberikan izin pilih pada semua tabel database yang diberikan kepada kepala sekolah.

```
aws lakeformation grant-permissions \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "TableWildcard": {}
    }
  },
  "Permissions": [
    "SELECT"
  ],
  "PermissionsWithGrantOption": []
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

Contoh 4: Untuk memberikan izin pada LF-tag kepada kepala sekolah

`grant-permissions` Contoh berikut memberikan izin asosiasi pada LF-tag kepada kepala sekolah.

```
aws lakeformation grant-permissions \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Principal": {  
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"  
  },  
  "Resource": {  
    "LFTag": {  
      "CatalogId": "123456789111",  
      "TagKey": "category",  
      "TagValues": [  
        "private", "public"  
      ]  
    }  
  },  
  "Permissions": [  
    "ASSOCIATE"  
  ],  
  "PermissionsWithGrantOption": []  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

Contoh 5: Untuk memberikan izin pada lokasi data kepada kepala sekolah

`grant-permissions` Contoh berikut memberikan izin pada lokasi data kepada kepala sekolah.

```
aws lakeformation grant-permissions \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Principal": {  
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"  
  },  
  "Resource": {  
    "DataLocation": {  
      "CatalogId": "123456789111",  
      "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111"  
    }  
  },  
  "Permissions": [  
    "DATA_LOCATION_ACCESS"  
  ],  
  "PermissionsWithGrantOption": []  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [GrantPermissions](#) di Referensi AWS CLI Perintah.

## **list-data-cells-filter**

Contoh kode berikut menunjukkan cara menggunakan `list-data-cells-filter`.

### AWS CLI

Untuk membuat daftar filter sel data

`list-data-cells-filter` Contoh berikut daftar filter sel data untuk tabel yang diberikan.

```
aws lakeformation list-data-cells-filter \  
  --cli-input-json file://input.json
```

## Isi dari `input.json`:

```
{
  "MaxResults": 2,
  "Table": {
    "CatalogId": "123456789111",
    "DatabaseName": "tpc",
    "Name": "dl_tpc_promotion"
  }
}
```

## Output:

```
{
  "DataCellsFilters": [{
    "TableCatalogId": "123456789111",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_promotion",
    "Name": "developer_promotion",
    "RowFilter": {
      "FilterExpression": "p_promo_name='ese'"
    }
  },
  "ColumnNames": [
    "p_channel_details",
    "p_start_date_sk",
    "p_purpose",
    "p_promo_id",
    "p_promo_name",
    "p_end_date_sk",
    "p_discount_active"
  ]
},
{
  "TableCatalogId": "123456789111",
  "DatabaseName": "tpc",
  "TableName": "dl_tpc_promotion",
  "Name": "developer_promotion_allrows",
  "RowFilter": {
    "FilterExpression": "TRUE",
    "AllRowsWildcard": {}
  },
  "ColumnNames": [
    "p_channel_details",
```

```

        "p_start_date_sk",
        "p_promo_name"
    ]
}
],
"NextToken": "2MDA2MTgwNiwibmFub3MiOjE0MDAwMDAwMH19"
}

```

Untuk informasi selengkapnya, lihat [Pemfilteran data dan keamanan tingkat sel di Lake Formation di Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [ListDataCellsFilter](#) di Referensi AWS CLI Perintah.

## list-permissions

Contoh kode berikut menunjukkan cara menggunakan `list-permissions`.

### AWS CLI

Contoh 1: Untuk mengambil daftar izin utama pada sumber daya

`list-permissions` Contoh berikut mengembalikan daftar izin utama pada sumber daya database.

```
aws lakeformation list-permissions \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "ResourceType": "DATABASE",
  "MaxResults": 2
}
```

Output:

```
{
  "PrincipalResourcePermissions": [{
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
campaign-manager"
    }
  }
]
```



```

    },
    "Resource": {
      "Database": {
        "CatalogId": "123456789111",
        "Name": "tpc"
      }
    },
    "Permissions": [
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": []
  ]],
  "NextToken":
  "E5S1JDSTZ1eUp6SWpvaU9UQTN0RE0zTXpFeE5Ua3pJbjE5TENKbGVIQnBjbUYwYVc5dUlqcDdJbk5sWTI5dVpITW1P
}

```

Untuk informasi selengkapnya, lihat [Mengelola izin Lake Formation](#) di Panduan Pengembang AWS Lake Formation.

Contoh 2: Untuk mengambil daftar izin utama pada tabel dengan filter data

`list-permissions` Contoh berikut mencantumkan izin pada tabel dengan filter data terkait yang diberikan kepada prinsipal.

```

aws lakeformation list-permissions \
  --cli-input-json file://input.json

```

Isi dari `input.json`:

```

{
  "CatalogId": "123456789111",
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_customer"
    }
  },
  "IncludeRelated": "TRUE",
  "MaxResults": 10
}

```

## Output:

```
{
  "PrincipalResourcePermissions": [{
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "customer",
        "Name": "customer_invoice"
      }
    },
    "Permissions": [
      "ALL",
      "ALTER",
      "DELETE",
      "DESCRIBE",
      "DROP",
      "INSERT"
    ],
    "PermissionsWithGrantOption": [
      "ALL",
      "ALTER",
      "DELETE",
      "DESCRIBE",
      "DROP",
      "INSERT"
    ]
  }],
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
    },
    "Resource": {
      "TableWithColumns": {
        "CatalogId": "123456789111",
        "DatabaseName": "customer",
        "Name": "customer_invoice",
        "ColumnWildcard": {}
      }
    }
  }
}
```

```

    },
    "Permissions": [
        "SELECT"
    ],
    "PermissionsWithGrantOption": [
        "SELECT"
    ]
},
{
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
  },
  "Resource": {
    "DataCellsFilter": {
      "TableCatalogId": "123456789111",
      "DatabaseName": "customer",
      "TableName": "customer_invoice",
      "Name": "dl_us_customer"
    }
  },
  "Permissions": [
    "DESCRIBE",
    "SELECT",
    "DROP"
  ],
  "PermissionsWithGrantOption": []
}
],
"NextToken": "VyeUFjY291bnRQZXJtaXNzaW9ucyI6ZmFsc2V9"
}

```

Untuk informasi selengkapnya, lihat [Mengelola izin Lake Formation](#) di Panduan Pengembang AWS Lake Formation.

Contoh 3: Untuk mengambil daftar izin utama pada LF-tag

`list-permissions` Contoh berikut mencantumkan izin pada LF-tag yang diberikan kepada prinsipal.

```

aws lakeformation list-permissions \
  --cli-input-json file://input.json

```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Resource": {
    "LFTag": {
      "CatalogId": "123456789111",
      "TagKey": "category",
      "TagValues": [
        "private"
      ]
    }
  },
  "MaxResults": 10
}
```

Output:

```
{
  "PrincipalResourcePermissions": [{
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"
    },
    "Resource": {
      "LFTag": {
        "CatalogId": "123456789111",
        "TagKey": "category",
        "TagValues": [
          "*"
        ]
      }
    },
    "Permissions": [
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": [
      "DESCRIBE"
    ]
  },
  {
    "Principal": {
```

```

        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
      },
      "Resource": {
        "LFTag": {
          "CatalogId": "123456789111",
          "TagKey": "category",
          "TagValues": [
            "*"
          ]
        }
      },
      "Permissions": [
        "ASSOCIATE"
      ],
      "PermissionsWithGrantOption": [
        "ASSOCIATE"
      ]
    }
  ],
  "NextToken": "EJwY21GMGFxOxVJanA3SW50cm1pc3Npb25zIjpmYWxzZX0="
}

```

Untuk informasi selengkapnya, lihat [Mengelola izin Lake Formation](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [ListPermissions](#) di Referensi AWS CLI Perintah.

## list-resources

Contoh kode berikut menunjukkan cara menggunakan `list-resources`.

### AWS CLI

Untuk daftar sumber daya yang dikelola oleh Lake Formation

`list-resources` Contoh berikut mencantumkan sumber daya yang cocok dengan kondisi yang dikelola oleh Lake Formation.

```
aws lakeformation list-resources \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "FilterConditionList": [{
    "Field": "ROLE_ARN",
    "ComparisonOperator": "CONTAINS",
    "StringValueList": [
      "123456789111"
    ]
  }],
  "MaxResults": 10
}
```

Output:

```
{
  "ResourceInfoList": [{
    "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111",
    "RoleArn": "arn:aws:iam::123456789111:role/LF-GlueServiceRole",
    "LastModified": "2022-07-21T02:12:46.669000+00:00"
  },
  {
    "ResourceArn": "arn:aws:s3:::lf-emr-test-123456789111",
    "RoleArn": "arn:aws:iam::123456789111:role/EMRIFS3Role",
    "LastModified": "2022-07-29T16:22:03.211000+00:00"
  }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola izin Lake Formation](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [ListResources](#) di Referensi AWS CLI Perintah.

## list-transactions

Contoh kode berikut menunjukkan cara menggunakan `list-transactions`.

### AWS CLI

Untuk mencantumkan semua detail transaksi

`list-transactions` Contoh berikut mengembalikan metadata tentang transaksi dan statusnya.

```
aws lakeformation list-transactions \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "StatusFilter": "ALL",  
  "MaxResults": 3  
}
```

Output:

```
{  
  "Transactions": [{  
    "TransactionId": "1234569f08804cb790d950d4d0fe485e",  
    "TransactionStatus": "committed",  
    "TransactionStartTime": "2022-08-10T14:32:29.220000+00:00",  
    "TransactionEndTime": "2022-08-10T14:32:33.751000+00:00"  
  },  
  {  
    "TransactionId": "12345972ca8347b89825e33c5774aec4",  
    "TransactionStatus": "committed",  
    "TransactionStartTime": "2022-08-10T14:29:04.046000+00:00",  
    "TransactionEndTime": "2022-08-10T14:29:09.681000+00:00"  
  },  
  {  
    "TransactionId": "12345daf6cb047dbba8ad9b0414613b2",  
    "TransactionStatus": "committed",  
    "TransactionStartTime": "2022-08-10T13:56:51.261000+00:00",  
    "TransactionEndTime": "2022-08-10T13:56:51.547000+00:00"  
  }  
  ],  
  "NextToken": "77X1ebypsI7os+X21hHsZLGNC DK3nNGpwRdFpicS0Hg cX1/  
QMoniUAKcpR3kj3ts3PVdMA=="  
}
```

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [ListTransactions](#) di Referensi AWS CLI Perintah.

## put-data-lake-settings

Contoh kode berikut menunjukkan cara menggunakan `put-data-lake-settings`.

### AWS CLI

Untuk mengatur pengaturan AWS danau data yang dikelola Lake Formation

`put-data-lake-settings` Contoh berikut menetapkan daftar administrator data lake dan pengaturan data lake lainnya.

```
aws lakeformation put-data-lake-settings \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "DataLakeSettings": {  
    "DataLakeAdmins": [{  
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-  
admin"  
    }  
  ],  
  "CreateDatabaseDefaultPermissions": [],  
  "CreateTableDefaultPermissions": [],  
  "TrustedResourceOwners": [],  
  "AllowExternalDataFiltering": true,  
  "ExternalDataFilteringAllowList": [{  
    "DataLakePrincipalIdentifier": "123456789111"  
  }],  
  "AuthorizedSessionTagValueList": ["Amazon EMR"]  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengubah setelan keamanan default untuk data lake Anda](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [PutDataLakeSettings](#) di Referensi AWS CLI Perintah.



## register-resource

Contoh kode berikut menunjukkan cara menggunakan `register-resource`.

### AWS CLI

Contoh 1: Untuk mendaftarkan penyimpanan data lake menggunakan Service Linked Role

`register-resource` Contoh berikut mendaftarkan sumber daya yang dikelola oleh Lake Formation menggunakan peran terkait Layanan.

```
aws lakeformation register-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123",  
  "UseServiceLinkedRole": true  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan lokasi Amazon S3 ke danau data Anda di Panduan](#) Pengembang AWS Lake Formation.

Contoh 2: Untuk mendaftarkan penyimpanan data lake menggunakan peran khusus

`register-resource` Contoh berikut mendaftarkan sumber daya yang dikelola oleh Lake Formation menggunakan peran kustom.

```
aws lakeformation register-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123",  
  "UseServiceLinkedRole": false,  
  "RoleArn": "arn:aws:iam::123456789111:role/LF-GlueServiceRole"  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan lokasi Amazon S3 ke danau data Anda di Panduan](#) Pengembang AWS Lake Formation.

- Untuk detail API, lihat [RegisterResource](#) di Referensi AWS CLI Perintah.

## **remove-lf-tags-from-resource**

Contoh kode berikut menunjukkan cara menggunakan `remove-lf-tags-from-resource`.

### AWS CLI

Untuk menghapus LF-tag dari sumber daya

`remove-lf-tags-from-resource` Contoh berikut menghapus asosiasi LF-tag dengan sumber daya tabel.

```
aws lakeformation remove-lf-tags-from-resource \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Resource": {  
    "Table": {  
      "CatalogId": "123456789111",  
      "DatabaseName": "tpc",  
      "Name": "dl_tpc_promotion"  
    }  
  },  
  "LFTags": [{  
    "CatalogId": "123456789111",  
    "TagKey": "usergroup",  
    "TagValues": [  
      "developer"  
    ]  
  }]  
}
```

Output:

```
{
  "Failures": []
}
```

Untuk informasi selengkapnya, lihat [Menetapkan LF-tag ke sumber daya Katalog Data di Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [RemoveLfTagsFromResource](#) di Referensi AWS CLI Perintah.

## revoke-permissions

Contoh kode berikut menunjukkan cara menggunakan `revoke-permissions`.

### AWS CLI

Untuk mencabut izin sumber daya dari prinsipal

`revoke-permissions` Contoh berikut mencabut akses utama ke tabel tertentu dari database yang diberikan.

```
aws lakeformation revoke-permissions \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_promotion"
    }
  },
  "Permissions": [
    "ALL"
  ],
  "PermissionsWithGrantOption": []
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memberikan dan mencabut izin pada sumber daya Katalog Data di](#) Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [RevokePermissions](#) di Referensi AWS CLI Perintah.

## search-databases-by-lf-tags

Contoh kode berikut menunjukkan cara menggunakan `search-databases-by-lf-tags`.

### AWS CLI

Untuk mencari sumber daya database dengan LFTags

`search-databases-by-lf-tags` Contoh berikut mencari sumber daya database yang cocok ekspresi LFTag.

```
aws lakeformation search-databases-by-lf-tags \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "MaxResults": 1,  
  "CatalogId": "123456789111",  
  "Expression": [{  
    "TagKey": "usergroup",  
    "TagValues": [  
      "developer"  
    ]  
  }]  
}
```

Output:

```
{  
  "DatabaseList": [{  
    "Database": {  
      "CatalogId": "123456789111",  
      "Name": "tpc"  
    },  
  },  
}
```

```
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat sumber daya yang ditetapkan LF-tag dalam Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [SearchDatabasesByLfTags](#) di Referensi AWS CLI Perintah.

## search-tables-by-lf-tags

Contoh kode berikut menunjukkan cara menggunakan `search-tables-by-lf-tags`.

### AWS CLI

Untuk mencari sumber daya tabel dengan LFTags

`search-tables-by-lf-tags` Contoh berikut mencari sumber daya tabel yang cocok ekspresi LFTag.

```
aws lakeformation search-tables-by-lf-tags \
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{
  "MaxResults": 2,
  "CatalogId": "123456789111",
  "Expression": [{
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
}
```

## Output:

```
{
  "NextToken": "c2VhcmNoQWxsVGFnc0luVGFibGVzIjpmYWxzZX0=",
  "TableList": [{
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_item"
    },
    "LFTagOnDatabase": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }
  ]},
  "LFTagsOnTable": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }
  ],
  "LFTagsOnColumns": [{
    "Name": "i_item_desc",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }
  ]
  },
  {
    "Name": "i_container",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }
  ]
  },
}
```

```
{
  "Name": "i_wholesale_cost",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_manufact_id",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_brand_id",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_formulation",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_current_price",
  "LFTags": [{
    "CatalogId": "123456789111",
```

```
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }
},
{
    "Name": "i_size",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
},
{
    "Name": "i_rec_start_date",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
},
{
    "Name": "i_manufact",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
},
{
    "Name": "i_item_sk",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
}
```



```
    ]],  
  },  
  {  
    "Name": "i_manager_id",  
    "LFTags": [{  
      "CatalogId": "123456789111",  
      "TagKey": "usergroup",  
      "TagValues": [  
        "developer"  
      ]  
    }]  
  },  
  {  
    "Name": "i_item_id",  
    "LFTags": [{  
      "CatalogId": "123456789111",  
      "TagKey": "usergroup",  
      "TagValues": [  
        "developer"  
      ]  
    }]  
  },  
  {  
    "Name": "i_class_id",  
    "LFTags": [{  
      "CatalogId": "123456789111",  
      "TagKey": "usergroup",  
      "TagValues": [  
        "developer"  
      ]  
    }]  
  },  
  {  
    "Name": "i_class",  
    "LFTags": [{  
      "CatalogId": "123456789111",  
      "TagKey": "usergroup",  
      "TagValues": [  
        "developer"  
      ]  
    }]  
  },  
  {  
    "Name": "i_category",
```

```
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_category_id",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_brand",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_units",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_rec_end_date",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
```

```
        "developer"
      ]
    ]
  },
  {
    "Name": "i_color",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_product_name",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  }
]
}]
}
```

Untuk informasi selengkapnya, lihat [Melihat sumber daya yang ditetapkan LF-tag dalam Panduan Pengembang AWS Lake Formation](#).

- Untuk detail API, lihat [SearchTablesByLfTags](#) di Referensi AWS CLI Perintah.

## start-query-planning

Contoh kode berikut menunjukkan cara menggunakan `start-query-planning`.

### AWS CLI

Untuk memproses pernyataan kueri

`start-query-planning` Contoh berikut mengirimkan permintaan untuk memproses pernyataan query.

```
aws lakeformation start-query-planning \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "QueryPlanningContext": {  
    "CatalogId": "012345678901",  
    "DatabaseName": "tpc"  
  },  
  "QueryString": "select * from dl_tpc_household_demographics_gov where  
hd_income_band_sk=9"  
}
```

Output:

```
{  
  "QueryId": "772a273f-4a62-4cda-8d98-69615ee8be9b"  
}
```

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [StartQueryPlanning](#) di Referensi AWS CLI Perintah.

## start-transaction

Contoh kode berikut menunjukkan cara menggunakan `start-transaction`.

### AWS CLI

Untuk memulai transaksi baru

`start-transaction` Contoh berikut memulai transaksi baru dan mengembalikan ID transaksinya.

```
aws lakeformation start-transaction \  
  --transaction-type = 'READ_AND_WRITE'
```

Output:

```
{
  "TransactionId": "b014d972ca8347b89825e33c5774aec4"
}
```

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [StartTransaction](#) di Referensi AWS CLI Perintah.

## update-lf-tag

Contoh kode berikut menunjukkan cara menggunakan `update-lf-tag`.

### AWS CLI

Untuk memperbarui definisi LF-tag

`update-lf-tag` Contoh berikut memperbarui definisi LF-tag.

```
aws lakeformation update-lf-tag \
  --catalog-id '123456789111' \
  --tag-key 'usergroup' \
  --tag-values-to-add ['"admin"]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola LF-tag untuk kontrol akses metadata](#) di Panduan Pengembang AWS Lake Formation.

- Untuk detail API, lihat [UpdateLfTag](#) di Referensi AWS CLI Perintah.

## update-table-objects

Contoh kode berikut menunjukkan cara menggunakan `update-table-objects`.

### AWS CLI

Untuk memodifikasi objek dari tabel yang diatur

`update-table-objects` Contoh berikut menambahkan objek S3 yang disediakan ke tabel diatur yang ditentukan.

```
aws lakeformation update-table-objects \  
  --cli-input-json file://input.json
```

Isi dari `input.json`:

```
{  
  "CatalogId": "012345678901",  
  "DatabaseName": "tpc",  
  "TableName": "dl_tpc_household_demographics_gov",  
  "TransactionId": "12347a9f75424b9b915f6ff201d2a190",  
  "WriteOperations": [{  
    "AddObject": {  
      "Uri": "s3://lf-data-lake-012345678901/target/  
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-  
ff26b17504414fe88b302cd795eabd00.parquet",  
      "ETag": "1234ab1fc50a316b149b4e1f21a73800",  
      "Size": 42200  
    }  
  }  
}]  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi lebih lanjut, lihat [Membaca dari dan menulis ke data lake dalam transaksi](#) di AWS Lake Formation Developer Guide.

- Untuk detail API, lihat [UpdateTableObjects](#) di Referensi AWS CLI Perintah.

## Contoh Lambda menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Lambda AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **add-layer-version-permission**

Contoh kode berikut menunjukkan cara menggunakan `add-layer-version-permission`.

## AWS CLI

Untuk menambahkan izin ke versi lapisan

`add-layer-version-permission` Contoh berikut memberikan izin untuk akun yang ditentukan untuk menggunakan versi 1 dari lapisan `my-layer`.

```
aws lambda add-layer-version-permission \  
  --layer-name my-layer \  
  --statement-id xaccount \  
  --action lambda:GetLayerVersion \  
  --principal 123456789012 \  
  --version-number 1
```

## Output:

```
{  
  "RevisionId": "35d87451-f796-4a3f-a618-95a3671b0a0c",  
  "Statement":  
  {  
    "Sid": "xaccount",  
    "Effect": "Allow",  
    "Principal": {  
      "AWS": "arn:aws:iam::210987654321:root"  
    },  
    "Action": "lambda:GetLayerVersion",  
    "Resource": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer:1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [AddLayerVersionPermission](#) di Referensi AWS CLI Perintah.

## add-permission

Contoh kode berikut menunjukkan cara menggunakan `add-permission`.

### AWS CLI

Untuk menambahkan izin ke fungsi Lambda yang ada

`add-permission` Contoh berikut memberikan izin layanan Amazon SNS untuk memanggil fungsi bernama `my-function`

```
aws lambda add-permission \  
  --function-name my-function \  
  --action lambda:InvokeFunction \  
  --statement-id sns \  
  --principal sns.amazonaws.com
```

Output:

```
{  
  "Statement":  
  {  
    "Sid": "sns",  
    "Effect": "Allow",  
    "Principal": {  
      "Service": "sns.amazonaws.com"  
    },  
    "Action": "lambda:InvokeFunction",  
    "Resource": "arn:aws:lambda:us-east-2:123456789012:function:my-function"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Kebijakan Berbasis Sumber Daya untuk Lambda AWS di Panduan Pengembang Lambda](#).AWS

- Untuk detail API, lihat [AddPermission](#) di Referensi AWS CLI Perintah.

## create-alias

Contoh kode berikut menunjukkan cara menggunakan `create-alias`.



## AWS CLI

Untuk membuat alias untuk fungsi Lambda

`create-alias` Contoh berikut membuat alias bernama LIVE yang menunjuk ke versi 1 dari fungsi `my-function` Lambda.

```
aws lambda create-alias \  
  --function-name my-function \  
  --description "alias for live version of function" \  
  --function-version 1 \  
  --name LIVE
```

Output:

```
{  
  "FunctionVersion": "1",  
  "Name": "LIVE",  
  "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",  
  "RevisionId": "873282ed-4cd3-4dc8-a069-d0c647e470c6",  
  "Description": "alias for live version of function"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [CreateAlias](#) di Referensi AWS CLI Perintah.

## `create-event-source-mapping`

Contoh kode berikut menunjukkan cara menggunakan `create-event-source-mapping`.

## AWS CLI

Untuk membuat pemetaan antara sumber peristiwa dan fungsi AWS Lambda

`create-event-source-mapping` Contoh berikut membuat pemetaan antara antrian SQS dan fungsi `my-function` Lambda.

```
aws lambda create-event-source-mapping \  
  --function-name my-function \  
  --
```

```
--batch-size 5 \  
--event-source-arn arn:aws:sqs:us-west-2:123456789012:mySQSqueue
```

### Output:

```
{  
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "StateTransitionReason": "USER_INITIATED",  
  "LastModified": 1569284520.333,  
  "BatchSize": 5,  
  "State": "Creating",  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"  
}
```

Untuk informasi selengkapnya, lihat [Pemetaan Sumber Peristiwa AWS Lambda di Panduan Pengembang](#) Lambda AWS .

- Untuk detail API, lihat [CreateEventSourceMapping](#) di Referensi AWS CLI Perintah.

## create-function

Contoh kode berikut menunjukkan cara menggunakan `create-function`.

### AWS CLI

Untuk membuat fungsi Lambda

`create-function` Contoh berikut menciptakan fungsi Lambda bernama `my-function`

```
aws lambda create-function \  
  --function-name my-function \  
  --runtime nodejs18.x \  
  --zip-file fileb://my-function.zip \  
  --handler my-function.handler \  
  --role arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-tges6bf4
```

Isi dari `my-function.zip`:

```
This file is a deployment package that contains your function code and any  
dependencies.
```

## Output:

```
{
  "TracingConfig": {
    "Mode": "PassThrough"
  },
  "CodeSha256": "PFn4S+er27qk+UuZSTKEQfNKG/XNn7QJs90mJgq6oH8=",
  "FunctionName": "my-function",
  "CodeSize": 308,
  "RevisionId": "873282ed-4cd3-4dc8-a069-d0c647e470c6",
  "MemorySize": 128,
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
  "Version": "$LATEST",
  "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-zgur6bf4",
  "Timeout": 3,
  "LastModified": "2023-10-14T22:26:11.234+0000",
  "Handler": "my-function.handler",
  "Runtime": "nodejs18.x",
  "Description": ""
}
```

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [CreateFunction](#) di Referensi AWS CLI Perintah.

## delete-alias

Contoh kode berikut menunjukkan cara menggunakan `delete-alias`.

### AWS CLI

Untuk menghapus alias fungsi Lambda

`delete-alias` Contoh berikut menghapus alias bernama LIVE dari fungsi Lambda `my-function`.

```
aws lambda delete-alias \
  --function-name my-function \
  --name LIVE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [DeleteAlias](#) di Referensi AWS CLI Perintah.

## **delete-event-source-mapping**

Contoh kode berikut menunjukkan cara menggunakan `delete-event-source-mapping`.

### AWS CLI

Untuk menghapus pemetaan antara sumber peristiwa dan fungsi AWS Lambda

`delete-event-source-mapping` Contoh berikut menghapus pemetaan antara antrian SQS dan fungsi Lambda. `my-function`

```
aws lambda delete-event-source-mapping \  
  --uuid a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{  
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "StateTransitionReason": "USER_INITIATED",  
  "LastModified": 1569285870.271,  
  "BatchSize": 5,  
  "State": "Deleting",  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"  
}
```

Untuk informasi selengkapnya, lihat [Pemetaan Sumber Peristiwa AWS Lambda di Panduan Pengembang](#) Lambda AWS .

- Untuk detail API, lihat [DeleteEventSourceMapping](#) di Referensi AWS CLI Perintah.

## **delete-function-concurrency**

Contoh kode berikut menunjukkan cara menggunakan `delete-function-concurrency`.

## AWS CLI

Untuk menghapus batas eksekusi bersamaan yang dicadangkan dari fungsi

`delete-function-concurrency` Contoh berikut menghapus batas eksekusi bersamaan yang dicadangkan dari fungsi. `my-function`

```
aws lambda delete-function-concurrency \  
  --function-name my-function
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Reservasi Konkurensi untuk Fungsi Lambda di](#) Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [DeleteFunctionConcurrency](#) di Referensi AWS CLI Perintah.

## `delete-function-event-invoke-config`

Contoh kode berikut menunjukkan cara menggunakan `delete-function-event-invoke-config`.

## AWS CLI

Untuk menghapus konfigurasi pemanggilan asinkron

`delete-function-event-invoke-config` Contoh berikut menghapus konfigurasi pemanggilan asinkron untuk alias fungsi yang GREEN ditentukan.

```
aws lambda delete-function-event-invoke-config --function-name my-function:GREEN
```

- Untuk detail API, lihat [DeleteFunctionEventInvokeConfig](#) di Referensi AWS CLI Perintah.

## `delete-function`

Contoh kode berikut menunjukkan cara menggunakan `delete-function`.

## AWS CLI

Contoh 1: Untuk menghapus fungsi Lambda dengan nama fungsi

`delete-function` Contoh berikut menghapus fungsi Lambda `my-function` bernama dengan menentukan nama fungsi.

```
aws lambda delete-function \  
  --function-name my-function
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menghapus fungsi Lambda dengan fungsi ARN

`delete-function` Contoh berikut menghapus fungsi Lambda `my-function` bernama dengan menentukan ARN fungsi.

```
aws lambda delete-function \  
  --function-name arn:aws:lambda:us-west-2:123456789012:function:my-function
```

Perintah ini tidak menghasilkan output.

Contoh 3: Untuk menghapus fungsi Lambda dengan fungsi sebagian ARN

`delete-function` Contoh berikut menghapus fungsi Lambda `my-function` bernama dengan menentukan ARN parsial fungsi.

```
aws lambda delete-function \  
  --function-name 123456789012:function:my-function
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [DeleteFunction](#) di Referensi AWS CLI Perintah.

## **delete-layer-version**

Contoh kode berikut menunjukkan cara menggunakan `delete-layer-version`.

### AWS CLI

Untuk menghapus versi layer Lambda

`delete-layer-version` Contoh berikut menghapus versi 2 dari layer bernama `my-layer`.

```
aws lambda delete-layer-version \  
  --layer-name my-layer \  
  --version-number 2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [DeleteLayerVersion](#) di Referensi AWS CLI Perintah.

## **delete-provisioned-concurrency-config**

Contoh kode berikut menunjukkan cara menggunakan `delete-provisioned-concurrency-config`.

### AWS CLI

Untuk menghapus konfigurasi konkurensi yang disediakan

`delete-provisioned-concurrency-config` Contoh berikut menghapus konfigurasi konkurensi yang disediakan untuk GREEN alias fungsi yang ditentukan.

```
aws lambda delete-provisioned-concurrency-config \  
  --function-name my-function \  
  --qualifier GREEN
```

- Untuk detail API, lihat [DeleteProvisionedConcurrencyConfig](#) di Referensi AWS CLI Perintah.

## **get-account-settings**

Contoh kode berikut menunjukkan cara menggunakan `get-account-settings`.

### AWS CLI

Untuk mengambil detail tentang akun Anda di suatu Wilayah AWS

`get-account-settings` Contoh berikut menampilkan batas Lambda dan informasi penggunaan untuk akun Anda.

```
aws lambda get-account-settings
```

**Output:**

```
{
  "AccountLimit": {
    "CodeSizeUnzipped": 262144000,
    "UnreservedConcurrentExecutions": 1000,
    "ConcurrentExecutions": 1000,
    "CodeSizeZipped": 52428800,
    "TotalCodeSize": 80530636800
  },
  "AccountUsage": {
    "FunctionCount": 4,
    "TotalCodeSize": 9426
  }
}
```

Untuk informasi selengkapnya, lihat [Batas AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [GetAccountSettings](#) di Referensi AWS CLI Perintah.

**get-alias**

Contoh kode berikut menunjukkan cara menggunakan `get-alias`.

**AWS CLI**

Untuk mengambil rincian tentang alias fungsi

`get-alias` Contoh berikut menampilkan rincian untuk alias bernama LIVE pada fungsi `my-function` Lambda.

```
aws lambda get-alias \
  --function-name my-function \
  --name LIVE
```

**Output:**

```
{
  "FunctionVersion": "3",
  "Name": "LIVE",
  "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
  "RevisionId": "594f41fb-b85f-4c20-95c7-6ca5f2a92c93",
```



```
"Description": "alias for live version of function"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [GetAlias](#) di Referensi AWS CLI Perintah.

## get-event-source-mapping

Contoh kode berikut menunjukkan cara menggunakan `get-event-source-mapping`.

### AWS CLI

Untuk mengambil detail tentang pemetaan sumber peristiwa

`get-event-source-mapping` Contoh berikut menampilkan rincian untuk pemetaan antara antrian SQS dan fungsi Lambda `my-function`.

```
aws lambda get-event-source-mapping \
  --uuid "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Output:

```
{
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "StateTransitionReason": "USER_INITIATED",
  "LastModified": 1569284520.333,
  "BatchSize": 5,
  "State": "Enabled",
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Untuk informasi selengkapnya, lihat [Pemetaan Sumber Peristiwa AWS Lambda di Panduan Pengembang](#) Lambda AWS .

- Untuk detail API, lihat [GetEventSourceMapping](#) di Referensi AWS CLI Perintah.

## get-function-concurrency

Contoh kode berikut menunjukkan cara menggunakan `get-function-concurrency`.

## AWS CLI

Untuk melihat setelan konkurensi cadangan untuk suatu fungsi

`get-function-concurrency` Contoh berikut mengambil pengaturan konkurensi cadangan untuk fungsi yang ditentukan.

```
aws lambda get-function-concurrency \  
  --function-name my-function
```

Output:

```
{  
  "ReservedConcurrentExecutions": 250  
}
```

- Untuk detail API, lihat [GetFunctionConcurrency](#) di Referensi AWS CLI Perintah.

## `get-function-configuration`

Contoh kode berikut menunjukkan cara menggunakan `get-function-configuration`.

## AWS CLI

Untuk mengambil pengaturan khusus versi dari fungsi Lambda

`get-function-configuration` Contoh berikut menampilkan pengaturan untuk versi 2 dari `my-function` fungsi.

```
aws lambda get-function-configuration \  
  --function-name my-function:2
```

Output:

```
{  
  "FunctionName": "my-function",  
  "LastModified": "2019-09-26T20:28:40.438+0000",  
  "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",  
  "MemorySize": 256,  
  "Version": "2",  
  "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy3l9yqq",
```

```
"Timeout": 3,
"Runtime": "nodejs10.x",
"TracingConfig": {
  "Mode": "PassThrough"
},
"CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJmlKidWoaCgk=",
"Description": "",
"VpcConfig": {
  "SubnetIds": [],
  "VpcId": "",
  "SecurityGroupIds": []
},
"CodeSize": 304,
"FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:2",
"Handler": "index.handler"
}
```

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [GetFunctionConfiguration](#) di Referensi AWS CLI Perintah.

## get-function-event-invoke-config

Contoh kode berikut menunjukkan cara menggunakan `get-function-event-invoke-config`.

### AWS CLI

Untuk melihat konfigurasi pemanggilan asinkron

`get-function-event-invoke-config` Contoh berikut mengambil konfigurasi pemanggilan asinkron untuk alias fungsi yang BLUE ditentukan.

```
aws lambda get-function-event-invoke-config \
  --function-name my-function:BLUE
```

### Output:

```
{
  "LastModified": 1577824396.653,
  "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
```

```

    "MaximumRetryAttempts": 0,
    "MaximumEventAgeInSeconds": 3600,
    "DestinationConfig": {
      "OnSuccess": {},
      "OnFailure": {
        "Destination": "arn:aws:sqs:us-east-2:123456789012:failed-invocations"
      }
    }
  }
}

```

- Untuk detail API, lihat [GetFunctionEventInvokeConfig](#) di Referensi AWS CLI Perintah.

## get-function

Contoh kode berikut menunjukkan cara menggunakan `get-function`.

### AWS CLI

Untuk mengambil informasi tentang suatu fungsi

`get-function` Contoh berikut menampilkan informasi tentang `my-function` fungsi.

```

aws lambda get-function \
  --function-name my-function

```

Output:

```

{
  "Concurrency": {
    "ReservedConcurrentExecutions": 100
  },
  "Code": {
    "RepositoryType": "S3",
    "Location": "https://awslambda-us-west-2-tasks.s3.us-west-2.amazonaws.com/snapshots/123456789012/my-function..."
  },
  "Configuration": {
    "TracingConfig": {
      "Mode": "PassThrough"
    },
    "Version": "$LATEST",
    "CodeSha256": "5tT2qgzYUHoqwR616pZ2dpkn/0J1FrzJm1KidWaaCgk=",

```

```

    "FunctionName": "my-function",
    "VpcConfig": {
      "SubnetIds": [],
      "VpcId": "",
      "SecurityGroupIds": []
    },
    "MemorySize": 128,
    "RevisionId": "28f0fb31-5c5c-43d3-8955-03e76c5c1075",
    "CodeSize": 304,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-role-uy3l9qq",
    "Timeout": 3,
    "LastModified": "2019-09-24T18:20:35.054+0000",
    "Runtime": "nodejs10.x",
    "Description": ""
  }
}

```

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [GetFunction](#) di Referensi AWS CLI Perintah.

## get-layer-version-by-arn

Contoh kode berikut menunjukkan cara menggunakan `get-layer-version-by-arn`.

### AWS CLI

Untuk mengambil informasi tentang versi lapisan Lambda

`get-layer-version-by-arn` Contoh berikut menampilkan informasi tentang versi layer dengan Amazon Resource Name (ARN) yang ditentukan.

```

aws lambda get-layer-version-by-arn \
  --arn "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-SciPy1x:2"

```

Output:

```
{
```

```
"LayerVersionArn": "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-
Python311-SciPy1x:2",
  "Description": "AWS Lambda SciPy layer for Python 3.11 (scipy-1.1.0,
numpy-1.15.4) https://github.com/scipy/scipy/releases/tag/v1.1.0 https://
github.com/numpy/numpy/releases/tag/v1.15.4",
  "CreateDate": "2023-10-12T10:09:38.398+0000",
  "LayerArn": "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-
SciPy1x",
  "Content": {
    "CodeSize": 41784542,
    "CodeSha256": "GGmv8ocUw4c1y0T8HL0Vx/f5V4RmSCGNjDIslY4VskM=",
    "Location": "https://awslambda-us-west-2-layers.s3.us-west-2.amazonaws.com/
snapshots/123456789012/..."
  },
  "Version": 2,
  "CompatibleRuntimes": [
    "python3.11"
  ],
  "LicenseInfo": "SciPy: https://github.com/scipy/scipy/blob/main/LICENSE.txt,
NumPy: https://github.com/numpy/numpy/blob/main/LICENSE.txt"
}
```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [GetLayerVersionByArn](#) di Referensi AWS CLI Perintah.

## get-layer-version-policy

Contoh kode berikut menunjukkan cara menggunakan `get-layer-version-policy`.

### AWS CLI

Untuk mengambil kebijakan izin untuk versi lapisan Lambda

`get-layer-version-policy` Contoh berikut menampilkan informasi kebijakan tentang versi 1 untuk lapisan bernama `my-layer`.

```
aws lambda get-layer-version-policy \
  --layer-name my-layer \
  --version-number 1
```

Output:

```
{
  "Policy": {
    "Version": "2012-10-17",
    "Id": "default",
    "Statement": [
      {
        "Sid": "xaccount",
        "Effect": "Allow",
        "Principal": {"AWS": "arn:aws:iam::123456789012:root"},
        "Action": "lambda:GetLayerVersion",
        "Resource": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer:1"
      }
    ]
  },
  "RevisionId": "c68f21d2-cbf0-4026-90f6-1375ee465cd0"
}
```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [GetLayerVersionPolicy](#) di Referensi AWS CLI Perintah.

## get-layer-version

Contoh kode berikut menunjukkan cara menggunakan `get-layer-version`.

### AWS CLI

Untuk mengambil informasi tentang versi lapisan Lambda

`get-layer-version` Contoh berikut menampilkan informasi untuk versi 1 dari layer bernama `my-layer`.

```
aws lambda get-layer-version \
  --layer-name my-layer \
  --version-number 1
```

Output:

```
{
  "Content": {
```

```

    "Location": "https://awslambda-us-east-2-layers.s3.us-east-2.amazonaws.com/
snapshots/123456789012/my-layer-4aaa2fbb-ff77-4b0a-ad92-5b78a716a96a?
versionId=27iWyA73cCAYqyH...",
    "CodeSha256": "tv9jJ0+rPbXUUXuRKi7CwHzKtLDkDRJLB3cC3Z/ouXo=",
    "CodeSize": 169
  },
  "LayerArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer",
  "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer:1",
  "Description": "My Python layer",
  "CreateDate": "2018-11-14T23:03:52.894+0000",
  "Version": 1,
  "LicenseInfo": "MIT",
  "CompatibleRuntimes": [
    "python3.10",
    "python3.11"
  ]
}

```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [GetLayerVersion](#) di Referensi AWS CLI Perintah.

## get-policy

Contoh kode berikut menunjukkan cara menggunakan `get-policy`.

### AWS CLI

Untuk mengambil kebijakan IAM berbasis sumber daya untuk fungsi, versi, atau alias

`get-policy` Contoh berikut menampilkan informasi kebijakan tentang fungsi `my-function` Lambda.

```

aws lambda get-policy \
  --function-name my-function

```

Output:

```

{
  "Policy": {
    "Version": "2012-10-17",

```



```

    "Id": "default",
    "Statement":
    [
      {
        "Sid": "iot-events",
        "Effect": "Allow",
        "Principal": {"Service": "iotevents.amazonaws.com"},
        "Action": "lambda:InvokeFunction",
        "Resource": "arn:aws:lambda:us-west-2:123456789012:function:my-
function"
      }
    ],
    "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668"
  }

```

Untuk informasi selengkapnya, lihat [Menggunakan Kebijakan Berbasis Sumber Daya untuk Lambda AWS di Panduan Pengembang Lambda.AWS](#)

- Untuk detail API, lihat [GetPolicy](#) di Referensi AWS CLI Perintah.

## get-provisioned-concurrency-config

Contoh kode berikut menunjukkan cara menggunakan `get-provisioned-concurrency-config`.

### AWS CLI

Untuk melihat konfigurasi konkurensi yang disediakan

`get-provisioned-concurrency-config` Contoh berikut menampilkan detail untuk konfigurasi konkurensi yang disediakan untuk BLUE alias fungsi yang ditentukan.

```

aws lambda get-provisioned-concurrency-config \
  --function-name my-function \
  --qualifier BLUE

```

### Output:

```

{
  "RequestedProvisionedConcurrentExecutions": 100,
  "AvailableProvisionedConcurrentExecutions": 100,
  "AllocatedProvisionedConcurrentExecutions": 100,

```

```
"Status": "READY",
"LastModified": "2019-12-31T20:28:49+0000"
}
```

- Untuk detail API, lihat [GetProvisionedConcurrencyConfig](#) di Referensi AWS CLI Perintah.

## invoke

Contoh kode berikut menunjukkan cara menggunakan `invoke`.

### AWS CLI

Contoh 1: Untuk menjalankan fungsi Lambda secara sinkron

`invoke` Contoh berikut memanggil `my-function` fungsi sinkron. `cli-binary-format` Opsi ini diperlukan jika Anda menggunakan AWS CLI versi 2. Untuk informasi selengkapnya, lihat [opsi baris perintah global yang didukung AWS CLI](#) di Panduan Pengguna Antarmuka Baris AWS Perintah.

```
aws lambda invoke \
  --function-name my-function \
  --cli-binary-format raw-in-base64-out \
  --payload '{ "name": "Bob" }' \
  response.json
```

Output:

```
{
  "ExecutedVersion": "$LATEST",
  "StatusCode": 200
}
```

Untuk informasi selengkapnya, lihat [Pemanggilan Sinkron di Panduan Pengembang](#) Lambda AWS .

Contoh 2: Untuk menjalankan fungsi Lambda secara asinkron

`invoke` Contoh berikut memanggil `my-function` fungsi asinkron. `cli-binary-format` Opsi ini diperlukan jika Anda menggunakan AWS CLI versi 2. Untuk informasi selengkapnya, lihat [opsi baris perintah global yang didukung AWS CLI](#) di Panduan Pengguna Antarmuka Baris AWS Perintah.

```
aws lambda invoke \  
  --function-name my-function \  
  --invocation-type Event \  
  --cli-binary-format raw-in-base64-out \  
  --payload '{ "name": "Bob" }' \  
  response.json
```

Output:

```
{  
  "StatusCode": 202  
}
```

Untuk informasi selengkapnya, lihat [Pemanggilan Asinkron](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [Memanggil](#) di Referensi AWS CLI Perintah.

## list-aliases

Contoh kode berikut menunjukkan cara menggunakan `list-aliases`.

### AWS CLI

Untuk mengambil daftar alias untuk fungsi Lambda

`list-aliases` Contoh berikut menampilkan daftar alias untuk fungsi `my-function` Lambda.

```
aws lambda list-aliases \  
  --function-name my-function
```

Output:

```
{  
  "Aliases": [  
    {  
      "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-  
function:BETA",  
      "RevisionId": "a410117f-ab16-494e-8035-7e204bb7933b",  
      "FunctionVersion": "2",  
      "Name": "BETA",  
      "Description": "alias for beta version of function"  
    }  
  ]  
}
```

```
    },
    {
      "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:LIVE",
      "RevisionId": "21d40116-f8b1-40ba-9360-3ea284da1bb5",
      "FunctionVersion": "1",
      "Name": "LIVE",
      "Description": "alias for live version of function"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [ListAliases](#) di Referensi AWS CLI Perintah.

## list-event-source-mappings

Contoh kode berikut menunjukkan cara menggunakan `list-event-source-mappings`.

### AWS CLI

Untuk membuat daftar pemetaan sumber peristiwa untuk suatu fungsi

`list-event-source-mappings` Contoh berikut menampilkan daftar pemetaan sumber peristiwa untuk fungsi `Lambdamy-function`.

```
aws lambda list-event-source-mappings \
  --function-name my-function
```

Output:

```
{
  "EventSourceMappings": [
    {
      "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "StateTransitionReason": "USER_INITIATED",
      "LastModified": 1569284520.333,
      "BatchSize": 5,
      "State": "Enabled",
      "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
```

```
        "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Pemetaan Sumber Peristiwa AWS Lambda di Panduan Pengembang](#) Lambda AWS .

- Untuk detail API, lihat [ListEventSourceMappings](#) di Referensi AWS CLI Perintah.

## list-function-event-invoke-configs

Contoh kode berikut menunjukkan cara menggunakan `list-function-event-invoke-configs`.

### AWS CLI

Untuk melihat daftar konfigurasi pemanggilan asinkron

`list-function-event-invoke-configs` Contoh berikut mencantumkan konfigurasi pemanggilan asinkron untuk fungsi yang ditentukan.

```
aws lambda list-function-event-invoke-configs \
  --function-name my-function
```

### Output:

```
{
  "FunctionEventInvokeConfigs": [
    {
      "LastModified": 1577824406.719,
      "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:GREEN",
      "MaximumRetryAttempts": 2,
      "MaximumEventAgeInSeconds": 1800
    },
    {
      "LastModified": 1577824396.653,
      "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
      "MaximumRetryAttempts": 0,
      "MaximumEventAgeInSeconds": 3600
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [ListFunctionEventInvokeConfigs](#) di Referensi AWS CLI Perintah.

## list-functions

Contoh kode berikut menunjukkan cara menggunakan `list-functions`.

### AWS CLI

Untuk mengambil daftar fungsi Lambda

`list-functions` Contoh berikut menampilkan daftar semua fungsi untuk pengguna saat ini.

```
aws lambda list-functions
```

Output:

```
{
  "Functions": [
    {
      "TracingConfig": {
        "Mode": "PassThrough"
      },
      "Version": "$LATEST",
      "CodeSha256": "dBG9m8SGdmlEjw/JYXlhhvCrAv5TxvXsbl/RMr0fT/I=",
      "FunctionName": "helloworld",
      "MemorySize": 128,
      "RevisionId": "1718e831-badf-4253-9518-d0644210af7b",
      "CodeSize": 294,
      "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:helloworld",
      "Handler": "helloworld.handler",
      "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-zgur6bf4",
      "Timeout": 3,
      "LastModified": "2023-09-23T18:32:33.857+0000",
      "Runtime": "nodejs18.x",
      "Description": ""
    },
    {
      "TracingConfig": {
        "Mode": "PassThrough"
      }
    }
  ]
}
```

```

    },
    "Version": "$LATEST",
    "CodeSha256": "sU0cJ2/h0ZevwV/1TxCuQqK3gDZP3i8gUoqUUVmY6E=",
    "FunctionName": "my-function",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "MemorySize": 256,
    "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668",
    "CodeSize": 266,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qqq",
    "Timeout": 3,
    "LastModified": "2023-10-01T16:47:28.490+0000",
    "Runtime": "nodejs18.x",
    "Description": ""
},
{
    "Layers": [
        {
            "CodeSize": 41784542,
            "Arn": "arn:aws:lambda:us-west-2:420165488524:layer:AWSLambda-
Python37-SciPy1x:2"
        },
        {
            "CodeSize": 4121,
            "Arn": "arn:aws:lambda:us-
west-2:123456789012:layer:pythonLayer:1"
        }
    ],
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "Version": "$LATEST",
    "CodeSha256": "ZQukCqxtkqFgyF2cU41Avj99TKQ/hNihPtDtRcc08mI=",
    "FunctionName": "my-python-function",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",

```

```

        "SecurityGroupIds": [],
      },
      "MemorySize": 128,
      "RevisionId": "80b4eabc-acf7-4ea8-919a-e874c213707d",
      "CodeSize": 299,
      "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
python-function",
      "Handler": "lambda_function.lambda_handler",
      "Role": "arn:aws:iam::123456789012:role/service-role/my-python-function-
role-z5g7dr6n",
      "Timeout": 3,
      "LastModified": "2023-10-01T19:40:41.643+0000",
      "Runtime": "python3.11",
      "Description": ""
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [ListFunctions](#) di Referensi AWS CLI Perintah.

## list-layer-versions

Contoh kode berikut menunjukkan cara menggunakan `list-layer-versions`.

### AWS CLI

Untuk membuat daftar versi layer AWS Lambda

`list-layers-versions` Contoh berikut menampilkan informasi tentang versi untuk layer bernama `my-layer`.

```
aws lambda list-layer-versions \
  --layer-name my-layer
```

Output:

```
{
  "Layers": [
    {
```



```

    "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-
layer:2",
    "Version": 2,
    "Description": "My layer",
    "CreateDate": "2023-11-15T00:37:46.592+0000",
    "CompatibleRuntimes": [
        "python3.10",
        "python3.11"
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [ListLayerVersions](#) di Referensi AWS CLI Perintah.

## list-layers

Contoh kode berikut menunjukkan cara menggunakan `list-layers`.

### AWS CLI

Untuk membuat daftar layer yang kompatibel dengan runtime fungsi Anda

`list-layers` Contoh berikut menampilkan informasi tentang layer yang kompatibel dengan runtime Python 3.11.

```

aws lambda list-layers \
  --compatible-runtime python3.11

```

### Output:

```

{
  "Layers": [
    {
      "LayerName": "my-layer",
      "LayerArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer",
      "LatestMatchingVersion": {
        "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-
layer:2",
        "Version": 2,

```

```

    "Description": "My layer",
    "CreateDate": "2023-11-15T00:37:46.592+0000",
    "CompatibleRuntimes": [
      "python3.10",
      "python3.11"
    ]
  }
]
}

```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [ListLayers](#) di Referensi AWS CLI Perintah.

## list-provisioned-concurrency-configs

Contoh kode berikut menunjukkan cara menggunakan `list-provisioned-concurrency-configs`.

### AWS CLI

Untuk mendapatkan daftar konfigurasi konkurensi yang disediakan

`list-provisioned-concurrency-configs` Contoh berikut mencantumkan konfigurasi konkurensi yang disediakan untuk fungsi yang ditentukan.

```

aws lambda list-provisioned-concurrency-configs \
  --function-name my-function

```

### Output:

```

{
  "ProvisionedConcurrencyConfigs": [
    {
      "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:GREEN",
      "RequestedProvisionedConcurrentExecutions": 100,
      "AvailableProvisionedConcurrentExecutions": 100,
      "AllocatedProvisionedConcurrentExecutions": 100,
      "Status": "READY",
    }
  ]
}

```

```
        "LastModified": "2019-12-31T20:29:00+0000"
      },
      {
        "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
        "RequestedProvisionedConcurrentExecutions": 100,
        "AvailableProvisionedConcurrentExecutions": 100,
        "AllocatedProvisionedConcurrentExecutions": 100,
        "Status": "READY",
        "LastModified": "2019-12-31T20:28:49+0000"
      }
    ]
  }
}
```

- Untuk detail API, lihat [ListProvisionedConcurrencyConfigs](#) di Referensi AWS CLI Perintah.

## list-tags

Contoh kode berikut menunjukkan cara menggunakan `list-tags`.

### AWS CLI

Untuk mengambil daftar tag untuk fungsi Lambda

`list-tags` Contoh berikut menampilkan tag yang dilampirkan ke fungsi `my-function` Lambda.

```
aws lambda list-tags \
  --resource arn:aws:lambda:us-west-2:123456789012:function:my-function
```

Output:

```
{
  "Tags": {
    "Category": "Web Tools",
    "Department": "Sales"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai Fungsi Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [ListTags](#) di Referensi AWS CLI Perintah.

## list-versions-by-function

Contoh kode berikut menunjukkan cara menggunakan `list-versions-by-function`.

### AWS CLI

Untuk mengambil daftar versi fungsi

`list-versions-by-function` Contoh berikut menampilkan daftar versi untuk fungsi `my-function` Lambda.

```
aws lambda list-versions-by-function \  
--function-name my-function
```

Output:

```
{  
  "Versions": [  
    {  
      "TracingConfig": {  
        "Mode": "PassThrough"  
      },  
      "Version": "$LATEST",  
      "CodeSha256": "sU0cJ2/h0ZevwV/1TxCuQqK3gDZP3i8gUoqUUVRmY6E=",  
      "FunctionName": "my-function",  
      "VpcConfig": {  
        "SubnetIds": [],  
        "VpcId": "",  
        "SecurityGroupIds": []  
      },  
      "MemorySize": 256,  
      "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668",  
      "CodeSize": 266,  
      "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-  
function:$LATEST",  
      "Handler": "index.handler",  
      "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-  
role-uy3l9qq",  
      "Timeout": 3,  
      "LastModified": "2019-10-01T16:47:28.490+0000",  
      "Runtime": "nodejs10.x",  
      "Description": ""  
    },  
  ],  
}
```

```
{
  "TracingConfig": {
    "Mode": "PassThrough"
  },
  "Version": "1",
  "CodeSha256": "5tT2qgzYUHoqWR616pZ2dpkn/0J1FrzJmlKidWaaCgk=",
  "FunctionName": "my-function",
  "VpcConfig": {
    "SubnetIds": [],
    "VpcId": "",
    "SecurityGroupIds": []
  },
  "MemorySize": 256,
  "RevisionId": "949c8914-012e-4795-998c-e467121951b1",
  "CodeSize": 304,
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:1",
  "Handler": "index.handler",
  "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
  "Timeout": 3,
  "LastModified": "2019-09-26T20:28:40.438+0000",
  "Runtime": "nodejs10.x",
  "Description": "new version"
},
{
  "TracingConfig": {
    "Mode": "PassThrough"
  },
  "Version": "2",
  "CodeSha256": "sU0cJ2/h0ZevwV/1TxCuQqK3gDZP3i8gUoqUUVRmY6E=",
  "FunctionName": "my-function",
  "VpcConfig": {
    "SubnetIds": [],
    "VpcId": "",
    "SecurityGroupIds": []
  },
  "MemorySize": 256,
  "RevisionId": "cd669f21-0f3d-4e1c-9566-948837f2e2ea",
  "CodeSize": 266,
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:2",
  "Handler": "index.handler",
```

```

        "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
        "Timeout": 3,
        "LastModified": "2019-10-01T16:47:28.490+0000",
        "Runtime": "nodejs10.x",
        "Description": "newer version"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [ListVersionsByFunction](#) di Referensi AWS CLI Perintah.

## publish-layer-version

Contoh kode berikut menunjukkan cara menggunakan `publish-layer-version`.

### AWS CLI

Untuk membuat versi layer Lambda

`publish-layer-version` Contoh berikut membuat versi layer pustaka Python baru. Perintah mengambil konten lapisan file bernama `layer.zip` dalam bucket S3 yang ditentukan.

```

aws lambda publish-layer-version \
  --layer-name my-layer \
  --description "My Python layer" \
  --license-info "MIT" \
  --content S3Bucket=lambda-layers-us-west-2-123456789012,S3Key=layer.zip \
  --compatible-runtimes python3.10 python3.11

```

Output:

```

{
  "Content": {
    "Location": "https://awslambda-us-west-2-layers.s3.us-west-2.amazonaws.com/
snapshots/123456789012/my-layer-4aaa2fbb-ff77-4b0a-ad92-5b78a716a96a?
versionId=27iWyA73cCAYqyH...",
    "CodeSha256": "tv9jJ0+rPbXUUXuRKi7CwHzKtLDkDRJLB3cC3Z/ouXo=",
    "CodeSize": 169
  }
}

```

```
    },
    "LayerArn": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer",
    "LayerVersionArn": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer:1",
    "Description": "My Python layer",
    "CreateDate": "2023-11-14T23:03:52.894+0000",
    "Version": 1,
    "LicenseInfo": "MIT",
    "CompatibleRuntimes": [
      "python3.10",
      "python3.11"
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [PublishLayerVersion](#) di Referensi AWS CLI Perintah.

## publish-version

Contoh kode berikut menunjukkan cara menggunakan `publish-version`.

### AWS CLI

Untuk mempublikasikan versi baru dari suatu fungsi

`publish-version` Contoh berikut menerbitkan versi baru dari fungsi `my-function` Lambda.

```
aws lambda publish-version \
  --function-name my-function
```

Output:

```
{
  "TracingConfig": {
    "Mode": "PassThrough"
  },
  "CodeSha256": "dBG9m8SGdm1Ejw/JYX1hhvCrAv5TxvXsbl/RM10fT/I=",
  "FunctionName": "my-function",
  "CodeSize": 294,
  "RevisionId": "f31d3d39-cc63-4520-97d4-43cd44c94c20",
  "MemorySize": 128,
```

```
"FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:3",
"Version": "2",
"Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-
zgur6bf4",
"Timeout": 3,
"LastModified": "2019-09-23T18:32:33.857+0000",
"Handler": "my-function.handler",
"Runtime": "nodejs10.x",
"Description": ""
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [PublishVersion](#) di Referensi AWS CLI Perintah.

## put-function-concurrency

Contoh kode berikut menunjukkan cara menggunakan `put-function-concurrency`.

### AWS CLI

Untuk mengonfigurasi batas konkurensi cadangan untuk suatu fungsi

`put-function-concurrency` Contoh berikut mengonfigurasi 100 eksekusi bersamaan yang dicadangkan untuk fungsi tersebut. `my-function`

```
aws lambda put-function-concurrency \
  --function-name my-function \
  --reserved-concurrent-executions 100
```

Output:

```
{
  "ReservedConcurrentExecutions": 100
}
```

Untuk informasi selengkapnya, lihat [Reservasi Konkurensi untuk Fungsi Lambda di](#) Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [PutFunctionConcurrency](#) di Referensi AWS CLI Perintah.



## put-function-event-invoke-config

Contoh kode berikut menunjukkan cara menggunakan `put-function-event-invoke-config`.

### AWS CLI

Untuk mengonfigurasi penanganan kesalahan untuk pemanggilan asinkron

`put-function-event-invoke-config` Contoh berikut menetapkan usia acara maksimum satu jam dan menonaktifkan percobaan ulang untuk fungsi yang ditentukan.

```
aws lambda put-function-event-invoke-config \  
  --function-name my-function \  
  --maximum-event-age-in-seconds 3600 \  
  --maximum-retry-attempts 0
```

### Output:

```
{  
  "LastModified": 1573686021.479,  
  "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-function:  
$LATEST",  
  "MaximumRetryAttempts": 0,  
  "MaximumEventAgeInSeconds": 3600,  
  "DestinationConfig": {  
    "OnSuccess": {},  
    "OnFailure": {}  
  }  
}
```

- Untuk detail API, lihat [PutFunctionEventInvokeConfig](#) di Referensi AWS CLI Perintah.

## put-provisioned-concurrency-config

Contoh kode berikut menunjukkan cara menggunakan `put-provisioned-concurrency-config`.

### AWS CLI

Untuk mengalokasikan konkurensi yang disediakan

`put-provisioned-concurrency-config` Contoh berikut mengalokasikan 100 konkurensi yang disediakan untuk BLUE alias fungsi yang ditentukan.

```
aws lambda put-provisioned-concurrency-config \  
  --function-name my-function \  
  --qualifier BLUE \  
  --provisioned-concurrent-executions 100
```

Output:

```
{  
  "Requested ProvisionedConcurrentExecutions": 100,  
  "Allocated ProvisionedConcurrentExecutions": 0,  
  "Status": "IN_PROGRESS",  
  "LastModified": "2019-11-21T19:32:12+0000"  
}
```

- Untuk detail API, lihat [PutProvisionedConcurrencyConfig](#) di Referensi AWS CLI Perintah.

## remove-layer-version-permission

Contoh kode berikut menunjukkan cara menggunakan `remove-layer-version-permission`.

### AWS CLI

Untuk menghapus izin versi lapisan

`remove-layer-version-permission` Contoh berikut menghapus izin untuk akun untuk mengkonfigurasi versi lapisan.

```
aws lambda remove-layer-version-permission \  
  --layer-name my-layer \  
  --statement-id xaccount \  
  --version-number 1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Lapisan AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [RemoveLayerVersionPermission](#) di Referensi AWS CLI Perintah.

## remove-permission

Contoh kode berikut menunjukkan cara menggunakan `remove-permission`.

### AWS CLI

Untuk menghapus izin dari fungsi Lambda yang ada

`remove-permission` Contoh berikut menghapus izin untuk memanggil fungsi bernama `my-function`.

```
aws lambda remove-permission \  
  --function-name my-function \  
  --statement-id sns
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan Kebijakan Berbasis Sumber Daya untuk Lambda AWS di Panduan Pengembang Lambda](#).AWS

- Untuk detail API, lihat [RemovePermission](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke fungsi Lambda yang ada

`tag-resource` Contoh berikut menambahkan tag dengan nama kunci `DEPARTMENT` dan nilai `Department A` untuk fungsi Lambda tertentu.

```
aws lambda tag-resource \  
  --resource arn:aws:lambda:us-west-2:123456789012:function:my-function \  
  --tags "DEPARTMENT=Department A"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Fungsi Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari fungsi Lambda yang ada

`untag-resource` Contoh berikut menghapus tag dengan DEPARTMENT tag nama kunci dari fungsi `my-function` Lambda.

```
aws lambda untag-resource \  
  --resource arn:aws:lambda:us-west-2:123456789012:function:my-function \  
  --tag-keys DEPARTMENT
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Fungsi Lambda di Panduan](#) Pengembang AWS Lambda.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-alias

Contoh kode berikut menunjukkan cara menggunakan `update-alias`.

### AWS CLI

Untuk memperbarui alias fungsi

`update-alias` Contoh berikut memperbarui alias bernama LIVE untuk menunjuk ke versi 3 dari fungsi `my-function` Lambda.

```
aws lambda update-alias \  
  --function-name my-function \  
  --function-version 3 \  
  --name LIVE
```

Output:

```
{
```

```
"FunctionVersion": "3",
"Name": "LIVE",
"AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
"RevisionId": "594f41fb-b85f-4c20-95c7-6ca5f2a92c93",
"Description": "alias for live version of function"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Alias Fungsi AWS Lambda](#) di Panduan Pengembang AWS Lambda.

- Untuk detail API, lihat [UpdateAlias](#) di Referensi AWS CLI Perintah.

## update-event-source-mapping

Contoh kode berikut menunjukkan cara menggunakan `update-event-source-mapping`.

### AWS CLI

Untuk memperbarui pemetaan antara sumber peristiwa dan fungsi AWS Lambda

`update-event-source-mapping` Contoh berikut memperbarui ukuran batch ke 8 dalam pemetaan yang ditentukan.

```
aws lambda update-event-source-mapping \
  --uuid "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \
  --batch-size 8
```

Output:

```
{
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "StateTransitionReason": "USER_INITIATED",
  "LastModified": 1569284520.333,
  "BatchSize": 8,
  "State": "Updating",
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Untuk informasi selengkapnya, lihat [Pemetaan Sumber Peristiwa AWS Lambda di Panduan Pengembang Lambda AWS](#).

- Untuk detail API, lihat [UpdateEventSourceMapping](#) di Referensi AWS CLI Perintah.

## update-function-code

Contoh kode berikut menunjukkan cara menggunakan `update-function-code`.

### AWS CLI

Untuk memperbarui kode fungsi Lambda

`update-function-code` Contoh berikut menggantikan kode versi `my-function` fungsi yang tidak dipublikasikan (`$LATEST`) dengan isi file zip yang ditentukan.

```
aws lambda update-function-code \  
  --function-name my-function \  
  --zip-file fileb://my-function.zip
```

### Output:

```
{  
  "FunctionName": "my-function",  
  "LastModified": "2019-09-26T20:28:40.438+0000",  
  "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",  
  "MemorySize": 256,  
  "Version": "$LATEST",  
  "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy319qqq",  
  "Timeout": 3,  
  "Runtime": "nodejs10.x",  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJm1KidWoaCgk=",  
  "Description": "",  
  "VpcConfig": {  
    "SubnetIds": [],  
    "VpcId": "",  
    "SecurityGroupIds": []  
  },  
  "CodeSize": 304,  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "Handler": "index.handler"  
}
```

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [UpdateFunctionCode](#) di Referensi AWS CLI Perintah.

## update-function-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-function-configuration`.

### AWS CLI

Untuk memodifikasi konfigurasi suatu fungsi

`update-function-configuration` Contoh berikut memodifikasi ukuran memori menjadi 256 MB untuk versi fungsi yang tidak dipublikasikan (`$LATEST`). `my-function`

```
aws lambda update-function-configuration \  
  --function-name my-function \  
  --memory-size 256
```

### Output:

```
{  
  "FunctionName": "my-function",  
  "LastModified": "2019-09-26T20:28:40.438+0000",  
  "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",  
  "MemorySize": 256,  
  "Version": "$LATEST",  
  "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy319qqq",  
  "Timeout": 3,  
  "Runtime": "nodejs10.x",  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJmLKidWoaCgk=",  
  "Description": "",  
  "VpcConfig": {  
    "SubnetIds": [],  
    "VpcId": "",  
    "SecurityGroupIds": []  
  },  
  "CodeSize": 304,
```

```
"FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
"Handler": "index.handler"
}
```

Untuk informasi selengkapnya, lihat [Konfigurasi Fungsi AWS Lambda di Panduan Pengembang AWS Lambda](#).

- Untuk detail API, lihat [UpdateFunctionConfiguration](#) di Referensi AWS CLI Perintah.

## update-function-event-invoke-config

Contoh kode berikut menunjukkan cara menggunakan `update-function-event-invoke-config`.

### AWS CLI

Untuk memperbarui konfigurasi pemanggilan asinkron

`update-function-event-invoke-config` Contoh berikut menambahkan tujuan on-failure ke konfigurasi pemanggilan asinkron yang ada untuk fungsi yang ditentukan.

```
aws lambda update-function-event-invoke-config \
  --function-name my-function \
  --destination-config '{"OnFailure":{"Destination": "arn:aws:sqs:us-
east-2:123456789012:destination"}}'
```

### Output:

```
{
  "LastModified": 1573687896.493,
  "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-function:
$LATEST",
  "MaximumRetryAttempts": 0,
  "MaximumEventAgeInSeconds": 3600,
  "DestinationConfig": {
    "OnSuccess": {},
    "OnFailure": {
      "Destination": "arn:aws:sqs:us-east-2:123456789012:destination"
    }
  }
}
```



- Untuk detail API, lihat [UpdateFunctionEventInvokeConfig](#) di Referensi AWS CLI Perintah.

## Contoh License Manager menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With License Manager.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-license-configuration**

Contoh kode berikut menunjukkan cara menggunakan `create-license-configuration`.

AWS CLI

Contoh 1: Untuk membuat konfigurasi lisensi

`create-license-configuration` Contoh berikut membuat konfigurasi lisensi dengan batas keras 10 core.

```
aws license-manager create-license-configuration --name my-license-configuration \  
  --license-counting-type Core \  
  --license-count 10 \  
  --license-count-hard-limit
```

Output:

```
{
  "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba41EXAMPLE1111"
}
```

Contoh 2: Untuk membuat konfigurasi lisensi

`create-license-configuration` Contoh berikut membuat konfigurasi lisensi dengan batas lunak 100 vCPU. Ini menggunakan aturan untuk mengaktifkan optimasi vCPU.

```
aws license-manager create-license-configuration --name my-license-configuration
--license-counting-type vCPU \
--license-count 100 \
--license-rules "#honorVcpuOptimization=true"
```

Output:

```
{
  "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba41EXAMPLE2222"
}
```

- Untuk detail API, lihat [CreateLicenseConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-license-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-license-configuration`.

AWS CLI

Untuk menghapus konfigurasi lisensi

`delete-license-configuration` Contoh berikut menghapus konfigurasi lisensi yang ditentukan.

```
aws license-manager delete-license-configuration \
--license-configuration-arn arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteLicenseConfiguration](#) di Referensi AWS CLI Perintah.

## get-license-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-license-configuration`.

### AWS CLI

Untuk mendapatkan informasi konfigurasi lisensi

`get-license-configuration` Contoh berikut menampilkan rincian untuk konfigurasi lisensi yang ditentukan.

```
aws license-manager get-license-configuration \  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

### Output:

```
{  
  "LicenseConfigurationId": "lic-38b658717b87478aaa7c00883EXAMPLE",  
  "LicenseConfigurationArn": "arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE",  
  "Name": "my-license-configuration",  
  "LicenseCountingType": "vCPU",  
  "LicenseRules": [],  
  "LicenseCountHardLimit": false,  
  "ConsumedLicenses": 0,  
  "Status": "AVAILABLE",  
  "OwnerAccountId": "123456789012",  
  "ConsumedLicenseSummaryList": [  
    {  
      "ResourceType": "EC2_INSTANCE",  
      "ConsumedLicenses": 0  
    },  
    {  
      "ResourceType": "EC2_HOST",  
      "ConsumedLicenses": 0  
    },  
    {  
      "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",  
      "ConsumedLicenses": 0  
    }  
  ]  
}
```

```
    }
  ],
  "ManagedResourceSummaryList": [
    {
      "ResourceType": "EC2_INSTANCE",
      "AssociationCount": 0
    },
    {
      "ResourceType": "EC2_HOST",
      "AssociationCount": 0
    },
    {
      "ResourceType": "EC2_AMI",
      "AssociationCount": 2
    },
    {
      "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
      "AssociationCount": 0
    }
  ]
}
```

- Untuk detail API, lihat [GetLicenseConfiguration](#) di Referensi AWS CLI Perintah.

## get-service-settings

Contoh kode berikut menunjukkan cara menggunakan `get-service-settings`.

### AWS CLI

Untuk mendapatkan pengaturan License Manager

`get-service-settings` Contoh berikut menampilkan pengaturan layanan untuk License Manager di Region saat ini.

```
aws license-manager get-service-settings
```

Berikut ini menunjukkan contoh keluaran jika penemuan sumber daya lintas akun dinonaktifkan.

```
{
  "OrganizationConfiguration": {
    "EnableIntegration": false
  }
}
```

```

    },
    "EnableCrossAccountsDiscovery": false
  }

```

Berikut ini menunjukkan contoh keluaran jika penemuan sumber daya lintas akun diaktifkan.

```

{
  "S3BucketArn": "arn:aws:s3:::aws-license-manager-service-c22d6279-35c4-47c4-bb",
  "OrganizationConfiguration": {
    "EnableIntegration": true
  },
  "EnableCrossAccountsDiscovery": true
}

```

- Untuk detail API, lihat [GetServiceSettings](#) di Referensi AWS CLI Perintah.

## list-associations-for-license-configuration

Contoh kode berikut menunjukkan cara menggunakan `list-associations-for-license-configuration`.

### AWS CLI

Untuk mendapatkan asosiasi untuk konfigurasi lisensi

`list-associations-for-license-configuration` Contoh berikut menampilkan informasi rinci untuk asosiasi konfigurasi lisensi yang ditentukan.

```

aws license-manager list-associations-for-license-configuration \
  --license-configuration-arn arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE

```

Output:

```

{
  "LicenseConfigurationAssociations": [
    {
      "ResourceArn": "arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0",
      "ResourceType": "EC2_AMI",
      "ResourceOwnerId": "123456789012",
      "AssociationTime": 1568825118.617
    },
  ],
}

```

```

    {
      "ResourceArn": "arn:aws:ec2:us-west-2::image/ami-0abcdef1234567890",
      "ResourceType": "EC2_AMI",
      "ResourceOwnerId": "123456789012",
      "AssociationTime": 1568825118.946
    }
  ]
}

```

- Untuk detail API, lihat [ListAssociationsForLicenseConfiguration](#) di Referensi AWS CLI Perintah.

## list-license-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-license-configurations`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua konfigurasi lisensi Anda

`list-license-configurations` Contoh berikut mencantumkan semua konfigurasi lisensi Anda.

```
aws license-manager list-license-configurations
```

Output:

```

{
  "LicenseConfigurations": [
    {
      "LicenseConfigurationId": "lic-6eb6586f508a786a2ba4f56c1EXAMPLE",
      "LicenseConfigurationArn": "arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE",
      "Name": "my-license-configuration",
      "LicenseCountingType": "Core",
      "LicenseRules": [],
      "LicenseCount": 10,
      "LicenseCountHardLimit": true,
      "ConsumedLicenses": 0,
      "Status": "AVAILABLE",
      "OwnerAccountId": "123456789012",
      "ConsumedLicenseSummaryList": [
        {

```

```

        "ResourceType": "EC2_INSTANCE",
        "ConsumedLicenses": 0
    },
    {
        "ResourceType": "EC2_HOST",
        "ConsumedLicenses": 0
    },
    {
        "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
        "ConsumedLicenses": 0
    }
],
"ManagedResourceSummaryList": [
    {
        "ResourceType": "EC2_INSTANCE",
        "AssociationCount": 0
    },
    {
        "ResourceType": "EC2_HOST",
        "AssociationCount": 0
    },
    {
        "ResourceType": "EC2_AMI",
        "AssociationCount": 0
    },
    {
        "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
        "AssociationCount": 0
    }
]
},
{
    ...
}
]
}

```

Contoh 2: Untuk membuat daftar konfigurasi lisensi tertentu

`list-license-configurations` Contoh berikut hanya mencantumkan konfigurasi lisensi yang ditentukan.

```
aws license-manager list-license-configurations \
```

```
--license-configuration-arns arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

- Untuk detail API, lihat [ListLicenseConfigurations](#) di Referensi AWS CLI Perintah.

## **list-license-specifications-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-license-specifications-for-resource`.

### AWS CLI

Untuk membuat daftar konfigurasi lisensi untuk sumber daya

`list-license-specifications-for-resource` Contoh berikut mencantumkan konfigurasi lisensi yang terkait dengan Amazon Machine Image (AMI) yang ditentukan.

```
aws license-manager list-license-specifications-for-resource \
  --resource-arn arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0
```

Output:

```
{
  "LicenseConfigurationArn": "arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE"
}
```

- Untuk detail API, lihat [ListLicenseSpecificationsForResource](#) di Referensi AWS CLI Perintah.

## **list-resource-inventory**

Contoh kode berikut menunjukkan cara menggunakan `list-resource-inventory`.

### AWS CLI

Untuk membuat daftar sumber daya dalam inventaris sumber daya

`list-resource-inventory` Contoh berikut mencantumkan sumber daya yang dikelola menggunakan inventaris Systems Manager.

```
aws license-manager list-resource-inventory
```



**Output:**

```
{
  "ResourceInventoryList": [
    {
      "Platform": "Red Hat Enterprise Linux Server",
      "ResourceType": "EC2Instance",
      "PlatformVersion": "7.4",
      "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-05d3cdfb05bd36376",
      "ResourceId": "i-05d3cdfb05bd36376",
      "ResourceOwningAccountId": "1234567890129"
    },
    {
      "Platform": "Amazon Linux",
      "ResourceType": "EC2Instance",
      "PlatformVersion": "2",
      "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-0b1d036cfd4594808",
      "ResourceId": "i-0b1d036cfd4594808",
      "ResourceOwningAccountId": "1234567890129"
    },
    {
      "Platform": "Microsoft Windows Server 2019 Datacenter",
      "ResourceType": "EC2Instance",
      "PlatformVersion": "10.0.17763",
      "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-0cdb3b54a2a8246ad",
      "ResourceId": "i-0cdb3b54a2a8246ad",
      "ResourceOwningAccountId": "1234567890129"
    }
  ]
}
```

- Untuk detail API, lihat [ListResourceInventory](#) di Referensi AWS CLI Perintah.

**list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

**AWS CLI**

Untuk membuat daftar tag untuk konfigurasi lisensi

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk konfigurasi lisensi yang ditentukan.

```
aws license-manager list-tags-for-resource \  
  --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-  
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "project",  
      "Value": "lima"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## `list-usage-for-license-configuration`

Contoh kode berikut menunjukkan cara menggunakan `list-usage-for-license-configuration`.

AWS CLI

Untuk membuat daftar lisensi yang digunakan untuk konfigurasi lisensi

`list-usage-for-license-configuration` Contoh berikut mencantumkan informasi tentang sumber daya yang menggunakan lisensi untuk konfigurasi lisensi yang ditentukan. Misalnya, jika jenis lisensinya adalah vCPU, setiap instance menggunakan satu lisensi per vCPU.

```
aws license-manager list-usage-for-license-configuration \  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

Output:

```
{
```

```
"LicenseConfigurationUsageList": [  
  {  
    "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/  
i-04a636d18e83cfacb",  
    "ResourceType": "EC2_INSTANCE",  
    "ResourceStatus": "running",  
    "ResourceOwnerId": "123456789012",  
    "AssociationTime": 1570892850.519,  
    "ConsumedLicenses": 2  
  }  
]  
}
```

- Untuk detail API, lihat [ListUsageForLicenseConfiguration](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag konfigurasi lisensi

`tag-resource` Contoh berikut menambahkan tag yang ditentukan (nama kunci dan nilai) ke konfigurasi lisensi yang ditentukan.

```
aws license-manager tag-resource \  
  --tags Key=project,Value=lima \  
  --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-  
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari konfigurasi lisensi

`untag-resource` Contoh berikut menghapus tag yang ditentukan (nama kunci dan sumber daya) dari konfigurasi lisensi yang ditentukan.

```
aws license-manager untag-resource \  
  --tag-keys project \  
  --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-  
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-license-configuration**

Contoh kode berikut menunjukkan cara menggunakan `update-license-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi lisensi

`update-license-configuration` Contoh berikut memperbarui konfigurasi lisensi yang ditentukan untuk menghapus batas keras.

```
aws license-manager update-license-configuration \  
  --no-license-count-hard-limit \  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:880185128111:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Perintah ini tidak menghasilkan output.

`update-license-configuration` Contoh berikut memperbarui konfigurasi lisensi yang ditentukan untuk mengubah statusnya menjadi `DISABLED`.

```
aws license-manager update-license-configuration \  
  --license-configuration-status DISABLED  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:880185128111:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateLicenseConfiguration](#) di Referensi AWS CLI Perintah.

## update-license-specifications-for-resource

Contoh kode berikut menunjukkan cara menggunakan `update-license-specifications-for-resource`.

### AWS CLI

Untuk memperbarui konfigurasi lisensi untuk sumber daya

`update-license-specifications-for-resource` Contoh berikut menggantikan konfigurasi lisensi yang terkait dengan Amazon Machine Image (AMI) yang ditentukan dengan menghapus satu konfigurasi lisensi dan menambahkan yang lain.

```
aws license-manager update-license-specifications-for-resource \
  --resource-arn arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0 \
  --remove-license-specifications LicenseConfigurationArn=arn:aws:license-
manager:us-west-2:123456789012:license-
configuration:lic-38b658717b87478aaa7c00883EXAMPLE \
  --add-license-specifications LicenseConfigurationArn=arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-42b6deb06e5399a980d555927EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateLicenseSpecificationsForResource](#) di Referensi AWS CLI Perintah.

## update-service-settings

Contoh kode berikut menunjukkan cara menggunakan `update-service-settings`.

### AWS CLI

Untuk memperbarui pengaturan License Manager

`update-service-settings` Contoh berikut memungkinkan penemuan sumber daya lintas akun untuk License Manager di AWS Wilayah saat ini. Bucket Amazon S3 adalah Sinkronisasi Data Sumber Daya yang diperlukan untuk inventaris Systems Manager.

```
aws license-manager update-service-settings \
  --organization-configuration EnableIntegration=true \
  --enable-cross-accounts-discovery \
  --s3-bucket-arn arn:aws:s3:::aws-license-manager-service-abcd1234EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateServiceSettings](#) di Referensi AWS CLI Perintah.

## Contoh Lightsail menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Lightsail.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **allocate-static-ip**

Contoh kode berikut menunjukkan cara menggunakan `allocate-static-ip`.

AWS CLI

Untuk membuat IP statis

`allocate-static-ip` Contoh berikut menciptakan IP statis yang ditentukan, yang dapat dilampirkan ke sebuah instance.

```
aws lightsail allocate-static-ip \  
  --static-ip-name StaticIp-1
```

Output:

```
{
```

```

"operations": [
  {
    "id": "b5d06d13-2f19-4683-889f-dEXAMPLEed79",
    "resourceName": "StaticIp-1",
    "resourceType": "StaticIp",
    "createdAt": 1571071325.076,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "AllocateStaticIp",
    "status": "Succeeded",
    "statusChangedAt": 1571071325.274
  }
]
}

```

- Untuk detail API, lihat [AllocateStaticIp](#) di Referensi AWS CLI Perintah.

## attach-disk

Contoh kode berikut menunjukkan cara menggunakan `attach-disk`.

### AWS CLI

Untuk melampirkan disk penyimpanan blok ke sebuah instance

`attach-disk` Contoh berikut melampirkan disk `Disk-1` untuk contoh `WordPress_Multisite-1` dengan jalur disk dari `/dev/xvdf`

```

aws lightsail attach-disk \
  --disk-name Disk-1 \
  --disk-path /dev/xvdf \
  --instance-name WordPress_Multisite-1

```

### Output:

```

{
  "operations": [
    {
      "id": "10a08267-19ce-43be-b913-6EXAMPLE7e80",

```

```

    "resourceName": "Disk-1",
    "resourceType": "Disk",
    "createdAt": 1571071465.472,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "WordPress_Multisite-1",
    "operationType": "AttachDisk",
    "status": "Started",
    "statusChangedAt": 1571071465.472
  },
  {
    "id": "2912c477-5295-4539-88c9-bEXAMPLEd1f0",
    "resourceName": "WordPress_Multisite-1",
    "resourceType": "Instance",
    "createdAt": 1571071465.474,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "Disk-1",
    "operationType": "AttachDisk",
    "status": "Started",
    "statusChangedAt": 1571071465.474
  }
]
}

```

- Untuk detail API, lihat [AttachDisk](#) di Referensi AWS CLI Perintah.

## **attach-instances-to-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `attach-instances-to-load-balancer`.

### AWS CLI

Untuk melampirkan instance ke penyeimbang beban

`attach-instances-to-load-balancer` Contoh berikut melampirkan instance MEAN-1, MEAN-2, dan MEAN-3 ke penyeimbang beban. `LoadBalancer-1`



```
aws lightsail attach-instances-to-load-balancer \  
  --instance-names {"MEAN-1","MEAN-2","MEAN-3"} \  
  --load-balancer-name LoadBalancer-1
```

**Output:**

```
{  
  "operations": [  
    {  
      "id": "8055d19d-abb2-40b9-b527-1EXAMPLE3c7b",  
      "resourceName": "LoadBalancer-1",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1571071699.892,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "MEAN-2",  
      "operationType": "AttachInstancesToLoadBalancer",  
      "status": "Started",  
      "statusChangedAt": 1571071699.892  
    },  
    {  
      "id": "c35048eb-8538-456a-a118-0EXAMPLEfb73",  
      "resourceName": "MEAN-2",  
      "resourceType": "Instance",  
      "createdAt": 1571071699.887,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "LoadBalancer-1",  
      "operationType": "AttachInstancesToLoadBalancer",  
      "status": "Started",  
      "statusChangedAt": 1571071699.887  
    },  
    {  
      "id": "910d09e0-adc5-4372-bc2e-0EXAMPLEd891",  
      "resourceName": "LoadBalancer-1",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1571071699.882,
```

```
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "MEAN-3",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.882
  },
  {
    "id": "178b18ac-43e8-478c-9bed-1EXAMPLE4755",
    "resourceName": "MEAN-3",
    "resourceType": "Instance",
    "createdAt": 1571071699.901,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "LoadBalancer-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.901
  },
  {
    "id": "fb62536d-2a98-4190-a6fc-4EXAMPLE7470",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571071699.885,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "MEAN-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.885
  },
  {
    "id": "787dac0d-f98d-46c3-8571-3EXAMPLE5a85",
    "resourceName": "MEAN-1",
    "resourceType": "Instance",
```

```

    "createdAt": 1571071699.901,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "LoadBalancer-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.901
  }
]
}

```

- Untuk detail API, lihat [AttachInstancesToLoadBalancer](#) di Referensi AWS CLI Perintah.

## attach-load-balancer-tls-certificate

Contoh kode berikut menunjukkan cara menggunakan `attach-load-balancer-tls-certificate`.

### AWS CLI

Untuk melampirkan sertifikat TLS ke penyeimbang beban

`attach-load-balancer-tls-certificate` Contoh berikut melampirkan sertifikat TLS penyeimbang beban `Certificate2` ke penyeimbang beban `LoadBalancer-1`

```

aws lightsail attach-load-balancer-tls-certificate \
  --certificate-name Certificate2 \
  --load-balancer-name LoadBalancer-1

```

### Output:

```

{
  "operations": [
    {
      "id": "cf1ad6e3-3cbb-4b8a-a7f2-3EXAMPLEa118",
      "resourceName": "LoadBalancer-1",
      "resourceType": "LoadBalancer",
      "createdAt": 1571072255.416,
      "location": {

```

```

        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "Certificate2",
    "operationType": "AttachLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1571072255.416
  },
  {
    "id": "dae1bcfb-d531-4c06-b4ea-bEXAMPLEc04e",
    "resourceName": "Certificate2",
    "resourceType": "LoadBalancerTlsCertificate",
    "createdAt": 1571072255.416,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "LoadBalancer-1",
    "operationType": "AttachLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1571072255.416
  }
]
}

```

- Untuk detail API, lihat [AttachLoadBalancerTlsCertificate](#) di Referensi AWS CLI Perintah.

## attach-static-ip

Contoh kode berikut menunjukkan cara menggunakan `attach-static-ip`.

### AWS CLI

Untuk melampirkan IP statis ke sebuah instance

`attach-static-ip` Contoh berikut melampirkan IP statis `StaticIp-1` ke instance `MEAN-1`.

```

aws lightsail attach-static-ip \
  --static-ip-name StaticIp-1 \
  --instance-name MEAN-1

```

**Output:**

```
{
  "operations": [
    {
      "id": "45e6fa13-4808-4b8d-9292-bEXAMPLE20b2",
      "resourceName": "StaticIp-1",
      "resourceType": "StaticIp",
      "createdAt": 1571072569.375,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationDetails": "MEAN-1",
      "operationType": "AttachStaticIp",
      "status": "Succeeded",
      "statusChangedAt": 1571072569.375
    },
    {
      "id": "9ee09a17-863c-4e51-8a6d-3EXAMPLE5475",
      "resourceName": "MEAN-1",
      "resourceType": "Instance",
      "createdAt": 1571072569.376,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationDetails": "StaticIp-1",
      "operationType": "AttachStaticIp",
      "status": "Succeeded",
      "statusChangedAt": 1571072569.376
    }
  ]
}
```

- Untuk detail API, lihat [AttachStaticIp](#) di Referensi AWS CLI Perintah.

**close-instance-public-ports**

Contoh kode berikut menunjukkan cara menggunakan `close-instance-public-ports`.

## AWS CLI

Untuk menutup port firewall untuk sebuah instance

`close-instance-public-ports` Contoh berikut menutup port TCP 22 pada contoh. MEAN-2

```
aws lightsail close-instance-public-ports \  
  --instance-name MEAN-2 \  
  --port-info fromPort=22,protocol=TCP,toPort=22
```

Output:

```
{  
  "operation": {  
    "id": "4f328636-1c96-4649-ae6d-1EXAMPLEf446",  
    "resourceName": "MEAN-2",  
    "resourceType": "Instance",  
    "createdAt": 1571072845.737,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "isTerminal": true,  
    "operationDetails": "22/tcp",  
    "operationType": "CloseInstancePublicPorts",  
    "status": "Succeeded",  
    "statusChangedAt": 1571072845.737  
  }  
}
```

- Untuk detail API, lihat [CloseInstancePublicPorts](#) di Referensi AWS CLI Perintah.

## copy-snapshot

Contoh kode berikut menunjukkan cara menggunakan `copy-snapshot`.

### AWS CLI

Contoh 1: Untuk menyalin snapshot dalam Wilayah yang sama AWS

`copy-snapshot` Contoh berikut menyalin snapshot instance MEAN-1-1571075291 sebagai snapshot instance MEAN-1-Copy dalam Wilayah yang sama AWS . us-west-2

```
aws lightsail copy-snapshot \  
  --source-snapshot-name MEAN-1-1571075291 \  
  --target-snapshot-name MEAN-1-Copy \  
  --source-region us-west-2
```

Output:

```
{  
  "operations": [  
    {  
      "id": "ced16fc1-f401-4556-8d82-1EXAMPLEb982",  
      "resourceName": "MEAN-1-Copy",  
      "resourceType": "InstanceSnapshot",  
      "createdAt": 1571075581.498,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "us-west-2:MEAN-1-1571075291",  
      "operationType": "CopySnapshot",  
      "status": "Started",  
      "statusChangedAt": 1571075581.498  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menyalin snapshot dari satu AWS Wilayah ke Wilayah lainnya di Amazon Lightsail di Panduan Pengembang Lightsail](#).

Contoh 2: Untuk menyalin snapshot dari satu AWS Wilayah ke Wilayah lainnya

copy-snapshot Contoh berikut menyalin snapshot instance MEAN-1-1571075291 sebagai snapshot instance MEAN-1-1571075291-Copy dari AWS Region us-west-2 ke us-east-1

```
aws lightsail copy-snapshot \  
  --source-snapshot-name MEAN-1-1571075291 \  
  --target-snapshot-name MEAN-1-1571075291-Copy \  
  --source-region us-west-2 \  
  --region us-east-1
```

Output:

```
{
  "operations": [
    {
      "id": "91116b79-119c-4451-b44a-dEXAMPLEd97b",
      "resourceName": "MEAN-1-1571075291-Copy",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1571075695.069,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-east-1"
      },
      "isTerminal": false,
      "operationDetails": "us-west-2:MEAN-1-1571075291",
      "operationType": "CopySnapshot",
      "status": "Started",
      "statusChangedAt": 1571075695.069
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyalin snapshot dari satu AWS Wilayah ke Wilayah lainnya di Amazon Lightsail di Panduan Pengembang Lightsail](#).

Contoh 3: Untuk menyalin snapshot otomatis dalam Wilayah yang sama AWS

copy-snapshot Contoh berikut menyalin snapshot 2019-10-14 otomatis instance WordPress-1 sebagai snapshot manual WordPress-1-10142019 di Wilayah. AWS us-west-2

```
aws lightsail copy-snapshot \
  --source-resource-name WordPress-1 \
  --restore-date 2019-10-14 \
  --target-snapshot-name WordPress-1-10142019 \
  --source-region us-west-2
```

Output:

```
{
  "operations": [
    {
      "id": "be3e6754-cd1d-48e6-ad9f-2EXAMPLE1805",
      "resourceName": "WordPress-1-10142019",
```



```

    "resourceType": "InstanceSnapshot",
    "createdAt": 1571082412.311,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "us-west-2:WordPress-1",
    "operationType": "CopySnapshot",
    "status": "Started",
    "statusChangedAt": 1571082412.311
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menyimpan snapshot otomatis instance atau disk di Amazon Lightsail dalam Panduan Pengembang Lightsail](#).

Contoh 4: Untuk menyalin snapshot otomatis dari satu AWS Wilayah ke Wilayah lainnya

copy-snapshot Contoh berikut menyalin snapshot 2019-10-14 otomatis instance WordPress-1 sebagai snapshot manual WordPress-1-10142019 dari AWS Region us-west-2 ke us-east-1

```

aws lightsail copy-snapshot \
  --source-resource-name WordPress-1 \
  --restore-date 2019-10-14 \
  --target-snapshot-name WordPress-1-10142019 \
  --source-region us-west-2 \
  --region us-east-1

```

Output:

```

{
  "operations": [
    {
      "id": "dffa128b-0b07-476e-b390-bEXAMPLE3775",
      "resourceName": "WordPress-1-10142019",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1571082493.422,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-east-1"
      }
    }
  ]
}

```

```
    },
    "isTerminal": false,
    "operationDetails": "us-west-2:WordPress-1",
    "operationType": "CopySnapshot",
    "status": "Started",
    "statusChangedAt": 1571082493.422
  }
]
}
```

Untuk informasi selengkapnya, lihat [Menyimpan snapshot otomatis instance atau disk di Amazon Lightsail dalam Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [CopySnapshot](#) di Referensi AWS CLI Perintah.

## create-disk-from-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-disk-from-snapshot`.

### AWS CLI

Untuk membuat membuat disk dari snapshot disk

`create-disk-from-snapshot` Contoh berikut membuat disk penyimpanan blok bernama `Disk-2` dari snapshot disk penyimpanan blok tertentu. Disk dibuat di AWS Wilayah dan Zona Ketersediaan yang ditentukan, dengan ruang penyimpanan 32 GB.

```
aws lightsail create-disk-from-snapshot \
  --disk-name Disk-2 \
  --disk-snapshot-name Disk-1-1566839161 \
  --availability-zone us-west-2a \
  --size-in-gb 32
```

Output:

```
{
  "operations": [
    {
      "id": "d42b605d-5ef1-4b4a-8791-7a3e8b66b5e7",
      "resourceName": "Disk-2",
      "resourceType": "Disk",
      "createdAt": 1569624941.471,

```

```

        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": false,
        "operationType": "CreateDiskFromSnapshot",
        "status": "Started",
        "statusChangedAt": 1569624941.791
    }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat disk penyimpanan blok dari snapshot di Amazon Lightsail di Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [CreateDiskFromSnapshot](#) di Referensi AWS CLI Perintah.

## create-disk-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-disk-snapshot`.

### AWS CLI

Contoh 1: Untuk membuat snapshot dari disk

`create-disk-snapshot` Contoh berikut membuat snapshot bernama `DiskSnapshot-1` disk penyimpanan blok tertentu.

```

aws lightsail create-disk-snapshot \
  --disk-name Disk-1 \
  --disk-snapshot-name DiskSnapshot-1

```

Output:

```

{
  "operations": [
    {
      "id": "fa74c6d2-03a3-4f42-a7c7-792f124d534b",
      "resourceName": "DiskSnapshot-1",
      "resourceType": "DiskSnapshot",
      "createdAt": 1569625129.739,
      "location": {
        "availabilityZone": "all",

```

```

        "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "Disk-1",
    "operationType": "CreateDiskSnapshot",
    "status": "Started",
    "statusChangedAt": 1569625129.739
},
{
    "id": "920a25df-185c-4528-87cd-7b85f5488c06",
    "resourceName": "Disk-1",
    "resourceType": "Disk",
    "createdAt": 1569625129.739,
    "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "DiskSnapshot-1",
    "operationType": "CreateDiskSnapshot",
    "status": "Started",
    "statusChangedAt": 1569625129.739
}
]
}

```

Contoh 2: Untuk membuat snapshot dari disk sistem instance

`create-disk-snapshot` Contoh berikut membuat snapshot dari disk sistem instance tertentu.

```

aws lightsail create-disk-snapshot \
  --instance-name WordPress-1 \
  --disk-snapshot-name SystemDiskSnapshot-1

```

Output:

```

{
  "operations": [
    {
      "id": "f508cf1c-6597-42a6-a4c3-4aebd75af0d9",
      "resourceName": "SystemDiskSnapshot-1",
      "resourceType": "DiskSnapshot",
      "createdAt": 1569625294.685,

```

```

    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "WordPress-1",
    "operationType": "CreateDiskSnapshot",
    "status": "Started",
    "statusChangedAt": 1569625294.685
  },
  {
    "id": "0bb9f712-da3b-4d99-b508-3bf871d989e5",
    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "createdAt": 1569625294.685,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "SystemDiskSnapshot-1",
    "operationType": "CreateDiskSnapshot",
    "status": "Started",
    "statusChangedAt": 1569625294.685
  }
]
}

```

Untuk informasi selengkapnya, lihat [Snapshot di Amazon Lightsail dan Membuat snapshot volume root instance di Amazon Lightsail di Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [CreateDiskSnapshot](#) di Referensi AWS CLI Perintah.

## create-disk

Contoh kode berikut menunjukkan cara menggunakan `create-disk`.

### AWS CLI

Untuk membuat disk penyimpanan blok

`create-disk` Contoh berikut membuat disk penyimpanan blok `Disk-1` di AWS Wilayah dan Availability Zone yang ditentukan, dengan 32 GB ruang penyimpanan.

```
aws lightsail create-disk \  
  --disk-name Disk-1 \  
  --availability-zone us-west-2a \  
  --size-in-gb 32
```

Output:

```
{  
  "operations": [  
    {  
      "id": "1c85e2ec-86ba-4697-b936-77f4d3dc013a",  
      "resourceName": "Disk-1",  
      "resourceType": "Disk",  
      "createdAt": 1569449220.36,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateDisk",  
      "status": "Started",  
      "statusChangedAt": 1569449220.588  
    }  
  ]  
}
```

- Untuk detail API, lihat [CreateDisk](#) di Referensi AWS CLI Perintah.

## create-domain-entry

Contoh kode berikut menunjukkan cara menggunakan `create-domain-entry`.

### AWS CLI

Untuk membuat entri domain (catatan DNS)

`create-domain-entry` Contoh berikut membuat catatan DNS (A) untuk puncak domain tertentu yang menunjuk ke alamat IP instans.

Catatan: Operasi API terkait domain Lightsail hanya tersedia di Wilayah. `us-east-1` Jika profil CLI Anda dikonfigurasi untuk menggunakan Wilayah yang berbeda, Anda harus menyertakan `--region us-east-1` parameter atau perintah gagal.

```
aws lightsail create-domain-entry \  
  --region us-east-1 \  
  --domain-name example.com \  
  --domain-entry name=example.com,type=A,target=192.0.2.0
```

Output:

```
{  
  "operation": {  
    "id": "5be4494d-56f4-41fc-8730-693dcd0ef9e2",  
    "resourceName": "example.com",  
    "resourceType": "Domain",  
    "createdAt": 1569865296.519,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "global"  
    },  
    "isTerminal": true,  
    "operationType": "CreateDomainEntry",  
    "status": "Succeeded",  
    "statusChangedAt": 1569865296.519  
  }  
}
```

Untuk informasi selengkapnya, lihat [DNS di Amazon Lightsail dan Membuat zona DNS untuk mengelola catatan DNS domain Anda di Amazon Lightsail dalam Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [CreateDomainEntry](#) di Referensi AWS CLI Perintah.

## create-domain

Contoh kode berikut menunjukkan cara menggunakan `create-domain`.

### AWS CLI

Untuk membuat domain (zona DNS)

`create-domain` Contoh berikut membuat zona DNS untuk domain tertentu.

Catatan: Operasi API terkait domain Lightsail hanya tersedia di Wilayah. `us-east-1` Jika profil CLI Anda dikonfigurasi untuk menggunakan Wilayah yang berbeda, Anda harus menyertakan `--region us-east-1` parameter atau perintah gagal.

```
aws lightsail create-domain \  
  --region us-east-1 \  
  --domain-name example.com
```

### Output:

```
{  
  "operation": {  
    "id": "64e522c8-9ae1-4c05-9b65-3f237324dc34",  
    "resourceName": "example.com",  
    "resourceType": "Domain",  
    "createdAt": 1569864291.92,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "global"  
    },  
    "isTerminal": true,  
    "operationType": "CreateDomain",  
    "status": "Succeeded",  
    "statusChangedAt": 1569864292.109  
  }  
}
```

Untuk informasi selengkapnya, lihat [DNS di Amazon Lightsail dan Membuat zona DNS untuk mengelola catatan DNS domain Anda di Amazon Lightsail dalam Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [CreateDomain](#) di Referensi AWS CLI Perintah.

## create-instance-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-instance-snapshot`.

### AWS CLI

Untuk membuat snapshot dari sebuah instance

`create-instance-snapshot` Contoh berikut membuat snapshot dari contoh yang ditentukan.

```
aws lightsail create-instance-snapshot \  
  --instance-name WordPress-1 \  
  --instance-snapshot-name WordPress-Snapshot-1
```



**Output:**

```
{
  "operations": [
    {
      "id": "4c3db559-9dd0-41e7-89c0-2cb88c19786f",
      "resourceName": "WordPress-Snapshot-1",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1569866438.48,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "WordPress-1",
      "operationType": "CreateInstanceSnapshot",
      "status": "Started",
      "statusChangedAt": 1569866438.48
    },
    {
      "id": "c04fdc45-2981-488c-88b5-d6d2fd759a6a",
      "resourceName": "WordPress-1",
      "resourceType": "Instance",
      "createdAt": 1569866438.48,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "WordPress-Snapshot-1",
      "operationType": "CreateInstanceSnapshot",
      "status": "Started",
      "statusChangedAt": 1569866438.48
    }
  ]
}
```

- Untuk detail API, lihat [CreateInstanceSnapshot](#) di Referensi AWS CLI Perintah.

**create-instances-from-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `create-instances-from-snapshot`.

## AWS CLI

Untuk membuat instance dari snapshot

`create-instances-from-snapshot` Contoh berikut membuat instance dari snapshot instance yang ditentukan, di AWS Region dan Availability Zone yang ditentukan, menggunakan bundel \$10 USD.

Catatan: Bundel yang Anda tentukan harus sama dengan atau lebih besar dalam spesifikasi daripada bundel instance sumber asli yang digunakan untuk membuat snapshot.

```
aws lightsail create-instances-from-snapshot \  
  --instance-snapshot-name WordPress-1-1569866208 \  
  --instance-names WordPress-2 \  
  --availability-zone us-west-2a \  
  --bundle-id medium_2_0
```

Output:

```
{  
  "operations": [  
    {  
      "id": "003f8271-b711-464d-b9b8-7f3806cb496e",  
      "resourceName": "WordPress-2",  
      "resourceType": "Instance",  
      "createdAt": 1569865914.908,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstancesFromSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569865914.908  
    }  
  ]  
}
```

- Untuk detail API, lihat [CreateInstancesFromSnapshot](#) di Referensi AWS CLI Perintah.

## create-instances

Contoh kode berikut menunjukkan cara menggunakan `create-instances`.

### AWS CLI

Contoh 1: Untuk membuat satu contoh

`create-instances` Contoh berikut membuat instance di AWS Region dan Availability Zone yang ditentukan, menggunakan WordPress cetak biru, dan bundel \$3,50 USD.

```
aws lightsail create-instances \  
  --instance-names Instance-1 \  
  --availability-zone us-west-2a \  
  --blueprint-id wordpress_5_1_1_2 \  
  --bundle-id nano_2_0
```

Output:

```
{  
  "operations": [  
    {  
      "id": "9a77158f-7be3-4d6d-8054-cf5ae2b720cc",  
      "resourceName": "Instance-1",  
      "resourceType": "Instance",  
      "createdAt": 1569447986.061,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstance",  
      "status": "Started",  
      "statusChangedAt": 1569447986.061  
    }  
  ]  
}
```

Contoh 2: Untuk membuat beberapa instance sekaligus

`create-instances` Contoh berikut membuat tiga instance di AWS Region dan Availability Zone yang ditentukan, menggunakan WordPress cetak biru, dan bundel \$3,50 USD.

```
aws lightsail create-instances \  
  --instance-names {"Instance1","Instance2","Instance3"} \  
  --availability-zone us-west-2a \  
  --blueprint-id wordpress_5_1_1_2 \  
  --bundle-id nano_2_0
```

## Output:

```
{  
  "operations": [  
    {  
      "id": "5492f015-9d2e-48c6-8eea-b516840e6903",  
      "resourceName": "Instance1",  
      "resourceType": "Instance",  
      "createdAt": 1569448780.054,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstance",  
      "status": "Started",  
      "statusChangedAt": 1569448780.054  
    },  
    {  
      "id": "c58b5f46-2676-44c8-b95c-3ad375898515",  
      "resourceName": "Instance2",  
      "resourceType": "Instance",  
      "createdAt": 1569448780.054,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstance",  
      "status": "Started",  
      "statusChangedAt": 1569448780.054  
    },  
    {  
      "id": "a5ad8006-9bee-4499-9eb7-75e42e6f5882",  
      "resourceName": "Instance3",  
      "resourceType": "Instance",  
      "createdAt": 1569448780.054,
```

```

        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": false,
        "operationType": "CreateInstance",
        "status": "Started",
        "statusChangedAt": 1569448780.054
    }
]
}

```

- Untuk detail API, lihat [CreateInstances](#) di Referensi AWS CLI Perintah.

## create-key-pair

Contoh kode berikut menunjukkan cara menggunakan `create-key-pair`.

### AWS CLI

Untuk membuat pasangan kunci

`create-key-pair` Contoh berikut membuat key pair yang dapat Anda gunakan untuk mengautentikasi dan menghubungkan ke sebuah instance.

```
aws lightsail create-key-pair \
  --key-pair-name MyPersonalKeyPair
```

Outputnya memberikan nilai private key base64 yang dapat Anda gunakan untuk mengautentikasi instance yang menggunakan key pair yang dibuat. Catatan: Salin dan tempel nilai kunci pribadi base64 ke lokasi yang aman karena Anda tidak dapat mengambilnya nanti.

```

{
  "keyPair": {
    "name": "MyPersonalKeyPair",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/55025c71-198f-403b-b42f-a69433e724fb",
    "supportCode": "621291663362/MyPersonalKeyPair",
    "createdAt": 1569866556.567,
    "location": {
      "availabilityZone": "all",

```

```

    "regionName": "us-west-2"
  },
  "resourceType": "KeyPair"
},
"publicKeyBase64": "ssh-rsa ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCV0xUEwx96amPERH7K1bVT1tTF190mNk6o7m5YVhk9x10dMbDRbFvhtXvw4jz
+BUUgedGUXno6uF7agqxZN01kPLJBIVTW26SSYBJ0tE
+y804UyVsjrBUqCaMXDhmfXpWuLMPWuXhwcKh7e8hwoTfkiX0E6Q1
+KqF/MiA3w6DCjEqvvdI07SiEZJFsuGNfYDDN3w60Re15MUhmn30Jdn4y/
A7NWb3IxL4pPfVE4rgFRKU8n1jp9kwRnLVMVB0WuGXk6n+H6M2f1 ",
"privateKeyBase64": "-----BEGIN RSA PRIVATE KEY-----
EXAMPLETCCAFICCCQD6m7oRw0uX0jANBgqhkig9w0BAQUFADCBiDELMAkGA1UEBhMC
\nVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6\nnb24xFDASBgNVBAwTC01BTSBD
\nBgqhkig9w0BCQEWEG5vb25lQGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
\nMTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
\nVQQHEwdTZWF0dGx1MQ8wDQEXAMPLEwZBbWF6b24xFDASBgNVBAwTC01BTSBDb25z
\nb2x1MRIwEAYDVQQDEw1UZXRhbnQ2IHN0YXZlYm90Y251Y251QGFT
\nYXpvbi5jb20wZGZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMEXAMPLE4GmWIWJ
\n21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
\nrDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
\nIbb30hjZncvQAaREXAMPLEM2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4\nnnUHVvXyUntneD9+h8Mg9q6q
+auNkyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
\nFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780EXAMPLELvjx79LjStb
\nNYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=\n-----END RSA PRIVATE KEY-----",
"operation": {
  "id": "67f984db-9994-45fe-ad38-59bafcaf82ef",
  "resourceName": "MyPersonalKeyPair",
  "resourceType": "KeyPair",
  "createdAt": 1569866556.567,
  "location": {
    "availabilityZone": "all",
    "regionName": "us-west-2"
  },
  "isTerminal": true,
  "operationType": "CreateKeyPair",
  "status": "Succeeded",
  "statusChangedAt": 1569866556.704
}
}

```

- Untuk detail API, lihat [CreateKeyPair](#) di Referensi AWS CLI Perintah.

## create-load-balancer-tls-certificate

Contoh kode berikut menunjukkan cara menggunakan `create-load-balancer-tls-certificate`.

### AWS CLI

Untuk membuat sertifikat TLS untuk penyeimbang beban

`create-load-balancer-tls-certificate` Contoh berikut membuat sertifikat TLS yang dilampirkan ke penyeimbang beban tertentu. Sertifikat yang dibuat berlaku untuk domain yang ditentukan. Catatan: Hanya dua sertifikat yang dapat dibuat untuk penyeimbang beban.

```
aws lightsail create-load-balancer-tls-certificate \  
  --certificate-alternative-names abc.example.com \  
  --certificate-domain-name example.com \  
  --certificate-name MySecondCertificate \  
  --load-balancer-name MyFirstLoadBalancer
```

### Output:

```
{  
  "operations": [  
    {  
      "id": "be663aed-cb46-41e2-9b23-e2f747245bd4",  
      "resourceName": "MySecondCertificate",  
      "resourceType": "LoadBalancerTlsCertificate",  
      "createdAt": 1569867364.971,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "MyFirstLoadBalancer",  
      "operationType": "CreateLoadBalancerTlsCertificate",  
      "status": "Succeeded",  
      "statusChangedAt": 1569867365.219  
    },  
    {  
      "id": "f3dfa930-969e-41cc-ac7d-337178716f6d",  
      "resourceName": "MyFirstLoadBalancer",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1569867364.971,  
    }  
  ]  
}
```

```

        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "MySecondCertificate",
        "operationType": "CreateLoadBalancerTlsCertificate",
        "status": "Succeeded",
        "statusChangedAt": 1569867365.219
    }
]
}

```

- Untuk detail API, lihat [CreateLoadBalancerTlsCertificate](#) di Referensi AWS CLI Perintah.

## create-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `create-load-balancer`.

### AWS CLI

Untuk membuat penyeimbang beban

`create-load-balancer` Contoh berikut membuat penyeimbang beban dengan sertifikat TLS. Sertifikat TLS berlaku untuk domain yang ditentukan, dan mengarahkan lalu lintas ke instance di port 80.

```

aws lightsail create-load-balancer \
  --certificate-alternative-names www.example.com test.example.com \
  --certificate-domain-name example.com \
  --certificate-name Certificate-1 \
  --instance-port 80 \
  --load-balancer-name LoadBalancer-1

```

Output:

```

{
  "operations": [
    {
      "id": "cc7b920a-83d8-4762-a74e-9174fe1540be",
      "resourceName": "LoadBalancer-1",
      "resourceType": "LoadBalancer",

```



```
    "createdAt": 1569867169.406,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "CreateLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1569867169.406
  },
  {
    "id": "658ed43b-f729-42f3-a8e4-3f8024d3c98d",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancerTlsCertificate",
    "createdAt": 1569867170.193,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "LoadBalancer-1",
    "operationType": "CreateLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1569867170.54
  },
  {
    "id": "4757a342-5181-4870-b1e0-227eebc35ab5",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1569867170.193,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "Certificate-1",
    "operationType": "CreateLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1569867170.54
  }
]
}
```

Untuk informasi selengkapnya, lihat penyeimbang [beban Lightsail](#) di Panduan Pengembang Lightsail.

- Untuk detail API, lihat [CreateLoadBalancer](#) di Referensi AWS CLI Perintah.

## create-relational-database-from-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-relational-database-from-snapshot`.

### AWS CLI

Untuk membuat database terkelola dari snapshot

`create-relational-database-from-snapshot` Contoh berikut membuat database terkelola dari snapshot yang ditentukan di AWS Region dan Availability Zone yang ditentukan, menggunakan paket database standar \$15 USD. Catatan: Bundel yang Anda tentukan harus sama dengan atau lebih besar dalam spesifikasi daripada bundel database sumber asli yang digunakan untuk membuat snapshot.

```
aws lightsail create-relational-database-from-snapshot \  
  --relational-database-snapshot-name Database-0regon-1-1566839359 \  
  --relational-database-name Database-1 \  
  --availability-zone us-west-2a \  
  --relational-database-bundle-id micro_1_0 \  
  --no-publicly-accessible
```

### Output:

```
{  
  "operations": [  
    {  
      "id": "ad6d9193-9d5c-4ea1-97ae-8fe6de600b4c",  
      "resourceName": "Database-1",  
      "resourceType": "RelationalDatabase",  
      "createdAt": 1569867916.938,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,
```

```

    "operationType": "CreateRelationalDatabaseFromSnapshot",
    "status": "Started",
    "statusChangedAt": 1569867918.643
  }
]
}

```

- Untuk detail API, lihat [CreateRelationalDatabaseFromSnapshot](#) di Referensi AWS CLI Perintah.

## create-relational-database-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-relational-database-snapshot`.

### AWS CLI

Untuk membuat snapshot dari database terkelola

`create-relational-database-snapshot` Contoh berikut membuat snapshot dari database terkelola yang ditentukan.

```

aws lightsail create-relational-database-snapshot \
  --relational-database-name Database1 \
  --relational-database-snapshot-name RelationalDatabaseSnapshot1

```

Output:

```

{
  "operations": [
    {
      "id": "853667fb-ea91-4c02-8d20-8fc5fd43b9eb",
      "resourceName": "RelationalDatabaseSnapshot1",
      "resourceType": "RelationalDatabaseSnapshot",
      "createdAt": 1569868074.645,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "Database1",
      "operationType": "CreateRelationalDatabaseSnapshot",
      "status": "Started",

```

```

        "statusChangedAt": 1569868074.645
    },
    {
        "id": "fbafa521-3cac-4be8-9773-1c143780b239",
        "resourceName": "Database1",
        "resourceType": "RelationalDatabase",
        "createdAt": 1569868074.645,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": false,
        "operationDetails": "RelationalDatabaseSnapshot1",
        "operationType": "CreateRelationalDatabaseSnapshot",
        "status": "Started",
        "statusChangedAt": 1569868074.645
    }
]
}

```

- Untuk detail API, lihat [CreateRelationalDatabaseSnapshot](#) di Referensi AWS CLI Perintah.

## create-relational-database

Contoh kode berikut menunjukkan cara menggunakan `create-relational-database`.

### AWS CLI

Untuk membuat database terkelola

`create-relational-database` Contoh berikut membuat database terkelola di AWS Region dan Availability Zone yang ditentukan, menggunakan mesin database MySQL 5.6 (`mysql_5_6`), dan paket database standar \$15 USD (`micro_1_0`). Basis data terkelola diisi sebelumnya dengan nama pengguna master, dan tidak dapat diakses publik.

```

aws lightsail create-relational-database \
  --relational-database-name Database-1 \
  --availability-zone us-west-2a \
  --relational-database-blueprint-id mysql_5_6 \
  --relational-database-bundle-id micro_1_0 \
  --master-database-name dbmaster \
  --master-username user \

```

```
--no-publicly-accessible
```

Output:

```
{
  "operations": [
    {
      "id": "b52bedee-73ed-4798-8d2a-9c12df89adcd",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1569450017.244,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "CreateRelationalDatabase",
      "status": "Started",
      "statusChangedAt": 1569450018.637
    }
  ]
}
```

- Untuk detail API, lihat [CreateRelationalDatabase](#) di Referensi AWS CLI Perintah.

## delete-auto-snapshot

Contoh kode berikut menunjukkan cara menggunakan `delete-auto-snapshot`.

AWS CLI

Untuk menghapus snapshot otomatis

`delete-auto-snapshot` Contoh berikut menghapus snapshot otomatis `instance2019-10-10.WordPress-1`

```
aws lightsail delete-auto-snapshot \
  --resource-name WordPress-1 \
  --date 2019-10-10
```

Output:

```
{
  "operations": [
    {
      "id": "31c36e09-3d52-46d5-b6d8-7EXAMPLE534a",
      "resourceName": "WordPress-1",
      "resourceType": "Instance",
      "createdAt": 1571088141.501,
      "location": {
        "availabilityZone": "us-west-2",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationDetails": "DeleteAutoSnapshot-2019-10-10",
      "operationType": "DeleteAutoSnapshot",
      "status": "Succeeded"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menghapus snapshot otomatis instance atau disk di Amazon Lightsail di Panduan Pengembangan Lightsail](#).

- Untuk detail API, lihat [DeleteAutoSnapshot](#) di Referensi AWS CLI Perintah.

## delete-disk-snapshot

Contoh kode berikut menunjukkan cara menggunakan `delete-disk-snapshot`.

### AWS CLI

Untuk menghapus snapshot dari disk penyimpanan blok

`delete-disk-snapshot` Contoh berikut menghapus snapshot yang ditentukan dari disk penyimpanan blok

```
aws lightsail delete-disk-snapshot \
  --disk-snapshot-name DiskSnapshot-1
```

Output:

```
{
```

```
"operations": [  
  {  
    "id": "d1e5766d-b81e-4595-ad5d-02afbcccfd5d",  
    "resourceName": "DiskSnapshot-1",  
    "resourceType": "DiskSnapshot",  
    "createdAt": 1569873552.79,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "us-west-2"  
    },  
    "isTerminal": true,  
    "operationType": "DeleteDiskSnapshot",  
    "status": "Succeeded",  
    "statusChangedAt": 1569873552.79  
  }  
]  
}
```

- Untuk detail API, lihat [DeleteDiskSnapshot](#) di Referensi AWS CLI Perintah.

## delete-disk

Contoh kode berikut menunjukkan cara menggunakan `delete-disk`.

### AWS CLI

Untuk menghapus disk penyimpanan blok

`delete-disk` Contoh berikut menghapus disk penyimpanan blok yang ditentukan.

```
aws lightsail delete-disk \  
  --disk-name Disk-1
```

Output:

```
{  
  "operations": [  
    {  
      "id": "6378c70f-4d75-4f7a-ab66-730fca0bb2fc",  
      "resourceName": "Disk-1",  
      "resourceType": "Disk",  
      "createdAt": 1569872887.864,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationType": "DeleteDiskSnapshot",  
      "status": "Succeeded",  
      "statusChangedAt": 1569872887.864  
    }  
  ]  
}
```

```
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationType": "DeleteDisk",
        "status": "Succeeded",
        "statusChangedAt": 1569872887.864
    }
]
}
```

- Untuk detail API, lihat [DeleteDisk](#) di Referensi AWS CLI Perintah.

## delete-domain-entry

Contoh kode berikut menunjukkan cara menggunakan `delete-domain-entry`.

### AWS CLI

Untuk menghapus entri domain (catatan DNS)

`delete-domain-entry` Contoh berikut menghapus entri domain tertentu dari domain yang ada.

Catatan: Operasi API terkait domain Lightsail hanya tersedia di Wilayah. `us-east-1` Jika profil CLI Anda dikonfigurasi untuk menggunakan Wilayah yang berbeda, Anda harus menyertakan `--region us-east-1` parameter atau perintah gagal.

```
aws lightsail delete-domain-entry \
  --region us-east-1 \
  --domain-name example.com \
  --domain-entry name=123.example.com,target=192.0.2.0,type=A
```

Output:

```
{
  "operation": {
    "id": "06eacd01-d785-420e-8daa-823150c7dca1",
    "resourceName": "example.com ",
    "resourceType": "Domain",
    "createdAt": 1569874157.005,
```



```
    "location": {
      "availabilityZone": "all",
      "regionName": "global"
    },
    "isTerminal": true,
    "operationType": "DeleteDomainEntry",
    "status": "Succeeded",
    "statusChangedAt": 1569874157.005
  }
}
```

- Untuk detail API, lihat [DeleteDomainEntry](#) di Referensi AWS CLI Perintah.

## delete-domain

Contoh kode berikut menunjukkan cara menggunakan `delete-domain`.

### AWS CLI

Untuk menghapus domain (zona DNS)

`delete-domain` Contoh berikut menghapus domain yang ditentukan dan semua entri dalam domain (catatan DNS).

Catatan: Operasi API terkait domain Lightsail hanya tersedia di Wilayah. `us-east-1` Jika profil CLI Anda dikonfigurasi untuk menggunakan Wilayah yang berbeda, Anda harus menyertakan `--region us-east-1` parameter atau perintah gagal.

```
aws lightsail delete-domain \
  --region us-east-1 \
  --domain-name example.com
```

Output:

```
{
  "operation": {
    "id": "fcef5265-5af1-4a46-a3d7-90b5e18b9b32",
    "resourceName": "example.com",
    "resourceType": "Domain",
    "createdAt": 1569873788.13,
    "location": {
```

```
        "availabilityZone": "all",
        "regionName": "global"
    },
    "isTerminal": true,
    "operationType": "DeleteDomain",
    "status": "Succeeded",
    "statusChangedAt": 1569873788.13
}
}
```

- Untuk detail API, lihat [DeleteDomain](#) di Referensi AWS CLI Perintah.

## delete-instance-snapshot

Contoh kode berikut menunjukkan cara menggunakan `delete-instance-snapshot`.

### AWS CLI

judul

`delete-instance-snapshot` Contoh berikut menghapus snapshot tertentu dari sebuah instance.

```
aws lightsail delete-instance-snapshot \
  --instance-snapshot-name WordPress-1-Snapshot-1
```

Output:

```
{
  "operations": [
    {
      "id": "14dad182-976a-46c6-bfd4-9480482bf0ea",
      "resourceName": "WordPress-1-Snapshot-1",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1569874524.562,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "DeleteInstanceSnapshot",

```

```
        "status": "Succeeded",
        "statusChangedAt": 1569874524.562
      }
    ]
  }
```

- Untuk detail API, lihat [DeleteInstanceSnapshot](#) di Referensi AWS CLI Perintah.

## delete-instance

Contoh kode berikut menunjukkan cara menggunakan `delete-instance`.

### AWS CLI

Untuk menghapus sebuah instance

`delete-instance` Contoh berikut menghapus contoh yang ditentukan.

```
aws lightsail delete-instance \
  --instance-name WordPress-1
```

Output:

```
{
  "operations": [
    {
      "id": "d77345a3-8f80-4d2e-b47d-aaa622718df2",
      "resourceName": "Disk-1",
      "resourceType": "Disk",
      "createdAt": 1569874357.469,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "WordPress-1",
      "operationType": "DetachDisk",
      "status": "Started",
      "statusChangedAt": 1569874357.469
    },
    {
      "id": "708fa606-2bfd-4e48-a2c1-0b856585b5b1",
```

```

    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "createdAt": 1569874357.465,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "Disk-1",
    "operationType": "DetachDisk",
    "status": "Started",
    "statusChangedAt": 1569874357.465
  },
  {
    "id": "3187e823-8acb-405d-b098-fad5ceb17bec",
    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "createdAt": 1569874357.829,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "DeleteInstance",
    "status": "Succeeded",
    "statusChangedAt": 1569874357.829
  }
]
}

```

- Untuk detail API, lihat [DeleteInstance](#) di Referensi AWS CLI Perintah.

## delete-key-pair

Contoh kode berikut menunjukkan cara menggunakan `delete-key-pair`.

### AWS CLI

Untuk menghapus pasangan kunci

`delete-key-pair` Contoh berikut menghapus key pair yang ditentukan.

```
aws lightsail delete-key-pair \
```

```
--key-pair-name MyPersonalKeyPair
```

### Output:

```
{
  "operation": {
    "id": "81621463-df38-4810-b866-6e801a15abbf",
    "resourceName": "MyPersonalKeyPair",
    "resourceType": "KeyPair",
    "createdAt": 1569874626.466,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "DeleteKeyPair",
    "status": "Succeeded",
    "statusChangedAt": 1569874626.685
  }
}
```

- Untuk detail API, lihat [DeleteKeyPair](#) di Referensi AWS CLI Perintah.

## delete-known-host-keys

Contoh kode berikut menunjukkan cara menggunakan `delete-known-host-keys`.

### AWS CLI

Untuk menghapus kunci host yang dikenal dari sebuah instance

`delete-known-host-keys` Contoh berikut menghapus kunci host dikenal dari instance yang ditentukan.

```
aws lightsail delete-known-host-keys \
  --instance-name Instance-1
```

### Output:

```
{
  "operations": [
```

```

    {
      "id": "c61afe9c-45a4-41e6-a97e-d212364da3f5",
      "resourceName": "Instance-1",
      "resourceType": "Instance",
      "createdAt": 1569874760.201,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "DeleteKnownHostKeys",
      "status": "Succeeded",
      "statusChangedAt": 1569874760.201
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Memecahkan masalah koneksi dengan klien SSH atau RDP berbasis browser Amazon Lightsail di Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [DeleteKnownHostKeys](#) di Referensi AWS CLI Perintah.

## delete-load-balancer-tls-certificate

Contoh kode berikut menunjukkan cara menggunakan `delete-load-balancer-tls-certificate`.

### AWS CLI

Untuk menghapus sertifikat TLS untuk penyeimbang beban

`delete-load-balancer-tls-certificate` Contoh berikut menghapus sertifikat TLS spesifik dari load balancer yang ditentukan.

```

aws lightsail delete-load-balancer-tls-certificate \
  --load-balancer-name MyFirstLoadBalancer \
  --certificate-name MyFirstCertificate

```

Output:

```

{
  "operations": [

```

```

    {
      "id": "50bec274-e45e-4caa-8a69-b763ef636583",
      "resourceName": "MyFirstCertificate",
      "resourceType": "LoadBalancerTlsCertificate",
      "createdAt": 1569874989.48,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "DeleteLoadBalancerTlsCertificate",
      "status": "Started",
      "statusChangedAt": 1569874989.48
    },
    {
      "id": "78c58cdc-a59a-4b27-8213-500638634a8f",
      "resourceName": "MyFirstLoadBalancer",
      "resourceType": "LoadBalancer",
      "createdAt": 1569874989.48,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "DeleteLoadBalancerTlsCertificate",
      "status": "Started",
      "statusChangedAt": 1569874989.48
    }
  ]
}

```

- Untuk detail API, lihat [DeleteLoadBalancerTlsCertificate](#) di Referensi AWS CLI Perintah.

## delete-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `delete-load-balancer`.

### AWS CLI

Untuk menghapus penyeimbang beban

`delete-load-balancer` Contoh berikut menghapus penyeimbang beban tertentu dan sertifikat TLS terkait.

```
aws lightsail delete-load-balancer \  
  --load-balancer-name MyFirstLoadBalancer
```

### Output:

```
{  
  "operations": [  
    {  
      "id": "a8c968c7-72a3-4680-a714-af8f03eea535",  
      "resourceName": "MyFirstLoadBalancer",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1569875092.125,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationType": "DeleteLoadBalancer",  
      "status": "Succeeded",  
      "statusChangedAt": 1569875092.125  
    },  
    {  
      "id": "f91a29fc-8ce3-4e69-a227-ea70ca890bf5",  
      "resourceName": "MySecondCertificate",  
      "resourceType": "LoadBalancerTlsCertificate",  
      "createdAt": 1569875091.938,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "DeleteLoadBalancerTlsCertificate",  
      "status": "Started",  
      "statusChangedAt": 1569875091.938  
    },  
    {  
      "id": "cf64c060-154b-4eb4-ba57-84e2e41563d6",  
      "resourceName": "MyFirstLoadBalancer",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1569875091.94,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      }  
    }  
  ]  
}
```



```
    },
    "isTerminal": false,
    "operationType": "DeleteLoadBalancerTlsCertificate",
    "status": "Started",
    "statusChangedAt": 1569875091.94
  }
]
}
```

Untuk informasi lebih lanjut, lihat judul di panduan.

- Untuk detail API, lihat [DeleteLoadBalancer](#) di Referensi AWS CLI Perintah.

## delete-relational-database-snapshot

Contoh kode berikut menunjukkan cara menggunakan `delete-relational-database-snapshot`.

### AWS CLI

Untuk menghapus snapshot dari database terkelola

`delete-relational-database-snapshot` Contoh berikut menghapus snapshot tertentu dari database terkelola.

```
aws lightsail delete-relational-database-snapshot \
  --relational-database-snapshot-name Database-0regon-1-1566839359
```

Output:

```
{
  "operations": [
    {
      "id": "b99acae8-735b-4823-922f-30af580e3729",
      "resourceName": "Database-0regon-1-1566839359",
      "resourceType": "RelationalDatabaseSnapshot",
      "createdAt": 1569875293.58,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      }
    },
  ],
}
```

```

        "isTerminal": true,
        "operationType": "DeleteRelationalDatabaseSnapshot",
        "status": "Succeeded",
        "statusChangedAt": 1569875293.58
    }
]
}

```

- Untuk detail API, lihat [DeleteRelationalDatabaseSnapshot](#) di Referensi AWS CLI Perintah.

## delete-relational-database

Contoh kode berikut menunjukkan cara menggunakan `delete-relational-database`.

### AWS CLI

Untuk menghapus database terkelola

`delete-relational-database` Contoh berikut menghapus database terkelola yang ditentukan.

```
aws lightsail delete-relational-database \
  --relational-database-name Database-1
```

Output:

```

{
  "operations": [
    {
      "id": "3b0c41c1-053d-46f0-92a3-14f76141dc86",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1569875210.999,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "DeleteRelationalDatabase",
      "status": "Started",
      "statusChangedAt": 1569875210.999
    }
  ]
}

```

```

    },
    {
      "id": "01ddeae8-a87a-4a4b-a1f3-092c71bf9180",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1569875211.029,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "Database-1-FinalSnapshot-1569875210793",
      "operationType": "CreateRelationalDatabaseSnapshot",
      "status": "Started",
      "statusChangedAt": 1569875211.029
    },
    {
      "id": "74d73681-30e8-4532-974e-1f23cd3f9f73",
      "resourceName": "Database-1-FinalSnapshot-1569875210793",
      "resourceType": "RelationalDatabaseSnapshot",
      "createdAt": 1569875211.029,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "Database-1",
      "operationType": "CreateRelationalDatabaseSnapshot",
      "status": "Started",
      "statusChangedAt": 1569875211.029
    }
  ]
}

```

- Untuk detail API, lihat [DeleteRelationalDatabase](#) di Referensi AWS CLI Perintah.

## detach-static-ip

Contoh kode berikut menunjukkan cara menggunakan `detach-static-ip`.

### AWS CLI

Untuk melepaskan IP statis dari sebuah instance

`detach-static-ip` Contoh berikut melepaskan IP statis `StaticIp-1` dari setiap instance terlampir.

```
aws lightsail detach-static-ip \  
  --static-ip-name StaticIp-1
```

Output:

```
{  
  "operations": [  
    {  
      "id": "2a43d8a3-9f2d-4fe7-bdd0-eEXAMPLE3cf3",  
      "resourceName": "StaticIp-1",  
      "resourceType": "StaticIp",  
      "createdAt": 1571088261.999,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "MEAN-1",  
      "operationType": "DetachStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571088261.999  
    },  
    {  
      "id": "41a7d40c-74e8-4d2e-a837-cEXAMPLEf747",  
      "resourceName": "MEAN-1",  
      "resourceType": "Instance",  
      "createdAt": 1571088262.022,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "StaticIp-1",  
      "operationType": "DetachStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571088262.022  
    }  
  ]  
}
```

- Untuk detail API, lihat [DetachStaticIp](#) di Referensi AWS CLI Perintah.

## get-active-names

Contoh kode berikut menunjukkan cara menggunakan `get-active-names`.

### AWS CLI

Untuk mendapatkan nama sumber daya aktif

`get-active-names` Contoh berikut mengembalikan nama sumber daya aktif di AWS Wilayah dikonfigurasi.

```
aws lightsail get-active-names
```

Output:

```
{
  "activeNames": [
    "WordPress-1",
    "StaticIp-1",
    "MEAN-1",
    "Plesk_Hosting_Stack_on_Ubuntu-1"
  ]
}
```

- Untuk detail API, lihat [GetActiveNames](#) di Referensi AWS CLI Perintah.

## get-auto-snapshots

Contoh kode berikut menunjukkan cara menggunakan `get-auto-snapshots`.

### AWS CLI

Untuk mendapatkan snapshot otomatis yang tersedia untuk sebuah instance

`get-auto-snapshots` Contoh berikut mengembalikan snapshot otomatis yang tersedia misalnya `WordPress-1`.

```
aws lightsail get-auto-snapshots \
```

```
--resource-name WordPress-1
```

Output:

```
{
  "resourceName": "WordPress-1",
  "resourceType": "Instance",
  "autoSnapshots": [
    {
      "date": "2019-10-14",
      "createdAt": 1571033872.0,
      "status": "Success",
      "fromAttachedDisks": []
    },
    {
      "date": "2019-10-13",
      "createdAt": 1570947473.0,
      "status": "Success",
      "fromAttachedDisks": []
    },
    {
      "date": "2019-10-12",
      "createdAt": 1570861072.0,
      "status": "Success",
      "fromAttachedDisks": []
    },
    {
      "date": "2019-10-11",
      "createdAt": 1570774672.0,
      "status": "Success",
      "fromAttachedDisks": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menyimpan snapshot otomatis instance atau disk di Amazon Lightsail dalam Panduan Pengembang Lightsail](#).

- Untuk detail API, lihat [GetAutoSnapshots](#) di Referensi AWS CLI Perintah.

## get-blueprints

Contoh kode berikut menunjukkan cara menggunakan `get-blueprints`.

## AWS CLI

Untuk mendapatkan cetak biru untuk instance baru

`get-blueprints` Contoh berikut menampilkan detail tentang semua cetak biru yang tersedia yang dapat digunakan untuk membuat instance baru di Amazon Lightsail.

```
aws lightsail get-blueprints
```

Output:

```
{
  "blueprints": [
    {
      "blueprintId": "wordpress",
      "name": "WordPress",
      "group": "wordpress",
      "type": "app",
      "description": "Bitnami, the leaders in application packaging, and Automattic, the experts behind WordPress, have teamed up to offer this official WordPress image. This image is a pre-configured, ready-to-run image for running WordPress on Amazon Lightsail. WordPress is the world's most popular content management platform. Whether it's for an enterprise or small business website, or a personal or corporate blog, content authors can easily create content using its new Gutenberg editor, and developers can extend the base platform with additional features. Popular plugins like Jetpack, Akismet, All in One SEO Pack, WP Mail, Google Analytics for WordPress, and Amazon Polly are all pre-installed in this image. Let's Encrypt SSL certificates are supported through an auto-configuration script.",
      "isActive": true,
      "minPower": 0,
      "version": "5.2.2-3",
      "versionCode": "1",
      "productUrl": "https://aws.amazon.com/marketplace/pp/B00NN8Y43U",
      "licenseUrl": "https://d7umqicpi7263.cloudfront.net/eula/product/7d426cb7-9522-4dd7-a56b-55dd8cc1c8d0/588fd495-6492-4610-b3e8-d15ce864454c.txt",
      "platform": "LINUX_UNIX"
    },
    {
      "blueprintId": "lamp_7_1_28",
      "name": "LAMP (PHP 7)",
      "group": "lamp_7",

```

```

        "type": "app",
        "description": "LAMP with PHP 7.x certified by Bitnami greatly
simplifies the development and deployment of PHP applications. It includes the
latest versions of PHP 7.x, Apache and MySQL together with phpMyAdmin and popular
PHP frameworks Zend, Symfony, CodeIgniter, CakePHP, Smarty, and Laravel. Other pre-
configured components and PHP modules include FastCGI, ModSecurity, SQLite, Varnish,
ImageMagick, xDebug, Xcache, OpenLDAP, Memcache, OAuth, PEAR, PECL, APC, GD and
cURL. It is secure by default and supports multiple applications, each with its own
virtual host and project directory. Let's Encrypt SSL certificates are supported
through an auto-configuration script.",
        "isActive": true,
        "minPower": 0,
        "version": "7.1.28",
        "versionCode": "1",
        "productUrl": "https://aws.amazon.com/marketplace/pp/B072JNJZ5C",
        "licenseUrl": "https://d7umqicpi7263.cloudfront.net/eula/product/
cb6afd05-a3b2-4916-a3e6-bccd414f5f21/12ab56cc-6a8c-4977-9611-dcd770824aad.txt",
        "platform": "LINUX_UNIX"
    },
    {
        "blueprintId": "nodejs",
        "name": "Node.js",
        "group": "node",
        "type": "app",
        "description": "Node.js certified by Bitnami is a pre-configured, ready
to run image for Node.js on Amazon EC2. It includes the latest version of Node.js,
Apache, Python and Redis. The image supports multiple Node.js applications, each
with its own virtual host and project directory. It is configured for production
use and is secure by default, as all ports except HTTP, HTTPS and SSH ports are
closed. Let's Encrypt SSL certificates are supported through an auto-configuration
script. Developers benefit from instant access to a secure, update and consistent
Node.js environment without having to manually install and configure multiple
components and libraries.",
        "isActive": true,
        "minPower": 0,
        "version": "12.7.0",
        "versionCode": "1",
        "productUrl": "https://aws.amazon.com/marketplace/pp/B00NNZUAK0",
        "licenseUrl": "https://d7umqicpi7263.cloudfront.net/
eula/product/033793fe-951d-47d0-aa94-5fbd0afb3582/25f8fa66-c868-4d80-
adf8-4a2b602064ae.txt",
        "platform": "LINUX_UNIX"
    },
    ...

```



```
    }  
  ]  
}
```

- Untuk detail API, lihat [GetBlueprints](#) di Referensi AWS CLI Perintah.

## get-bundles

Contoh kode berikut menunjukkan cara menggunakan `get-bundles`.

### AWS CLI

Untuk mendapatkan bundel untuk instance baru

`get-bundles` Contoh berikut menampilkan detail tentang semua bundel yang tersedia yang dapat digunakan untuk membuat instance baru di Amazon Lightsail.

```
aws lightsail get-bundles
```

Output:

```
{  
  "bundles": [  
    {  
      "price": 3.5,  
      "cpuCount": 1,  
      "diskSizeInGb": 20,  
      "bundleId": "nano_2_0",  
      "instanceType": "nano",  
      "isActive": true,  
      "name": "Nano",  
      "power": 300,  
      "ramSizeInGb": 0.5,  
      "transferPerMonthInGb": 1024,  
      "supportedPlatforms": [  
        "LINUX_UNIX"  
      ]  
    },  
    {  
      "price": 5.0,  
      "cpuCount": 1,  
      "diskSizeInGb": 20,  
      "bundleId": "micro_2_0",  
      "instanceType": "micro",  
      "isActive": true,  
      "name": "Micro",  
      "power": 100,  
      "ramSizeInGb": 0.25,  
      "transferPerMonthInGb": 1024,  
      "supportedPlatforms": [  
        "LINUX_UNIX"  
      ]  
    }  
  ]  
}
```

```

        "diskSizeInGb": 40,
        "bundleId": "micro_2_0",
        "instanceType": "micro",
        "isActive": true,
        "name": "Micro",
        "power": 500,
        "ramSizeInGb": 1.0,
        "transferPerMonthInGb": 2048,
        "supportedPlatforms": [
            "LINUX_UNIX"
        ]
    },
    {
        "price": 10.0,
        "cpuCount": 1,
        "diskSizeInGb": 60,
        "bundleId": "small_2_0",
        "instanceType": "small",
        "isActive": true,
        "name": "Small",
        "power": 1000,
        "ramSizeInGb": 2.0,
        "transferPerMonthInGb": 3072,
        "supportedPlatforms": [
            "LINUX_UNIX"
        ]
    },
    ...
}
]
}

```

- Untuk detail API, lihat [GetBundles](#) di Referensi AWS CLI Perintah.

## get-cloud-formation-stack-records

Contoh kode berikut menunjukkan cara menggunakan `get-cloud-formation-stack-records`.

### AWS CLI

Untuk mendapatkan catatan CloudFormation tumpukan dan tumpukan terkait

`get-cloud-formation-stack-records` Contoh berikut menampilkan detail tentang catatan CloudFormation tumpukan dan tumpukan terkait yang digunakan untuk membuat sumber daya Amazon EC2 dari snapshot Amazon Lightsail yang diekspor.

```
aws lightsail get-cloud-formation-stack-records
```

Output:

```
{
  "cloudFormationStackRecords": [
    {
      "name": "CloudFormationStackRecord-588a4243-
e2d1-490d-8200-3a7513ecebdf",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:CloudFormationStackRecord/28d646ab-27bc-48d9-a422-1EXAMPLE6d37",
      "createdAt": 1565301666.586,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "CloudFormationStackRecord",
      "state": "Succeeded",
      "sourceInfo": [
        {
          "resourceType": "ExportSnapshotRecord",
          "name": "ExportSnapshotRecord-
e02f23d7-0453-4aa9-9c95-91aa01a141dd",
          "arn": "arn:aws:lightsail:us-
west-2:111122223333:ExportSnapshotRecord/f12b8792-f3ea-4d6f-b547-2EXAMPLE8796"
        }
      ],
      "destinationInfo": {
        "id": "arn:aws:cloudformation:us-west-2:111122223333:stack/
Lightsail-Stack-588a4243-e2d1-490d-8200-3EXAMPLEebdf/063203b0-
ba28-11e9-838b-0EXAMPLE8b00",
        "service": "Aws::CloudFormation::Stack"
      }
    }
  ]
}
```

- Untuk detail API, lihat [GetCloudFormationStackRecords](#) di Referensi AWS CLI Perintah.

## get-disk-snapshot

Contoh kode berikut menunjukkan cara menggunakan `get-disk-snapshot`.

### AWS CLI

Untuk mendapatkan informasi tentang snapshot disk

`get-disk-snapshot` Contoh berikut menampilkan detail tentang snapshot `Disk-1-1566839161` disk.

```
aws lightsail get-disk-snapshot \  
  --disk-snapshot-name Disk-1-1566839161
```

Output:

```
{  
  "diskSnapshot": {  
    "name": "Disk-1-1566839161",  
    "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/  
e2d0fa53-8ee0-41a0-8e56-0EXAMPLE1051",  
    "supportCode": "6EXAMPLE3362/snap-0EXAMPLE06100d09",  
    "createdAt": 1566839163.749,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "DiskSnapshot",  
    "tags": [],  
    "sizeInGb": 8,  
    "state": "completed",  
    "progress": "100%",  
    "fromDiskName": "Disk-1",  
    "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/  
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",  
    "isFromAutoSnapshot": false  
  }  
}
```

Untuk informasi lebih lanjut, lihat judul di panduan.

- Untuk detail API, lihat [GetDiskSnapshot](#) di Referensi AWS CLI Perintah.

## get-disk-snapshots

Contoh kode berikut menunjukkan cara menggunakan `get-disk-snapshots`.

### AWS CLI

Untuk mendapatkan informasi tentang semua snapshot disk

`get-disk-snapshots` Contoh berikut menampilkan detail tentang semua snapshot disk di AWS Wilayah yang dikonfigurasi.

```
aws lightsail get-disk-snapshots
```

Output:

```
{
  "diskSnapshots": [
    {
      "name": "Disk-2-1571090588",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/32e889a9-38d4-4687-9f21-eEXAMPLE7839",
      "supportCode": "6EXAMPLE3362/snap-0EXAMPLE1ca192a4",
      "createdAt": 1571090591.226,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "DiskSnapshot",
      "tags": [],
      "sizeInGb": 8,
      "state": "completed",
      "progress": "100%",
      "fromDiskName": "Disk-2",
      "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/6a343ff8-6341-422d-86e2-bEXAMPLE16c2",
      "isFromAutoSnapshot": false
    },
    {
      "name": "Disk-1-1566839161",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/e2d0fa53-8ee0-41a0-8e56-0EXAMPLE1051",
      "supportCode": "6EXAMPLE3362/snap-0EXAMPLEe06100d09",
      "createdAt": 1566839163.749,
```

```

    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "DiskSnapshot",
    "tags": [],
    "sizeInGb": 8,
    "state": "completed",
    "progress": "100%",
    "fromDiskName": "Disk-1",
    "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
    "isFromAutoSnapshot": false
  }
]
}

```

- Untuk detail API, lihat [GetDiskSnapshots](#) di Referensi AWS CLI Perintah.

## get-disk

Contoh kode berikut menunjukkan cara menggunakan `get-disk`.

### AWS CLI

Untuk mendapatkan informasi tentang disk penyimpanan blok

`get-disk` Contoh berikut menampilkan rincian tentang disk `Disk-1`.

```
aws lightsail get-disk \
  --disk-name Disk-1
```

### Output:

```

{
  "disk": {
    "name": "Disk-1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
    "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
    "createdAt": 1566585439.587,
    "location": {

```

```
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
    },
    "resourceType": "Disk",
    "tags": [],
    "sizeInGb": 8,
    "isSystemDisk": false,
    "iops": 100,
    "path": "/dev/xvdf",
    "state": "in-use",
    "attachedTo": "WordPress_Multisite-1",
    "isAttached": true,
    "attachmentState": "attached"
}
}
```

Untuk informasi lebih lanjut, lihat judul di panduan.

- Untuk detail API, lihat [GetDisk](#) di Referensi AWS CLI Perintah.

## get-disks

Contoh kode berikut menunjukkan cara menggunakan `get-disks`.

### AWS CLI

Untuk mendapatkan informasi tentang semua disk penyimpanan blok

`get-disks` Contoh berikut menampilkan rincian tentang semua disk di AWS Wilayah dikonfigurasi.

```
aws lightsail get-disks
```

Output:

```
{
  "disks": [
    {
      "name": "Disk-2",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/6a343ff8-6341-422d-86e2-bEXAMPLE16c2",
      "supportCode": "6EXAMPLE3362/vol-0EXAMPLE929602087",
```

```

    "createdAt": 1571090461.634,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "resourceType": "Disk",
    "tags": [],
    "sizeInGb": 8,
    "isSystemDisk": false,
    "iops": 100,
    "state": "available",
    "isAttached": false,
    "attachmentState": "detached"
  },
  {
    "name": "Disk-1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
    "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
    "createdAt": 1566585439.587,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "resourceType": "Disk",
    "tags": [],
    "sizeInGb": 8,
    "isSystemDisk": false,
    "iops": 100,
    "path": "/dev/xvdf",
    "state": "in-use",
    "attachedTo": "WordPress_Multisite-1",
    "isAttached": true,
    "attachmentState": "attached"
  }
]
}

```

- Untuk detail API, lihat [GetDisks](#) di Referensi AWS CLI Perintah.

## get-domain

Contoh kode berikut menunjukkan cara menggunakan `get-domain`.



## AWS CLI

Untuk mendapatkan informasi tentang domain

`get-domain` berikut menampilkan rincian tentang `domainexample.com`.

Catatan: Operasi API terkait domain Lightsail hanya tersedia di Wilayah `us-east-1` AWS. Jika profil CLI Anda dikonfigurasi untuk menggunakan Wilayah yang berbeda, Anda harus menyertakan parameter `--region us-east-1` atau perintah gagal.

```
aws lightsail get-domain \
  --domain-name example.com \
  --region us-east-1
```

Output:

```
{
  "domain": {
    "name": "example.com",
    "arn":
      "arn:aws:lightsail:global:111122223333:Domain/28cda903-3f15-44b2-9baf-3EXAMPLEb304",
    "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEONGSC1",
    "createdAt": 1570728588.6,
    "location": {
      "availabilityZone": "all",
      "regionName": "global"
    },
    "resourceType": "Domain",
    "tags": [],
    "domainEntries": [
      {
        "id": "-1682899164",
        "name": "example.com",
        "target": "192.0.2.0",
        "isAlias": false,
        "type": "A"
      },
      {
        "id": "1703104243",
        "name": "example.com",
        "target": "ns-137.awsdns-17.com",
        "isAlias": false,
        "type": "NS"
      }
    ]
  }
}
```

```

    },
    {
      "id": "-1038331153",
      "name": "example.com",
      "target": "ns-1710.awsdns-21.co.uk",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "-2107289565",
      "name": "example.com",
      "target": "ns-692.awsdns-22.net",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "1582095705",
      "name": "example.com",
      "target": "ns-1436.awsdns-51.org",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "-1769796132",
      "name": "example.com",
      "target": "ns-1710.awsdns-21.co.uk. awsdns-hostmaster.amazon.com. 1
7200 900 1209600 86400",
      "isAlias": false,
      "type": "SOA"
    }
  ]
}

```

- Untuk detail API, lihat [GetDomain](#) di Referensi AWS CLI Perintah.

## get-domains

Contoh kode berikut menunjukkan cara menggunakan `get-domains`.

### AWS CLI

Untuk mendapatkan informasi tentang semua domain

`get-domains` Contoh berikut menampilkan rincian tentang semua domain di AWS Wilayah dikonfigurasi.

Catatan: Operasi API terkait domain Lightsail hanya tersedia di Wilayah. `us-east-1` AWS Jika profil CLI Anda dikonfigurasi untuk menggunakan Wilayah yang berbeda, Anda harus menyertakan `--region us-east-1` parameter atau perintah gagal.

```
aws lightsail get-domains \  
  --region us-east-1
```

Output:

```
{  
  "domains": [  
    {  
      "name": "example.com",  
      "arn":  
"arn:aws:lightsail:global:111122223333:Domain/28cda903-3f15-44b2-9baf-3EXAMPLEb304",  
      "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEONGSC1",  
      "createdAt": 1570728588.6,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "global"  
      },  
      "resourceType": "Domain",  
      "tags": [],  
      "domainEntries": [  
        {  
          "id": "-1682899164",  
          "name": "example.com",  
          "target": "192.0.2.0",  
          "isAlias": false,  
          "type": "A"  
        },  
        {  
          "id": "1703104243",  
          "name": "example.com",  
          "target": "ns-137.awsdns-17.com",  
          "isAlias": false,  
          "type": "NS"  
        },  
        {  
          "id": "-1038331153",
```

```

        "name": "example.com",
        "target": "ns-4567.awsdns-21.co.uk",
        "isAlias": false,
        "type": "NS"
    },
    {
        "id": "-2107289565",
        "name": "example.com",
        "target": "ns-333.awsdns-22.net",
        "isAlias": false,
        "type": "NS"
    },
    {
        "id": "1582095705",
        "name": "example.com",
        "target": "ns-1111.awsdns-51.org",
        "isAlias": false,
        "type": "NS"
    },
    {
        "id": "-1769796132",
        "name": "example.com",
        "target": "ns-1234.awsdns-21.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
        "isAlias": false,
        "type": "SOA"
    },
    {
        "id": "1029454894",
        "name": "_dead6a124ede046a0319eb44a4eb3cbc.example.com",
        "target": "_be133b0a0899fb7b6bf79d9741d1a383.hkvuijqjoua.acm-
validations.aws",
        "isAlias": false,
        "type": "CNAME"
    }
]
},
{
    "name": "example.net",
    "arn": "arn:aws:lightsail:global:111122223333:Domain/9c9f0d70-
c92e-4753-86c2-6EXAMPLE029d",
    "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLE5TPKMV",
    "createdAt": 1556661071.384,
    "location": {

```

```
    "availabilityZone": "all",
    "regionName": "global"
  },
  "resourceType": "Domain",
  "tags": [],
  "domainEntries": [
    {
      "id": "-766320943",
      "name": "example.net",
      "target": "192.0.2.2",
      "isAlias": false,
      "type": "A"
    },
    {
      "id": "-453913825",
      "name": "example.net",
      "target": "ns-123.awsdns-10.net",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "1553601564",
      "name": "example.net",
      "target": "ns-4444.awsdns-47.co.uk",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "1653797661",
      "name": "example.net",
      "target": "ns-7890.awsdns-61.org",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "706414698",
      "name": "example.net",
      "target": "ns-123.awsdns-44.com",
      "isAlias": false,
      "type": "NS"
    },
    {
      "id": "337271745",
      "name": "example.net",
```

```
        "target": "ns-4444.awsdns-47.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
        "isAlias": false,
        "type": "SOA"
    },
    {
        "id": "-1785431096",
        "name": "www.example.net",
        "target": "192.0.2.2",
        "isAlias": false,
        "type": "A"
    }
]
},
{
    "name": "example.org",
    "arn": "arn:aws:lightsail:global:111122223333:Domain/
f0f13ba3-3df0-4fdc-8ebb-1EXAMPLEf26e",
    "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEAF038",
    "createdAt": 1556661199.106,
    "location": {
        "availabilityZone": "all",
        "regionName": "global"
    },
    "resourceType": "Domain",
    "tags": [],
    "domainEntries": [
        {
            "id": "2065301345",
            "name": "example.org",
            "target": "192.0.2.4",
            "isAlias": false,
            "type": "A"
        },
        {
            "id": "-447198516",
            "name": "example.org",
            "target": "ns-123.awsdns-45.com",
            "isAlias": false,
            "type": "NS"
        },
        {
            "id": "136463022",
            "name": "example.org",
```

```

        "target": "ns-9999.awsdns-15.co.uk",
        "isAlias": false,
        "type": "NS"
    },
    {
        "id": "1395941679",
        "name": "example.org",
        "target": "ns-555.awsdns-01.net",
        "isAlias": false,
        "type": "NS"
    },
    {
        "id": "872052569",
        "name": "example.org",
        "target": "ns-6543.awsdns-38.org",
        "isAlias": false,
        "type": "NS"
    },
    {
        "id": "1001949377",
        "name": "example.org",
        "target": "ns-1234.awsdns-15.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
        "isAlias": false,
        "type": "SOA"
    },
    {
        "id": "1046191192",
        "name": "www.example.org",
        "target": "192.0.2.4",
        "isAlias": false,
        "type": "A"
    }
]
}
]
}

```

- Untuk detail API, lihat [GetDomains](#) di Referensi AWS CLI Perintah.

## get-export-snapshot-record

Contoh kode berikut menunjukkan cara menggunakan `get-export-snapshot-record`.

## AWS CLI

Untuk mendapatkan catatan snapshot yang diekspor ke Amazon EC2

`get-export-snapshot-record` Contoh berikut menampilkan detail tentang instans Amazon Lightsail atau snapshot disk yang diekspor ke Amazon EC2.

```
aws lightsail get-export-snapshot-records
```

Output:

```
{
  "exportSnapshotRecords": [
    {
      "name": "ExportSnapshotRecord-d2da10ce-0b3c-4ae1-ab3a-2EXAMPLEa586",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:ExportSnapshotRecord/076c7060-b0cc-4162-98f0-2EXAMPLEe28e",
      "createdAt": 1543534665.678,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "ExportSnapshotRecord",
      "state": "Succeeded",
      "sourceInfo": {
        "resourceType": "InstanceSnapshot",
        "createdAt": 1540339310.706,
        "name": "WordPress-512MB-0regon-1-1540339219",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/5446f534-ed60-4c17-b4a5-bEXAMPLEf8b7",
        "fromResourceName": "WordPress-512MB-0regon-1",
        "fromResourceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/4b8f1f24-e4d1-4cf3-88ff-cEXAMPLEa397",
        "instanceSnapshotInfo": {
          "fromBundleId": "nano_2_0",
          "fromBlueprintId": "wordpress_4_9_8",
          "fromDiskInfo": [
            {
              "path": "/dev/sda1",
              "sizeInGb": 20,
              "isSystemDisk": true
            }
          ]
        }
      }
    }
  ]
}
```



```

    }
  },
  "destinationInfo": {
    "id": "ami-0EXAMPLEc0d65058e",
    "service": "Aws::EC2::Image"
  }
},
{
  "name": "ExportSnapshotRecord-1c94e884-40ff-4fe1-9302-0EXAMPLE14c2",
  "arn": "arn:aws:lightsail:us-west-2:111122223333:ExportSnapshotRecord/
fb392ce8-6567-4013-9bfd-3EXAMPLE5b4c",
  "createdAt": 1543432110.2,
  "location": {
    "availabilityZone": "all",
    "regionName": "us-west-2"
  },
  "resourceType": "ExportSnapshotRecord",
  "state": "Succeeded",
  "sourceInfo": {
    "resourceType": "InstanceSnapshot",
    "createdAt": 1540833603.545,
    "name": "LAMP_PHP_5-512MB-0regon-1-1540833565",
    "arn": "arn:aws:lightsail:us-
west-2:111122223333:InstanceSnapshot/82334399-b5f2-49ec-8382-0EXAMPLEe45f",
    "fromResourceName": "LAMP_PHP_5-512MB-0regon-1",
    "fromResourceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/863b9f35-ab1e-4418-bdd2-1EXAMPLEebab2",
    "instanceSnapshotInfo": {
      "fromBundleId": "nano_2_0",
      "fromBlueprintId": "lamp_5_6_37_2",
      "fromDiskInfo": [
        {
          "path": "/dev/sda1",
          "sizeInGb": 20,
          "isSystemDisk": true
        }
      ]
    }
  }
},
  "destinationInfo": {
    "id": "ami-0EXAMPLE7c5ec84e2",
    "service": "Aws::EC2::Image"
  }
}

```

```
]
}
```

- Untuk detail API, lihat [GetExportSnapshotRecord](#) di Referensi AWS CLI Perintah.

## get-instance-access-details

Contoh kode berikut menunjukkan cara menggunakan `get-instance-access-details`.

### AWS CLI

Untuk mendapatkan informasi kunci host untuk sebuah instance

`get-instance-access-details` Contoh berikut menampilkan informasi kunci host misalnya `WordPress_Multisite-1`.

```
aws lightsail get-instance-access-details \
  --instance-name WordPress_Multisite-1
```

### Output:

```
{
  "accessDetails": {
    "certKey": "ssh-rsa-cert-v01@openssh.com
AEXAMPLEEaC1yc2EtY2VydC12MDFAb3B1bnZaC5jb20AAAAgNf076Dt3ppmPd0fPxZVMmS491aEAYYH9cHqAJ3fNML8
vEXAMPLE2eBWJyQvn7o1/
i0+s966h5sx8qUD791PB7q5UESd5VZGFtytrykfQJnjiwqe7EV5agzvjb1Lj26Fb37EKda9HVfCOu8pWbvky7Tyn9w29
+xMfQM9xVz0rXZmqx8uJidJpRgLCMTviofWQJU/
K1EXAMPLEAAAAAAAAABAAAALS00MzMzMdu4MzA40Dg1MTY2NjM40np6UW1ndHk4UE1RSG9STit0TG5QSEE9PQAAAAsAAA
+LiB+ozNbUA0cdNL9Y67x7qPv/R7XhTc21+2A+8+GuVpK/Kz9dqDMKNAEXAMPLE+YYN
+tiXm7Y80gziK+7iDB7xUuQ4vghmn4+qgz9mKwYgWvVe2+0XLuV7cnWPB7iU1HQg
+E3LUK1rV4ZFw9pj7X2dFdNKfMxwWgI1ISWKimEXAMPLEEehjrf1Rqc/
QH6TpWCvPfcx8uvwVqdwTfke/SfA5BCzbGGI1UmIUadh8nHcb5FamQ1hK7kECy47K/x9FMn/
KwmM7pCwJbSLDM07n9bnbvck6m8ZoB2N2YLMG5dW7BerEXAMPLEEobqfdtyYJHHe11EyyEJs1fWNU3D5JIG1gzcPAV
+Z1bQyUCZXf0os1Sa+HE85f0/
FRq9SVSBSHrmb0fr1PhgMzgSmqLeyhlbr6wwWIDbREXAMPLEJZ49H7RdQxdKyYrZPWvRgcr0qI2EL0tAajnpQQ8UZc
Aqter0xN5PhFL0J490WTacwCGRAjLhibAx7K1t/1ZXWo6c+ijq8c1l1327EXAMPLE/
e89GC89KcmKCxfGQniDAUgF8UqofIbq3Z0UgiAAYCVXclI4L68NhVXyoWuQXPBRQSEXAMPLEWm74tDL9tFN3c7tSe/
Oz0cTR+4sAAAIIPAAAAB3NzaC1yc2EAAAIAQnG/
L0DqiSnLrWhEox4aHqMgd0m0oLLAYx60QH9F0TM9EXAMPLE961rzSCMon7ZgsWNNL00wZQgDG
+rtJ4N0B7H0Vwns4ynUFbzNQ3qFGGeE31KwX1L41vV1iSy7sDk8aI0LmrKJi1LE1Qc1l8uboRlwoX0YEXAMPLEEaUCeX
+10+WEXAMPLEg6Y4U4ZvE2B3xyRdpvysb5TGFNTk5qPs1acnVkoL0GsZZXMPLGJnG40BpQLLtpj9sNMxAgZPCAujhkqk
```

```

+nx0904NUZ2pTwbVSUaV1gm6pug9xbwN01Im21t34JeLlKTqxcJ6zzS8W0c0KKpAm5c4hWkseMbyutS2jav/4hiS
+BhrYgptzfwe5qRXEXAMPLEEHZQr3YfGzYoBJ/
lLK3NHhx0ihhsfAYwMei0BFZT1F/7CT3IH4iitEkIgodio6/
Mw6UDqMPoZYQCK11EA6LFhYCOZG9drWcoRa74lM4kY9TP028Za8gDMh1WpkXLq9Gixon50HP8aM/
sEXAMPLEEr2+fnkw+1Bto05L6+VKoPlXaGqZ/fBYEXAMPLEAMQHjnLM1JYNvtEEPhp+TNzXHzuixWf/
Ht04m0AVpXrzIDXaS102tXY=",
    "ipAddress": "192.0.2.0",
    "privateKey": "-----BEGIN RSA PRIVATE KEY-----
\nEXAMPLEBAAKCAQEA+AD3qeU2toBy505v7wnRLVo/tngVickL5+6Jf4tPrPeuoebM
\nfKlA+/ZTwe6uVBENEVWRhbcra8pH0CZ44sKnuxFeWoM7425S49uhW9+xCnWvR1Xw
\njrvKvm75Mu08p/cNvfWugrBuaPB65DspgxNn0fZWMVxpIpSq0SPWmSwQHV597d6C
\nrEXAMPLEe08hJmqz2KFQ09X7fB2lBruGgr9aXiNPmWmovYKqWfmrnFvR7odFmDecq
\n5EXAMPLE9dyU1ZsrWhGby77eYrVaF10GNGQ8qy1HGUIScquZ9NDIL49n4mXbfsTH
\n0EXAMPLE12ZqsfLiYnSaUYCwjE74qH8ECVPytQIDAQABaoIBAHeZV9Z58JHAjifz
\nCEXAMPLEEqC3do0VDgXS1kKI92qNo4z2VcUEho878paCuVVXVHcCGgSnGeyIh2tN
\nMEXAMPLESohR427BhH3YLA+3Z5SIVnejbTgYPfLC37B8khTaYqkqMvdZiFVZK5qn
\nIEXAMPLEM93oF9eSZCjclKB/jGHsfb0eCDMP8BshHE2beuqzVMoK1Dx0nvoP3+Fp
\nAEXAMPLESq6pDpCo9YVUX8g1u3Ro9cPl2LXHDy+oVEY5KhbZQJ7VU1I72W0vppWW
\n0EXAMPLEkgY1q7p6qYtYcSgTEjz14gDiMfQ7SyHB3alkIoNONQ9ZPaWHyJvymeud
\noQTNuz0CgYEA/LFWNTEZrzdZdR1kJmyNRmAermU0B6utyNENChAlHGSHkB+1lVSh
\nbEXAMPLEQo9ooUeW5Ux03YwacZLoDT1mwxw1PtC1+PNycZoLe1fE9UdARrdmGTob
\n8l7CPLSXp3xuR8VqSp2fnIc7hfiQs/NrPX9gm/E0rB0we0RKYDSzWScGyEA+z/r
\niob+nJZq0Ybn0SuP6oMULP4vnWniWj8MIhUJU53LwSAM8DeJd0NKDdkui0d52aAL
\nVgn7nLo88rVWKhJwVc4tu/rNgZLcR3bP4+kL6zand0KQnMly0zNA2Ys26aa5udH1\nnqWl0WTt9WEm/
h10ndC1kn0MectrsvG17b38y5sMCgYEA54NiRGGz8oCPW6GN/FZA
\nKEXAMPLE5tw34GEH3UxlC9n3CejDaQmCz0ATwX4nIwRZDEqWyYZcS0btg1jhGiBD\nYEXAMPLEkc8Z71L/
agZEAaVCEog9FqfSqwB
+XTfoKh8qur74X1yCu9p6gof1q6k9\nneEXAMPLEechJcNN0g4ETIfMkCgYBdV0RRhE4mqvWp0dzA7v66FdEz2YSkjAXKk
\naEXAMPLE8Z/8yBSmuBv1Qv03XA12my462uB92uzzGAuW
+1yBc2Kn1sXqYTy0y1z0\nngEXAMPLEBogjw4MqHKL1bPKMHYQU8/
q24PaYgzHPzy13w1H6pTYf1Xq1HdE2D6Vv\nnyEXAMPLEegQC3i/
kVvhky/2XRwRVLC7J02Bg3QGTx38hpmDa5IuofKANjA+Wa3/zy\nnbEXAMPLE6ytQgD9GN/YtBq+uh0
+2ZkvXPL+CWRi0ZRxpPwYDBBFU9Cw0AuWWG1L8\nnwEXAMPLExM1cysRgcWB9RNgf3AuOpFd2i6XT/
riNsvvkpmJ+VooU8g==\n-----END RSA PRIVATE KEY-----\n",
    "protocol": "ssh",
    "instanceName": "WordPress_Multisite-1",
    "username": "bitnami",
    "hostKeys": [
        {
            "algorithm": "ssh-rsa",
            "publicKey":
                "AEXAMPLEaC1yc2EAAAADAQABAAQCoer9ieZTjQ3pXCHczuAYZFj1F7t
+uBkXuqeGMRex78pCvmS+DiEXAMPLEEuJ1Q8dcKhrQL4HpXbD9dosVCTaJnJwb4MQqsuSVFdHFzy3guP
+BKclWqtXJEXAMPLEsBGqZzlrIv6a9bTA0TCplZ8AD+hSRTaSXXqg6FT

```

```

+Qf16IktH0X1Ms7xIEXAMPLEmNtjCpzZiGXDHzytoMvUgwa8uHPp440g36EUu4VqQxoUHPJKoXvcQizyk3K8ym0hP0Tp
0t6y9HwvykEXAMPLEAfbKjBR42+u6+OSlkr4d339q2U1sTDytJhhs8HUel1wTfGRfp",
    "witnessedAt": 1570744377.699,
    "fingerprintSHA1": "SHA1:GEXAMPLEMoYgUg0ucadqU9Bt3Lk",
    "fingerprintSHA256": "SHA256:IEXAMPLEcB5vgxnAUoJawbdZ
+MwELhIp6FUxuwq/LIU"
    },
    {
        "algorithm": "ssh-ed25519",
        "publicKey":
        "AEXAMPLEeAC11ZDI1NTE5AAAAIC1gwGPDfGa0NxEXAMPLEJX3UNap781QxHQmn8nzlrUv",
        "witnessedAt": 1570744377.697,
        "fingerprintSHA1": "SHA1:VEXAMPLE5ReqSmTgv03sSUw9toU",
        "fingerprintSHA256": "SHA256:0EXAMPLEdE6tI95k3TJpG
+qhJbAoknB0yz9nAEaDt3A"
    },
    {
        "algorithm": "ecdsa-sha2-nistp256",
        "publicKey":
        "AEXAMPLEZHNhLXNoYTIItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABEXAMPLE9B4mZy8YSsZW7cixCDq5yHSAAxjJkDo5
+EnK1DCsYtUkxxEXAMPLE6V0WL2z63RTKa2AUPgd8irjxWI=",
        "witnessedAt": 1570744377.707,
        "fingerprintSHA1": "SHA1:UEXAMPLE0YCFxScf2G6tDg+7YG0",
        "fingerprintSHA256": "SHA256:wEXAMPLEQ9a/
iEXAMPLEhRufm6U9vFU4cpkMPHnBsNA"
    }
]
}
}
}

```

- Untuk detail API, lihat [GetInstanceAccessDetails](#) di Referensi AWS CLI Perintah.

## get-instance-metric-data

Contoh kode berikut menunjukkan cara menggunakan `get-instance-metric-data`.

### AWS CLI

Untuk mendapatkan data metrik untuk sebuah instance

`get-instance-metric-data` Contoh berikut mengembalikan persentase rata-rata CPU `Utilization` setiap 7200 detik (2 jam) antara 1571342400 dan 1571428800 misalnya `MEAN-1`.

Kami menyarankan Anda menggunakan konverter waktu unix untuk mengidentifikasi waktu mulai dan akhir.

```
aws lightsail get-instance-metric-data \  
  --instance-name MEAN-1 \  
  --metric-name CPUUtilization \  
  --period 7200 \  
  --start-time 1571342400 \  
  --end-time 1571428800 \  
  --unit Percent \  
  --statistics Average
```

Output:

```
{  
  "metricName": "CPUUtilization",  
  "metricData": [  
    {  
      "average": 0.26113718770120725,  
      "timestamp": 1571342400.0,  
      "unit": "Percent"  
    },  
    {  
      "average": 0.26861268928111953,  
      "timestamp": 1571392800.0,  
      "unit": "Percent"  
    },  
    {  
      "average": 0.28187475104748777,  
      "timestamp": 1571378400.0,  
      "unit": "Percent"  
    },  
    {  
      "average": 0.2651936960458352,  
      "timestamp": 1571421600.0,  
      "unit": "Percent"  
    },  
    {  
      "average": 0.2561856213712188,  
      "timestamp": 1571371200.0,  
      "unit": "Percent"  
    },  
    {
```

```
    "average": 0.3021383254607764,  
    "timestamp": 1571356800.0,  
    "unit": "Percent"  
  },  
  {  
    "average": 0.2618381649223539,  
    "timestamp": 1571407200.0,  
    "unit": "Percent"  
  },  
  {  
    "average": 0.26331929394825787,  
    "timestamp": 1571400000.0,  
    "unit": "Percent"  
  },  
  {  
    "average": 0.2576348407007818,  
    "timestamp": 1571385600.0,  
    "unit": "Percent"  
  },  
  {  
    "average": 0.2513008454658378,  
    "timestamp": 1571364000.0,  
    "unit": "Percent"  
  },  
  {  
    "average": 0.26329974562758346,  
    "timestamp": 1571414400.0,  
    "unit": "Percent"  
  },  
  {  
    "average": 0.2667092536656445,  
    "timestamp": 1571349600.0,  
    "unit": "Percent"  
  }  
]  
}
```

- Untuk detail API, lihat [GetInstanceMetricData](#) di Referensi AWS CLI Perintah.

## get-instance-port-states

Contoh kode berikut menunjukkan cara menggunakan `get-instance-port-states`.

## AWS CLI

Untuk mendapatkan informasi firewall untuk sebuah contoh

`get-instance-port-states` Contoh berikut mengembalikan port firewall dikonfigurasi misalnya `MEAN-1`.

```
aws lightsail get-instance-port-states \  
  --instance-name MEAN-1
```

Output:

```
{  
  "portStates": [  
    {  
      "fromPort": 80,  
      "toPort": 80,  
      "protocol": "tcp",  
      "state": "open"  
    },  
    {  
      "fromPort": 22,  
      "toPort": 22,  
      "protocol": "tcp",  
      "state": "open"  
    },  
    {  
      "fromPort": 443,  
      "toPort": 443,  
      "protocol": "tcp",  
      "state": "open"  
    }  
  ]  
}
```

- Untuk detail API, lihat [GetInstancePortStates](#) di Referensi AWS CLI Perintah.

## `get-instance-snapshot`

Contoh kode berikut menunjukkan cara menggunakan `get-instance-snapshot`.

## AWS CLI

Untuk mendapatkan informasi tentang snapshot instance tertentu

`get-instance-snapshot` Contoh berikut menampilkan rincian tentang snapshot contoh yang ditentukan.

```
aws lightsail get-instance-snapshot \  
  --instance-snapshot-name MEAN-1-1571419854
```

Output:

```
{  
  "instanceSnapshot": {  
    "name": "MEAN-1-1571419854",  
    "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/  
ac54700c-48a8-40fd-b065-2EXAMPLEac8f",  
    "supportCode": "6EXAMPLE3362/ami-0EXAMPLE67a73020d",  
    "createdAt": 1571419891.927,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "InstanceSnapshot",  
    "tags": [],  
    "state": "available",  
    "fromAttachedDisks": [],  
    "fromInstanceName": "MEAN-1",  
    "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/  
bd470fc5-a68b-44c5-8dbc-8EXAMPLEbada",  
    "fromBlueprintId": "mean_4_0_9",  
    "fromBundleId": "medium_2_0",  
    "isFromAutoSnapshot": false,  
    "sizeInGb": 80  
  }  
}
```

- Untuk detail API, lihat [GetInstanceSnapshot](#) di Referensi AWS CLI Perintah.

## `get-instance-snapshots`

Contoh kode berikut menunjukkan cara menggunakan `get-instance-snapshots`.



## AWS CLI

Untuk mendapatkan informasi tentang semua snapshot instans Anda

`get-instance-snapshots` Contoh berikut menampilkan detail tentang semua snapshot instance di AWS Wilayah yang dikonfigurasi.

```
aws lightsail get-instance-snapshots
```

Output:

```
{
  "instanceSnapshots": [
    {
      "name": "MEAN-1-1571421498",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/a20e6ebe-b0ee-4ae4-a750-3EXAMPLEcb0c",
      "supportCode": "6EXAMPLE3362/ami-0EXAMPLEe33cabfa1",
      "createdAt": 1571421527.755,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "InstanceSnapshot",
      "tags": [
        {
          "key": "no_delete"
        }
      ],
      "state": "available",
      "fromAttachedDisks": [],
      "fromInstanceName": "MEAN-1",
      "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/1761aa0a-6038-4f25-8b94-2EXAMPLE19fd",
      "fromBlueprintId": "wordpress_5_1_1_2",
      "fromBundleId": "micro_2_0",
      "isFromAutoSnapshot": false,
      "sizeInGb": 40
    },
    {
      "name": "MEAN-1-1571419854",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/ac54700c-48a8-40fd-b065-2EXAMPLEac8f",

```

```

    "supportCode": "6EXAMPLE3362/ami-0EXAMPLE67a73020d",
    "createdAt": 1571419891.927,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "InstanceSnapshot",
    "tags": [],
    "state": "available",
    "fromAttachedDisks": [],
    "fromInstanceName": "MEAN-1",
    "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/
bd470fc5-a68b-44c5-8dbc-8EXAMPLEbada",
    "fromBlueprintId": "mean_4_0_9",
    "fromBundleId": "medium_2_0",
    "isFromAutoSnapshot": false,
    "sizeInGb": 80
  }
]
}

```

- Untuk detail API, lihat [GetInstanceSnapshots](#) di Referensi AWS CLI Perintah.

## get-instance-state

Contoh kode berikut menunjukkan cara menggunakan `get-instance-state`.

### AWS CLI

Untuk mendapatkan informasi tentang keadaan sebuah instance

`get-instance-state` Contoh berikut mengembalikan keadaan contoh yang ditentukan.

```
aws lightsail get-instance-state \
  --instance-name MEAN-1
```

Output:

```
{
  "state": {
    "code": 16,
    "name": "running"
  }
}
```

```
}
```

- Untuk detail API, lihat [GetInstanceState](#) di Referensi AWS CLI Perintah.

## get-instance

Contoh kode berikut menunjukkan cara menggunakan `get-instance`.

### AWS CLI

Untuk mendapatkan informasi tentang sebuah instance

`get-instance` Contoh berikut menampilkan rincian tentang instance `MEAN-1`.

```
aws lightsail get-instance \  
  --instance-name MEAN-1
```

Output:

```
{  
  "instance": {  
    "name": "MEAN-1",  
    "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/bd470fc5-  
a68b-44c5-8dbc-EXAMPLE4bada",  
    "supportCode": "6EXAMPLE3362/i-05EXAMPLE407c97d3",  
    "createdAt": 1570635023.124,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "Instance",  
    "tags": [],  
    "blueprintId": "mean_4_0_9",  
    "blueprintName": "MEAN",  
    "bundleId": "medium_2_0",  
    "isStaticIp": false,  
    "privateIpAddress": "192.0.2.0",  
    "publicIpAddress": "192.0.2.0",  
    "hardware": {  
      "cpuCount": 2,  
      "disks": [  
        {  
          "createdAt": 1570635023.124,
```

```
        "sizeInGb": 80,
        "isSystemDisk": true,
        "iops": 240,
        "path": "/dev/sda1",
        "attachedTo": "MEAN-1",
        "attachmentState": "attached"
    }
],
"ramSizeInGb": 4.0
},
"networking": {
    "monthlyTransfer": {
        "gbPerMonthAllocated": 4096
    },
    "ports": [
        {
            "fromPort": 80,
            "toPort": 80,
            "protocol": "tcp",
            "accessFrom": "Anywhere (0.0.0.0/0)",
            "accessType": "public",
            "commonName": "",
            "accessDirection": "inbound"
        },
        {
            "fromPort": 22,
            "toPort": 22,
            "protocol": "tcp",
            "accessFrom": "Anywhere (0.0.0.0/0)",
            "accessType": "public",
            "commonName": "",
            "accessDirection": "inbound"
        },
        {
            "fromPort": 443,
            "toPort": 443,
            "protocol": "tcp",
            "accessFrom": "Anywhere (0.0.0.0/0)",
            "accessType": "public",
            "commonName": "",
            "accessDirection": "inbound"
        }
    ]
},
],
},
```

```
    "state": {
      "code": 16,
      "name": "running"
    },
    "username": "bitnami",
    "sshKeyName": "MyKey"
  }
}
```

- Untuk detail API, lihat [GetInstance](#) di Referensi AWS CLI Perintah.

## get-instances

Contoh kode berikut menunjukkan cara menggunakan `get-instances`.

### AWS CLI

Untuk mendapatkan informasi tentang semua contoh

`get-instances` Contoh berikut menampilkan rincian tentang semua instance di AWS Wilayah dikonfigurasi.

```
aws lightsail get-instances
```

Output:

```
{
  "instances": [
    {
      "name": "Windows_Server_2016-1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/0f44fbb9-8f55-4e47-a25e-EXAMPLE04763",
      "supportCode": "62EXAMPLE362/i-0bEXAMPLE71a686b9",
      "createdAt": 1571332358.665,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "resourceType": "Instance",
      "tags": [],
      "blueprintId": "windows_server_2016",
      "blueprintName": "Windows Server 2016",
    }
  ]
}
```

```
"bundleId": "small_win_2_0",
"isStaticIp": false,
"privateIpAddress": "192.0.2.0",
"publicIpAddress": "192.0.2.0",
"hardware": {
  "cpuCount": 1,
  "disks": [
    {
      "createdAt": 1571332358.665,
      "sizeInGb": 60,
      "isSystemDisk": true,
      "iops": 180,
      "path": "/dev/sda1",
      "attachedTo": "Windows_Server_2016-1",
      "attachmentState": "attached"
    },
    {
      "name": "my-disk-for-windows-server",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/4123a81c-484c-49ea-afea-5EXAMPLEda87",
      "supportCode": "6EXAMPLE3362/vol-0EXAMPLEb2b99ca3d",
      "createdAt": 1571355063.494,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "resourceType": "Disk",
      "tags": [],
      "sizeInGb": 128,
      "isSystemDisk": false,
      "iops": 384,
      "path": "/dev/xvdf",
      "state": "in-use",
      "attachedTo": "Windows_Server_2016-1",
      "isAttached": true,
      "attachmentState": "attached"
    }
  ],
  "ramSizeInGb": 2.0
},
"networking": {
  "monthlyTransfer": {
    "gbPerMonthAllocated": 3072
  }
},
```

```
    "ports": [
      {
        "fromPort": 80,
        "toPort": 80,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
      },
      {
        "fromPort": 22,
        "toPort": 22,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
      },
      {
        "fromPort": 3389,
        "toPort": 3389,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
      }
    ]
  },
  "state": {
    "code": 16,
    "name": "running"
  },
  "username": "Administrator",
  "sshKeyName": "LightsailDefaultKeyPair"
},
{
  "name": "MEAN-1",
  "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/bd470fc5-
a68b-44c5-8dbc-8EXAMPLEbada",
  "supportCode": "6EXAMPLE3362/i-0EXAMPLEa407c97d3",
  "createdAt": 1570635023.124,
  "location": {
```

```
    "availabilityZone": "us-west-2a",
    "regionName": "us-west-2"
  },
  "resourceType": "Instance",
  "tags": [],
  "blueprintId": "mean_4_0_9",
  "blueprintName": "MEAN",
  "bundleId": "medium_2_0",
  "isStaticIp": false,
  "privateIpAddress": "192.0.2.0",
  "publicIpAddress": "192.0.2.0",
  "hardware": {
    "cpuCount": 2,
    "disks": [
      {
        "name": "Disk-1",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
        "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
        "createdAt": 1566585439.587,
        "location": {
          "availabilityZone": "us-west-2a",
          "regionName": "us-west-2"
        },
        "resourceType": "Disk",
        "tags": [
          {
            "key": "test"
          }
        ],
        "sizeInGb": 8,
        "isSystemDisk": false,
        "iops": 100,
        "path": "/dev/xvdf",
        "state": "in-use",
        "attachedTo": "MEAN-1",
        "isAttached": true,
        "attachmentState": "attached"
      },
      {
        "createdAt": 1570635023.124,
        "sizeInGb": 80,
        "isSystemDisk": true,
        "iops": 240,
```



```
        "path": "/dev/sda1",
        "attachedTo": "MEAN-1",
        "attachmentState": "attached"
    }
  ],
  "ramSizeInGb": 4.0
},
"networking": {
  "monthlyTransfer": {
    "gbPerMonthAllocated": 4096
  },
  "ports": [
    {
      "fromPort": 80,
      "toPort": 80,
      "protocol": "tcp",
      "accessFrom": "Anywhere (0.0.0.0/0)",
      "accessType": "public",
      "commonName": "",
      "accessDirection": "inbound"
    },
    {
      "fromPort": 22,
      "toPort": 22,
      "protocol": "tcp",
      "accessFrom": "Anywhere (0.0.0.0/0)",
      "accessType": "public",
      "commonName": "",
      "accessDirection": "inbound"
    },
    {
      "fromPort": 443,
      "toPort": 443,
      "protocol": "tcp",
      "accessFrom": "Anywhere (0.0.0.0/0)",
      "accessType": "public",
      "commonName": "",
      "accessDirection": "inbound"
    }
  ]
},
"state": {
  "code": 16,
  "name": "running"
}
```

```
    },
    "username": "bitnami",
    "sshKeyName": "MyTestKey"
  }
]
}
```

- Untuk detail API, lihat [GetInstance](#) di Referensi AWS CLI Perintah.

## get-key-pair

Contoh kode berikut menunjukkan cara menggunakan `get-key-pair`.

### AWS CLI

Untuk mendapatkan informasi tentang key pair

`get-key-pair` Contoh berikut menampilkan rincian tentang key pair yang ditentukan.

```
aws lightsail get-key-pair \
  --key-pair-name MyKey1
```

Output:

```
{
  "keyPair": {
    "name": "MyKey1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/19a4efdf-3054-43d6-91fd-eEXAMPLE21bf",
    "supportCode": "6EXAMPLE3362/MyKey1",
    "createdAt": 1571255026.975,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "KeyPair",
    "tags": [],
    "fingerprint": "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff:gg:hh:ii:jj"
  }
}
```

- Untuk detail API, lihat [GetKeyPair](#) di Referensi AWS CLI Perintah.

## get-key-pairs

Contoh kode berikut menunjukkan cara menggunakan `get-key-pairs`.

### AWS CLI

Untuk mendapatkan informasi tentang semua pasangan kunci

`get-key-pairs` Contoh berikut menampilkan rincian tentang semua pasangan kunci di AWS Wilayah dikonfigurasi.

```
aws lightsail get-key-pairs
```

Output:

```
{
  "keyPairs": [
    {
      "name": "MyKey1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/19a4efdf-3054-43d6-91fd-eEXAMPLE21bf",
      "supportCode": "6EXAMPLE3362/MyKey1",
      "createdAt": 1571255026.975,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "KeyPair",
      "tags": [],
      "fingerprint":
      "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff:gg:hh:ii:jj"
    }
  ]
}
```

- Untuk detail API, lihat [GetKeyPairs](#) di Referensi AWS CLI Perintah.

## get-load-balancer-tls-certificates

Contoh kode berikut menunjukkan cara menggunakan `get-load-balancer-tls-certificates`.

## AWS CLI

Untuk mendapatkan informasi tentang sertifikat TLS untuk penyeimbang beban

`get-load-balancer-tls-certificates` Contoh berikut menampilkan rincian tentang sertifikat TLS untuk penyeimbang beban tertentu.

```
aws lightsail get-load-balancer-tls-certificates \  
  --load-balancer-name LoadBalancer-1
```

Output:

```
{  
  "tlsCertificates": [  
    {  
      "name": "example-com",  
      "arn": "arn:aws:lightsail:us-  
west-2:111122223333:LoadBalancerTlsCertificate/d7bf4643-6a02-4cd4-b3c4-  
fEXAMPLE9b4d",  
      "supportCode": "6EXAMPLE3362/arn:aws:acm:us-  
west-2:333322221111:certificate/9af8e32c-a54e-4a67-8c63-cEXAMPLEb314",  
      "createdAt": 1571678025.3,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "resourceType": "LoadBalancerTlsCertificate",  
      "loadBalancerName": "LoadBalancer-1",  
      "isAttached": false,  
      "status": "ISSUED",  
      "domainName": "example.com",  
      "domainValidationRecords": [  
        {  
          "name": "_dEXAMPLE4ede046a0319eb44a4eb3cbc.example.com.",  
          "type": "CNAME",  
          "value": "_bEXAMPLE0899fb7b6bf79d9741d1a383.hkvuiqjoua.acm-  
validations.aws.",  
          "validationStatus": "SUCCESS",  
          "domainName": "example.com"  
        }  
      ],  
      "issuedAt": 1571678070.0,  
      "issuer": "Amazon",
```

```

        "keyAlgorithm": "RSA-2048",
        "notAfter": 1605960000.0,
        "notBefore": 1571616000.0,
        "serial": "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff",
        "signatureAlgorithm": "SHA256WITHRSA",
        "subject": "CN=example.com",
        "subjectAlternativeNames": [
            "example.com"
        ]
    }
]
}

```

- Untuk detail API, lihat [GetLoadBalancerTlsCertificates](#) di Referensi AWS CLI Perintah.

## get-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `get-load-balancer`.

### AWS CLI

Untuk mendapatkan informasi tentang penyeimbang beban

`get-load-balancer` Contoh berikut menampilkan rincian tentang penyeimbang beban yang ditentukan.

```
aws lightsail get-load-balancer \
  --load-balancer-name LoadBalancer-1
```

Output:

```

{
  "loadBalancer": {
    "name": "LoadBalancer-1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:LoadBalancer/40486b2b-1ad0-4152-83e4-cEXAMPLE6f4b",
    "supportCode": "6EXAMPLE3362/arn:aws:elasticloadbalancing:us-west-2:333322221111:loadbalancer/app/bEXAMPLE128cb59d86f946a9395dd304/1EXAMPLE8dd9d77e",
    "createdAt": 1571677906.723,
    "location": {

```

```
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "resourceType": "LoadBalancer",
    "tags": [],
    "dnsName": "bEXAMPLE128cb59d86f946a9395dd304-1486911371.us-
west-2.elb.amazonaws.com",
    "state": "active",
    "protocol": "HTTP",
    "publicPorts": [
        80
    ],
    "healthCheckPath": "/",
    "instancePort": 80,
    "instanceHealthSummary": [
        {
            "instanceName": "MEAN-3",
            "instanceHealth": "healthy"
        },
        {
            "instanceName": "MEAN-1",
            "instanceHealth": "healthy"
        },
        {
            "instanceName": "MEAN-2",
            "instanceHealth": "healthy"
        }
    ],
    "tlsCertificateSummaries": [
        {
            "name": "example-com",
            "isAttached": false
        }
    ],
    "configurationOptions": {
        "SessionStickinessEnabled": "false",
        "SessionStickiness_LB_CookieDurationSeconds": "86400"
    }
}
}
```

- Untuk detail API, lihat [GetLoadBalancer](#) di Referensi AWS CLI Perintah.

## get-load-balancers

Contoh kode berikut menunjukkan cara menggunakan `get-load-balancers`.

### AWS CLI

Untuk mendapatkan informasi tentang semua penyeimbang beban

`get-load-balancers` Contoh berikut menampilkan detail tentang semua penyeimbang beban di Wilayah yang dikonfigurasi AWS .

```
aws lightsail get-load-balancers
```

Output:

```
{
  "loadBalancers": [
    {
      "name": "LoadBalancer-1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:LoadBalancer/40486b2b-1ad0-4152-83e4-cEXAMPLE6f4b",
      "supportCode": "6EXAMPLE3362/arn:aws:elasticloadbalancing:us-west-2:333322221111:loadbalancer/app/bEXAMPLE128cb59d86f946a9395dd304/1EXAMPLE8dd9d77e",
      "createdAt": 1571677906.723,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "LoadBalancer",
      "tags": [],
      "dnsName": "bEXAMPLE128cb59d86f946a9395dd304-1486911371.us-west-2.elb.amazonaws.com",
      "state": "active",
      "protocol": "HTTP",
      "publicPorts": [
        80
      ],
      "healthCheckPath": "/",
      "instancePort": 80,
      "instanceHealthSummary": [
        {
          "instanceName": "MEAN-3",
```

```

        "instanceHealth": "healthy"
      },
      {
        "instanceName": "MEAN-1",
        "instanceHealth": "healthy"
      },
      {
        "instanceName": "MEAN-2",
        "instanceHealth": "healthy"
      }
    ],
    "tlsCertificateSummaries": [
      {
        "name": "example-com",
        "isAttached": false
      }
    ],
    "configurationOptions": {
      "SessionStickinessEnabled": "false",
      "SessionStickiness_LB_CookieDurationSeconds": "86400"
    }
  }
]
}

```

- Untuk detail API, lihat [GetLoadBalancers](#) di Referensi AWS CLI Perintah.

## get-operation

Contoh kode berikut menunjukkan cara menggunakan `get-operation`.

### AWS CLI

Untuk mendapatkan informasi tentang satu operasi

`get-operation` Contoh berikut menampilkan rincian tentang operasi yang ditentukan.

```
aws lightsail get-operation \
  --operation-id e5700e8a-daf2-4b49-bc01-3EXAMPLE910a
```

Output:

```
{
```



```
"operation": {
  "id": "e5700e8a-daf2-4b49-bc01-3EXAMPLE910a",
  "resourceName": "Instance-1",
  "resourceType": "Instance",
  "createdAt": 1571679872.404,
  "location": {
    "availabilityZone": "us-west-2a",
    "regionName": "us-west-2"
  },
  "isTerminal": true,
  "operationType": "CreateInstance",
  "status": "Succeeded",
  "statusChangedAt": 1571679890.304
}
```

- Untuk detail API, lihat [GetOperation](#) di Referensi AWS CLI Perintah.

## get-operations-for-resource

Contoh kode berikut menunjukkan cara menggunakan `get-operations-for-resource`.

### AWS CLI

Untuk mendapatkan semua operasi untuk sumber daya

`get-operations-for-resource` Contoh berikut menampilkan rincian tentang semua operasi untuk sumber daya yang ditentukan.

```
aws lightsail get-operations-for-resource \
  --resource-name LoadBalancer-1
```

Output:

```
{
  "operations": [
    {
      "id": "e2973046-43f8-4252-a4b4-9EXAMPLE69ce",
      "resourceName": "LoadBalancer-1",
      "resourceType": "LoadBalancer",
      "createdAt": 1571678786.071,
      "location": {
```

```
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "MEAN-1",
    "operationType": "DetachInstancesFromLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571679087.57
},
{
    "id": "2d742a18-0e7f-48c8-9705-3EXAMPLEf98a",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571678782.784,
    "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "MEAN-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571678798.465
},
{
    "id": "6c700fcc-4246-40ab-952b-1EXAMPLEdac2",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571678775.297,
    "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "MEAN-3",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571678842.806
},
...
}
]
}
```

- Untuk detail API, lihat [GetOperationsForResource](#) di Referensi AWS CLI Perintah.

## get-operations

Contoh kode berikut menunjukkan cara menggunakan `get-operations`.

### AWS CLI

Untuk mendapatkan informasi tentang semua operasi

`get-operations` Contoh berikut menampilkan rincian tentang semua operasi di AWS Wilayah dikonfigurasi.

```
aws lightsail get-operations
```

Output:

```
{
  "operations": [
    {
      "id": "e5700e8a-daf2-4b49-bc01-3EXAMPLE910a",
      "resourceName": "Instance-1",
      "resourceType": "Instance",
      "createdAt": 1571679872.404,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "CreateInstance",
      "status": "Succeeded",
      "statusChangedAt": 1571679890.304
    },
    {
      "id": "701a3339-930e-4914-a9f9-7EXAMPLE68d7",
      "resourceName": "WordPress-1",
      "resourceType": "Instance",
      "createdAt": 1571678786.072,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
    },
  ],
}
```

```

        "isTerminal": true,
        "operationDetails": "LoadBalancer-1",
        "operationType": "DetachInstancesFromLoadBalancer",
        "status": "Succeeded",
        "statusChangedAt": 1571679086.399
    },
    {
        "id": "e2973046-43f8-4252-a4b4-9EXAMPLE69ce",
        "resourceName": "LoadBalancer-1",
        "resourceType": "LoadBalancer",
        "createdAt": 1571678786.071,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "WordPress-1",
        "operationType": "DetachInstancesFromLoadBalancer",
        "status": "Succeeded",
        "statusChangedAt": 1571679087.57
    },
    ...
}
]
}

```

- Untuk detail API, lihat [GetOperations](#) di Referensi AWS CLI Perintah.

## get-regions

Contoh kode berikut menunjukkan cara menggunakan `get-regions`.

### AWS CLI

Untuk mendapatkan semua AWS Wilayah untuk Amazon Lightsail

`get-regions` Contoh berikut menampilkan detail tentang semua AWS Wilayah untuk Amazon Lightsail.

```
aws lightsail get-regions
```

Output:

```

{
  "regions": [
    {
      "continentCode": "NA",
      "description": "This region is recommended to serve users in the eastern
United States",
      "displayName": "Virginia",
      "name": "us-east-1",
      "availabilityZones": [],
      "relationalDatabaseAvailabilityZones": []
    },
    {
      "continentCode": "NA",
      "description": "This region is recommended to serve users in the eastern
United States",
      "displayName": "Ohio",
      "name": "us-east-2",
      "availabilityZones": [],
      "relationalDatabaseAvailabilityZones": []
    },
    {
      "continentCode": "NA",
      "description": "This region is recommended to serve users in the
northwestern United States, Alaska, and western Canada",
      "displayName": "Oregon",
      "name": "us-west-2",
      "availabilityZones": [],
      "relationalDatabaseAvailabilityZones": []
    },
    ...
  ]
}

```

- Untuk detail API, lihat [GetRegions](#) di Referensi AWS CLI Perintah.

## get-relational-database-blueprints

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-blueprints`.

### AWS CLI

Untuk mendapatkan cetak biru untuk database relasional baru

`get-relational-database-blueprints` Contoh berikut menampilkan detail tentang semua cetak biru database relasional yang tersedia yang dapat digunakan untuk membuat database relasional baru di Amazon Lightsail.

```
aws lightsail get-relational-database-blueprints
```

Output:

```
{
  "blueprints": [
    {
      "blueprintId": "mysql_5_6",
      "engine": "mysql",
      "engineVersion": "5.6.44",
      "engineDescription": "MySQL Community Edition",
      "engineVersionDescription": "MySQL 5.6.44",
      "isEngineDefault": false
    },
    {
      "blueprintId": "mysql_5_7",
      "engine": "mysql",
      "engineVersion": "5.7.26",
      "engineDescription": "MySQL Community Edition",
      "engineVersionDescription": "MySQL 5.7.26",
      "isEngineDefault": true
    },
    {
      "blueprintId": "mysql_8_0",
      "engine": "mysql",
      "engineVersion": "8.0.16",
      "engineDescription": "MySQL Community Edition",
      "engineVersionDescription": "MySQL 8.0.16",
      "isEngineDefault": false
    },
    {
      "blueprintId": "postgres_9_6",
      "engine": "postgres",
      "engineVersion": "9.6.15",
      "engineDescription": "PostgreSQL",
      "engineVersionDescription": "PostgreSQL 9.6.15-R1",
      "isEngineDefault": false
    },
    {
```

```
    "blueprintId": "postgres_10",
    "engine": "postgres",
    "engineVersion": "10.10",
    "engineDescription": "PostgreSQL",
    "engineVersionDescription": "PostgreSQL 10.10-R1",
    "isEngineDefault": false
  },
  {
    "blueprintId": "postgres_11",
    "engine": "postgres",
    "engineVersion": "11.5",
    "engineDescription": "PostgreSQL",
    "engineVersionDescription": "PostgreSQL 11.5-R1",
    "isEngineDefault": true
  }
]
```

- Untuk detail API, lihat [GetRelationalDatabaseBlueprints](#) di Referensi AWS CLI Perintah.

## get-relational-database-bundles

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-bundles`.

### AWS CLI

Untuk mendapatkan bundel untuk database relasional baru

`get-relational-database-bundles` Contoh berikut menampilkan detail tentang semua bundel database relasional yang tersedia yang dapat digunakan untuk membuat database relasional baru di Amazon Lightsail. Perhatikan bahwa respon tidak termasuk bundel tidak aktif karena `--include-inactive` bendera tidak ditentukan dalam perintah. Anda tidak dapat menggunakan bundel yang tidak aktif untuk membuat database relasional baru.

```
aws lightsail get-relational-database-bundles
```

Output:

```
{
  "bundles": [
    {
      "bundleId": "micro_2_0",
```

```
    "name": "Micro",
    "price": 15.0,
    "ramSizeInGb": 1.0,
    "diskSizeInGb": 40,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "micro_ha_2_0",
    "name": "Micro with High Availability",
    "price": 30.0,
    "ramSizeInGb": 1.0,
    "diskSizeInGb": 40,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "small_2_0",
    "name": "Small",
    "price": 30.0,
    "ramSizeInGb": 2.0,
    "diskSizeInGb": 80,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "small_ha_2_0",
    "name": "Small with High Availability",
    "price": 60.0,
    "ramSizeInGb": 2.0,
    "diskSizeInGb": 80,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "medium_2_0",
```



```
    "name": "Medium",
    "price": 60.0,
    "ramSizeInGb": 4.0,
    "diskSizeInGb": 120,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "medium_ha_2_0",
    "name": "Medium with High Availability",
    "price": 120.0,
    "ramSizeInGb": 4.0,
    "diskSizeInGb": 120,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "large_2_0",
    "name": "Large",
    "price": 115.0,
    "ramSizeInGb": 8.0,
    "diskSizeInGb": 240,
    "transferPerMonthInGb": 200,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "large_ha_2_0",
    "name": "Large with High Availability",
    "price": 230.0,
    "ramSizeInGb": 8.0,
    "diskSizeInGb": 240,
    "transferPerMonthInGb": 200,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  }
]
```

```
}
```

Untuk informasi selengkapnya, lihat [Membuat database di Amazon Lightsail di Panduan Pengembang Amazon Lightsail](#).

- Untuk detail API, lihat [GetRelationalDatabaseBundles](#) di Referensi AWS CLI Perintah.

## get-relational-database-events

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-events`.

### AWS CLI

Untuk mendapatkan acara untuk database relasional

`get-relational-database-events` Contoh berikut menampilkan rincian tentang peristiwa dalam 17 jam terakhir (1020 menit) untuk database relasional yang ditentukan.

```
aws lightsail get-relational-database-events \
  --relational-database-name Database-1 \
  --duration-in-minutes 1020
```

Output:

```
{
  "relationalDatabaseEvents": [
    {
      "resource": "Database-1",
      "createdAt": 1571654146.553,
      "message": "Backing up Relational Database",
      "eventCategories": [
        "backup"
      ]
    },
    {
      "resource": "Database-1",
      "createdAt": 1571654249.98,
      "message": "Finished Relational Database backup",
      "eventCategories": [
        "backup"
      ]
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [GetRelationalDatabaseEvents](#) di Referensi AWS CLI Perintah.

## get-relational-database-log-events

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-log-events`.

### AWS CLI

Untuk mendapatkan peristiwa log untuk database relasional

`get-relational-database-log-events` Contoh berikut menampilkan rincian tentang log yang ditentukan antara 1570733176 dan 1571597176 untuk database Database1 relasional. Informasi yang dikembalikan dikonfigurasi untuk memulai `head`.

Kami menyarankan Anda menggunakan konverter waktu unix untuk mengidentifikasi waktu mulai dan akhir.

```
aws lightsail get-relational-database-log-events \
  --relational-database-name Database1 \
  --log-stream-name error \
  --start-from-head \
  --start-time 1570733176 \
  --end-time 1571597176
```

### Output:

```
{
  "resourceLogEvents": [
    {
      "createdAt": 1570820267.0,
      "message": "2019-10-11 18:57:47 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Name or service not known"
    },
    {
      "createdAt": 1570860974.0,
      "message": "2019-10-12 06:16:14 20969 [Warning] IP address '8192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860977.0,
```

```

      "message": "2019-10-12 06:16:17 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860979.0,
      "message": "2019-10-12 06:16:19 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860981.0,
      "message": "2019-10-12 06:16:21 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860982.0,
      "message": "2019-10-12 06:16:22 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860984.0,
      "message": "2019-10-12 06:16:24 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860986.0,
      "message": "2019-10-12 06:16:26 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    ...
  ],
  "nextBackwardToken":
  "eEXAMPLEZXJUZXh0IjoiZnRwb3F3cUpRS1Q5NndMYThxe1RUZ1FhR3J6c2dKWEEvM2kvajZMzVzVzVWpqRDN0YjFXTj
  "nextForwardToken":
  "eEXAMPLEZXJUZXh0IjoiT09Lb0Z6ZFRJbHhaNEQ5N2tPbkkwRmwwNUxPZjFTbFFwUk1Qbz1SaWgvMwVXbEk4aG56VH
}

```

- Untuk detail API, lihat [GetRelationalDatabaseLogEvents](#) di Referensi AWS CLI Perintah.

## get-relational-database-log-streams

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-log-streams`.

## AWS CLI

Untuk mendapatkan aliran log untuk database relasional

`get-relational-database-log-streams` Contoh berikut mengembalikan semua aliran log yang tersedia untuk database relasional tertentu.

```
aws lightsail get-relational-database-log-streams \  
--relational-database-name Database1
```

Output:

```
{  
  "logStreams": [  
    "audit",  
    "error",  
    "general",  
    "slowquery"  
  ]  
}
```

- Untuk detail API, lihat [GetRelationalDatabaseLogStreams](#) di Referensi AWS CLI Perintah.

## `get-relational-database-master-user-password`

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-master-user-password`.

## AWS CLI

Untuk mendapatkan kata sandi pengguna utama untuk database relasional

`get-relational-database-master-user-password` Contoh berikut mengembalikan informasi tentang password pengguna master untuk database relasional tertentu.

```
aws lightsail get-relational-database-master-user-password \  
--relational-database-name Database-1
```

Output:

```
{  
  "masterUserPassword": "VEXAMPLEec.9qvx, _t<)Wkf)kwboM, >2",
```

```
"createdAt": 1571259453.959
}
```

- Untuk detail API, lihat [GetRelationalDatabaseMasterUserPassword](#) di Referensi AWS CLI Perintah.

## get-relational-database-metric-data

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-metric-data`.

### AWS CLI

Untuk mendapatkan data metrik untuk database relasional

`get-relational-database-metric-data` Contoh berikut mengembalikan jumlah hitungan metrik `DatabaseConnections` selama periode 24 jam (86400detik) antara 1570733176 dan 1571597176 untuk database `Database1` relasional.

Kami menyarankan Anda menggunakan konverter waktu unix untuk mengidentifikasi waktu mulai dan akhir.

```
aws lightsail get-relational-database-metric-data \
  --relational-database-name Database1 \
  --metric-name DatabaseConnections \
  --period 86400 \
  --start-time 1570733176 \
  --end-time 1571597176 \
  --unit Count \
  --statistics Sum
```

### Output:

```
{
  "metricName": "DatabaseConnections",
  "metricData": [
    {
      "sum": 1.0,
      "timestamp": 1571510760.0,
      "unit": "Count"
    },
    {
```

```
    "sum": 1.0,  
    "timestamp": 1570733160.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 1.0,  
    "timestamp": 1570992360.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 0.0,  
    "timestamp": 1571251560.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 721.0,  
    "timestamp": 1570819560.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 1.0,  
    "timestamp": 1571078760.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 2.0,  
    "timestamp": 1571337960.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 684.0,  
    "timestamp": 1570905960.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 0.0,  
    "timestamp": 1571165160.0,  
    "unit": "Count"  
  },  
  {  
    "sum": 1.0,  
    "timestamp": 1571424360.0,  
    "unit": "Count"  
  }  
}
```

```
]
}
```

- Untuk detail API, lihat [GetRelationalDatabaseMetricData](#) di Referensi AWS CLI Perintah.

## get-relational-database-parameters

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-parameters`.

### AWS CLI

Untuk mendapatkan parameter untuk database relasional

`get-relational-database-parameters` Contoh berikut mengembalikan informasi tentang semua parameter yang tersedia untuk database relasional tertentu.

```
aws lightsail get-relational-database-parameters \
  --relational-database-name Database-1
```

Output:

```
{
  "parameters": [
    {
      "allowedValues": "0,1",
      "applyMethod": "pending-reboot",
      "applyType": "dynamic",
      "dataType": "boolean",
      "description": "Automatically set all granted roles as active after the user has authenticated successfully.",
      "isModifiable": true,
      "parameterName": "activate_all_roles_on_login",
      "parameterValue": "0"
    },
    {
      "allowedValues": "0,1",
      "applyMethod": "pending-reboot",
      "applyType": "static",
      "dataType": "boolean",
      "description": "Controls whether user-defined functions that have only an xxx symbol for the main function can be loaded",
      "isModifiable": false,

```



```

        "parameterName": "allow-suspicious-udfs"
    },
    {
        "allowedValues": "0,1",
        "applyMethod": "pending-reboot",
        "applyType": "dynamic",
        "dataType": "boolean",
        "description": "Sets the autocommit mode",
        "isModifiable": true,
        "parameterName": "autocommit"
    },
    {
        "allowedValues": "0,1",
        "applyMethod": "pending-reboot",
        "applyType": "static",
        "dataType": "boolean",
        "description": "Controls whether the server autogenerates SSL key and
certificate files in the data directory, if they do not already exist.",
        "isModifiable": false,
        "parameterName": "auto_generate_certs"
    },
    ...
}
]
}

```

Untuk informasi selengkapnya, lihat [Memperbarui parameter database di Amazon Lightsail](#) dalam Panduan Pengembang Lightsail.

- Untuk detail API, lihat [GetRelationalDatabaseParameters](#) di Referensi AWS CLI Perintah.

## get-relational-database-snapshot

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-snapshot`.

### AWS CLI

Untuk mendapatkan informasi tentang snapshot database relasional

`get-relational-database-snapshot` Contoh berikut menampilkan rincian tentang snapshot database relasional yang ditentukan.

```
aws lightsail get-relational-database-snapshot \
```

```
--relational-database-snapshot-name Database-1-1571350042
```

### Output:

```
{
  "relationalDatabaseSnapshot": {
    "name": "Database-1-1571350042",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabaseSnapshot/0389bbad-4b85-4c3d-9EXAMPLEaee3643d2",
    "supportCode": "6EXAMPLE3362/1s-8EXAMPLE2ba7ad041451946fafc2ad19cfbd9eb2",
    "createdAt": 1571350046.238,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "RelationalDatabaseSnapshot",
    "tags": [],
    "engine": "mysql",
    "engineVersion": "8.0.16",
    "sizeInGb": 40,
    "state": "available",
    "fromRelationalDatabaseName": "Database-1",
    "fromRelationalDatabaseArn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
    "fromRelationalDatabaseBundleId": "micro_1_0",
    "fromRelationalDatabaseBlueprintId": "mysql_8_0"
  }
}
```

- Untuk detail API, lihat [GetRelationalDatabaseSnapshot](#) di Referensi AWS CLI Perintah.

## get-relational-database-snapshots

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database-snapshots`.

### AWS CLI

Untuk mendapatkan informasi tentang semua snapshot database relasional

`get-relational-database-snapshots` Contoh berikut menampilkan rincian tentang semua snapshot database relasional di Wilayah dikonfigurasi AWS .

```
aws lightsail get-relational-database-snapshots
```

**Output:**

```
{
  "relationalDatabaseSnapshots": [
    {
      "name": "Database-1-1571350042",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabaseSnapshot/0389bbad-4b85-4c3d-9861-6EXAMPLE43d2",
      "supportCode": "6EXAMPLE3362/1s-8EXAMPLE2ba7ad041451946fafc2ad19cfbd9eb2",
      "createdAt": 1571350046.238,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "RelationalDatabaseSnapshot",
      "tags": [],
      "engine": "mysql",
      "engineVersion": "8.0.16",
      "sizeInGb": 40,
      "state": "available",
      "fromRelationalDatabaseName": "Database-1",
      "fromRelationalDatabaseArn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
      "fromRelationalDatabaseBundleId": "micro_1_0",
      "fromRelationalDatabaseBlueprintId": "mysql_8_0"
    },
    {
      "name": "Database1-Console",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabaseSnapshot/8b94136e-06ec-4b1a-a3fb-5EXAMPLEEe1e9",
      "supportCode": "6EXAMPLE3362/1s-9EXAMPLE14b000d34c8d1c432734e137612d5b5c",
      "createdAt": 1571249981.025,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "RelationalDatabaseSnapshot",
      "tags": [
```

```

        {
            "key": "test"
        }
    ],
    "engine": "mysql",
    "engineVersion": "5.6.44",
    "sizeInGb": 40,
    "state": "available",
    "fromRelationalDatabaseName": "Database1",
    "fromRelationalDatabaseArn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/a6161cb7-4535-4f16-9dcf-8EXAMPLE3d4e",
    "fromRelationalDatabaseBundleId": "micro_1_0",
    "fromRelationalDatabaseBlueprintId": "mysql_5_6"
    }
]
}

```

- Untuk detail API, lihat [GetRelationalDatabaseSnapshots](#) di Referensi AWS CLI Perintah.

## get-relational-database

Contoh kode berikut menunjukkan cara menggunakan `get-relational-database`.

### AWS CLI

Untuk mendapatkan informasi tentang database relasional

`get-relational-database` Contoh berikut menampilkan rincian tentang database relasional tertentu.

```
aws lightsail get-relational-database \
  --relational-database-name Database-1
```

Output:

```

{
  "relationalDatabase": {
    "name": "Database-1",
    "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
    "supportCode": "6EXAMPLE3362/1s-9EXAMPLE8ad863723b62cc8901a8aa6e794ae0d2",
  }
}

```

```

    "createdAt": 1571259453.795,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "resourceType": "RelationalDatabase",
    "tags": [],
    "relationalDatabaseBlueprintId": "mysql_8_0",
    "relationalDatabaseBundleId": "micro_1_0",
    "masterDatabaseName": "dbmaster",
    "hardware": {
      "cpuCount": 1,
      "diskSizeInGb": 40,
      "ramSizeInGb": 1.0
    },
    "state": "available",
    "backupRetentionEnabled": false,
    "pendingModifiedValues": {},
    "engine": "mysql",
    "engineVersion": "8.0.16",
    "masterUsername": "dbmasteruser",
    "parameterApplyStatus": "in-sync",
    "preferredBackupWindow": "10:01-10:31",
    "preferredMaintenanceWindow": "sat:11:14-sat:11:44",
    "publiclyAccessible": true,
    "masterEndpoint": {
      "port": 3306,
      "address": "ls-9EXAMPLE8ad863723b62ccEXAMPLEa6e794ae0d2.czowadgeezqi.us-
west-2.rds.amazonaws.com"
    },
    "pendingMaintenanceActions": []
  }
}

```

- Untuk detail API, lihat [GetRelationalDatabase](#) di Referensi AWS CLI Perintah.

## get-relational-databases

Contoh kode berikut menunjukkan cara menggunakan `get-relational-databases`.

### AWS CLI

Untuk mendapatkan informasi tentang semua database relasional

`get-relational-databases` Contoh berikut menampilkan rincian tentang semua database relasional di Wilayah dikonfigurasi AWS .

```
aws lightsail get-relational-databases
```

Output:

```
{
  "relationalDatabases": [
    {
      "name": "MySQL",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/8529020c-3ab9-4d51-92af-5EXAMPLE8979",
      "supportCode": "6EXAMPLE3362/1s-3EXAMPLEa995d8c3b06b4501356e5f2f28e1aeba",
      "createdAt": 1554306019.155,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "resourceType": "RelationalDatabase",
      "tags": [],
      "relationalDatabaseBlueprintId": "mysql_8_0",
      "relationalDatabaseBundleId": "micro_1_0",
      "masterDatabaseName": "dbmaster",
      "hardware": {
        "cpuCount": 1,
        "diskSizeInGb": 40,
        "ramSizeInGb": 1.0
      },
      "state": "available",
      "backupRetentionEnabled": true,
      "pendingModifiedValues": {},
      "engine": "mysql",
      "engineVersion": "8.0.15",
      "latestRestorableTime": 1571686200.0,
      "masterUsername": "dbmasteruser",
      "parameterApplyStatus": "in-sync",
      "preferredBackupWindow": "07:51-08:21",
      "preferredMaintenanceWindow": "tue:12:18-tue:12:48",
      "publiclyAccessible": true,
      "masterEndpoint": {
        "port": 3306,

```

```
        "address":
    "ls-3EXAMPLEa995d8c3b06b4501356e5f2fEXAMPLEa.czowadgeezqi.us-
west-2.rds.amazonaws.com"
    },
    "pendingMaintenanceActions": []
  },
  {
    "name": "Postgres",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/
e9780b6b-d0ab-4af2-85f1-1EXAMPLEac68",
    "supportCode": "6EXAMPLE3362/
ls-3EXAMPLEb4ffffb5cec056220c734713e14bd5fcd",
    "createdAt": 1554306000.814,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "resourceType": "RelationalDatabase",
    "tags": [],
    "relationalDatabaseBlueprintId": "postgres_11",
    "relationalDatabaseBundleId": "micro_1_0",
    "masterDatabaseName": "dbmaster",
    "hardware": {
      "cpuCount": 1,
      "diskSizeInGb": 40,
      "ramSizeInGb": 1.0
    },
    "state": "available",
    "backupRetentionEnabled": true,
    "pendingModifiedValues": {},
    "engine": "postgres",
    "engineVersion": "11.1",
    "latestRestorableTime": 1571686339.0,
    "masterUsername": "dbmasteruser",
    "parameterApplyStatus": "in-sync",
    "preferredBackupWindow": "06:19-06:49",
    "preferredMaintenanceWindow": "sun:10:19-sun:10:49",
    "publiclyAccessible": false,
    "masterEndpoint": {
      "port": 5432,
      "address":
    "ls-3EXAMPLEb4ffffb5cec056220c734713eEXAMPLEd.czowadgeezqi.us-
west-2.rds.amazonaws.com"
    }
  },
```

```

        "pendingMaintenanceActions": []
    }
]
}

```

- Untuk detail API, lihat [GetRelationalDatabases](#) di Referensi AWS CLI Perintah.

## get-static-ip

Contoh kode berikut menunjukkan cara menggunakan `get-static-ip`.

### AWS CLI

Untuk mendapatkan informasi tentang IP statis

`get-static-ip` Contoh berikut menampilkan rincian tentang IP statis yang ditentukan.

```

aws lightsail get-static-ip \
  --static-ip-name StaticIp-1

```

Output:

```

{
  "staticIp": {
    "name": "StaticIp-1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:StaticIp/2257cd76-1f0e-4ac0-82e2-2EXAMPLE23ad",
    "supportCode": "6EXAMPLE3362/192.0.2.0",
    "createdAt": 1571071325.076,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "StaticIp",
    "ipAddress": "192.0.2.0",
    "isAttached": false
  }
}

```

- Untuk detail API, lihat [GetStaticIp](#) di Referensi AWS CLI Perintah.



## get-static-ips

Contoh kode berikut menunjukkan cara menggunakan `get-static-ips`.

### AWS CLI

Untuk mendapatkan informasi tentang semua IP statis

`get-static-ips` Contoh berikut menampilkan rincian tentang semua IP statis di AWS Wilayah dikonfigurasi.

```
aws lightsail get-static-ips
```

Output:

```
{
  "staticIps": [
    {
      "name": "StaticIp-1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:StaticIp/2257cd76-1f0e-4ac0-8EXAMPLE16f9423ad",
      "supportCode": "6EXAMPLE3362/192.0.2.0",
      "createdAt": 1571071325.076,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "StaticIp",
      "ipAddress": "192.0.2.0",
      "isAttached": false
    },
    {
      "name": "StaticIP-2",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:StaticIp/c61edb40-e5f0-4fd6-ae7c-8EXAMPLE19f8",
      "supportCode": "6EXAMPLE3362/192.0.2.2",
      "createdAt": 1568305385.681,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "StaticIp",
      "ipAddress": "192.0.2.2",
    }
  ]
}
```

```
        "attachedTo": "WordPress-1",
        "isAttached": true
    }
]
}
```

- Untuk detail API, lihat [GetStaticIps](#) di Referensi AWS CLI Perintah.

## is-vpc-peered

Contoh kode berikut menunjukkan cara menggunakan `is-vpc-peered`.

### AWS CLI

Untuk mengidentifikasi apakah cloud pribadi virtual Amazon Lightsail Anda diintip

`is-vpc-peered` Contoh berikut mengembalikan status peering Amazon Lightsail virtual private cloud (VPC) untuk Wilayah tertentu. AWS

```
aws lightsail is-vpc-peered \
  --region us-west-2
```

Output:

```
{
  "isPeered": true
}
```

- Untuk detail API, lihat [IsVpcPeered](#) di Referensi AWS CLI Perintah.

## open-instance-public-ports

Contoh kode berikut menunjukkan cara menggunakan `open-instance-public-ports`.

### AWS CLI

Untuk membuka port firewall untuk sebuah contoh

`open-instance-public-ports` Contoh berikut membuka port TCP 22 pada contoh yang ditentukan.

```
aws lightsail open-instance-public-ports \
```

```
--instance-name MEAN-2 \  
--port-info fromPort=22,protocol=TCP,toPort=22
```

**Output:**

```
{  
  "operation": {  
    "id": "719744f0-a022-46f2-9f11-6EXAMPLE4642",  
    "resourceName": "MEAN-2",  
    "resourceType": "Instance",  
    "createdAt": 1571072906.849,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "isTerminal": true,  
    "operationDetails": "22/tcp",  
    "operationType": "OpenInstancePublicPorts",  
    "status": "Succeeded",  
    "statusChangedAt": 1571072906.849  
  }  
}
```

- Untuk detail API, lihat [OpenInstancePublicPorts](#) di Referensi AWS CLI Perintah.

**peer-vpc**

Contoh kode berikut menunjukkan cara menggunakan `peer-vpc`.

**AWS CLI**

Untuk mengintip cloud pribadi virtual Amazon Lightsail

`peer-vpc` Contoh berikut mengintip Amazon Lightsail virtual private cloud (VPC) untuk Wilayah tertentu. AWS

```
aws lightsail peer-vpc \  
--region us-west-2
```

**Output:**

```
{
```

```

    "operation": {
      "id": "787e846a-54ac-497f-bce2-9EXAMPLE5d91",
      "resourceName": "vpc-0EXAMPLEa5261efb3",
      "resourceType": "PeeredVpc",
      "createdAt": 1571694233.104,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationDetails": "vpc-e2b3eb9b",
      "operationType": "PeeredVpc",
      "status": "Succeeded",
      "statusChangedAt": 1571694233.104
    }
  }
}

```

- Untuk detail API, lihat [PeerVpc](#) di Referensi AWS CLI Perintah.

## reboot-instance

Contoh kode berikut menunjukkan cara menggunakan `reboot-instance`.

### AWS CLI

Untuk me-reboot sebuah instance

`reboot-instance` Contoh berikut reboot instance yang ditentukan.

```

aws lightsail reboot-instance \
  --instance-name MEAN-1

```

Output:

```

{
  "operations": [
    {
      "id": "2b679f1c-8b71-4bb4-8e97-8EXAMPLEed93",
      "resourceName": "MEAN-1",
      "resourceType": "Instance",
      "createdAt": 1571694445.49,
      "location": {
        "availabilityZone": "us-west-2a",

```

```

        "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "",
    "operationType": "RebootInstance",
    "status": "Succeeded",
    "statusChangedAt": 1571694445.49
  }
]
}

```

- Untuk detail API, lihat [RebootInstance](#) di Referensi AWS CLI Perintah.

## reboot-relational-database

Contoh kode berikut menunjukkan cara menggunakan `reboot-relational-database`.

### AWS CLI

Untuk me-reboot database relasional

`reboot-relational-database` Contoh berikut reboot database relasional tertentu.

```
aws lightsail reboot-relational-database \
  --relational-database-name Database-1
```

Output:

```

{
  "operations": [
    {
      "id": "e4c980c0-3137-496c-9c91-1EXAMPLEdec2",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1571694532.91,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
    },
    "isTerminal": false,
    "operationDetails": "",
    "operationType": "RebootRelationalDatabase",
    "status": "Started",
  ]
}

```

```

    "statusChangedAt": 1571694532.91
  }
]
}

```

- Untuk detail API, lihat [RebootRelationalDatabase](#) di Referensi AWS CLI Perintah.

## release-static-ip

Contoh kode berikut menunjukkan cara menggunakan `release-static-ip`.

### AWS CLI

Untuk menghapus IP statis

`release-static-ip` Contoh berikut menghapus IP statis yang ditentukan.

```

aws lightsail release-static-ip \
  --static-ip-name StaticIp-1

```

Output:

```

{
  "operations": [
    {
      "id": "e374c002-dc6d-4c7f-919f-2EXAMPLE13ce",
      "resourceName": "StaticIp-1",
      "resourceType": "StaticIp",
      "createdAt": 1571694962.003,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "ReleaseStaticIp",
      "status": "Succeeded",
      "statusChangedAt": 1571694962.003
    }
  ]
}

```

- Untuk detail API, lihat [ReleaseStaticIp](#) di Referensi AWS CLI Perintah.

## start-instance

Contoh kode berikut menunjukkan cara menggunakan `start-instance`.

### AWS CLI

Untuk memulai sebuah instance

`start-instance` Contoh berikut memulai contoh yang ditentukan.

```
aws lightsail start-instance \  
  --instance-name WordPress-1
```

Output:

```
{  
  "operations": [  
    {  
      "id": "f88d2a93-7cea-4165-afce-2d688cb18f23",  
      "resourceName": "WordPress-1",  
      "resourceType": "Instance",  
      "createdAt": 1571695583.463,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "StartInstance",  
      "status": "Started",  
      "statusChangedAt": 1571695583.463  
    }  
  ]  
}
```

- Untuk detail API, lihat [StartInstance](#) di Referensi AWS CLI Perintah.

## start-relational-database

Contoh kode berikut menunjukkan cara menggunakan `start-relational-database`.

### AWS CLI

Untuk memulai database relasional

`start-relational-database` Contoh berikut memulai database relasional yang ditentukan.

```
aws lightsail start-relational-database \  
  --relational-database-name Database-1
```

Output:

```
{  
  "operations": [  
    {  
      "id": "4d5294ec-a38a-4fda-9e37-aEXAMPLE0d24",  
      "resourceName": "Database-1",  
      "resourceType": "RelationalDatabase",  
      "createdAt": 1571695998.822,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "StartRelationalDatabase",  
      "status": "Started",  
      "statusChangedAt": 1571695998.822  
    }  
  ]  
}
```

- Untuk detail API, lihat [StartRelationalDatabase](#) di Referensi AWS CLI Perintah.

## stop-instance

Contoh kode berikut menunjukkan cara menggunakan `stop-instance`.

### AWS CLI

Untuk menghentikan sebuah instance

`stop-instance` Contoh berikut menghentikan contoh yang ditentukan.

```
aws lightsail stop-instance \  
  --instance-name WordPress-1
```

Output:



```
{
  "operations": [
    {
      "id": "265357e2-2943-4d51-888a-1EXAMPLE7585",
      "resourceName": "WordPress-1",
      "resourceType": "Instance",
      "createdAt": 1571695471.134,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "StopInstance",
      "status": "Started",
      "statusChangedAt": 1571695471.134
    }
  ]
}
```

- Untuk detail API, lihat [StopInstance](#) di Referensi AWS CLI Perintah.

## stop-relational-database

Contoh kode berikut menunjukkan cara menggunakan `stop-relational-database`.

### AWS CLI

Untuk menghentikan database relasional

`stop-relational-database` Contoh berikut menghentikan database relasional yang ditentukan.

```
aws lightsail stop-relational-database \
  --relational-database-name Database-1
```

Output:

```
{
  "operations": [
    {
      "id": "cc559c19-4adb-41e4-b75b-5EXAMPLE4e61",
      "resourceName": "Database-1",
```

```

    "resourceType": "RelationalDatabase",
    "createdAt": 1571695526.29,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "StopRelationalDatabase",
    "status": "Started",
    "statusChangedAt": 1571695526.29
  }
]
}

```

- Untuk detail API, lihat [StopRelationalDatabase](#) di Referensi AWS CLI Perintah.

## unpeer-vpc

Contoh kode berikut menunjukkan cara menggunakan `unpeer-vpc`.

### AWS CLI

Untuk membuka awan pribadi virtual Amazon Lightsail

`unpeer-vpc` Contoh berikut mengacaukan cloud pribadi virtual Amazon Lightsail (VPC) untuk Wilayah tertentu. AWS

```
aws lightsail unpeer-vpc \
  --region us-west-2
```

Output:

```

{
  "operation": {
    "id": "531aca64-7157-47ab-84c6-eEXAMPLEd898",
    "resourceName": "vpc-0EXAMPLEa5261efb3",
    "resourceType": "PeeredVpc",
    "createdAt": 1571694109.945,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    }
  },

```

```
    "isTerminal": true,  
    "operationDetails": "vpc-e2b3eb9b",  
    "operationType": "UnpeeredVpc",  
    "status": "Succeeded",  
    "statusChangedAt": 1571694109.945  
  }  
}
```

- Untuk detail API, lihat [UnpeerVpc](#) di Referensi AWS CLI Perintah.

## Contoh Macie menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Macie.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-buckets**

Contoh kode berikut menunjukkan cara menggunakan `describe-buckets`.

AWS CLI

Untuk menanyakan data tentang satu atau beberapa bucket S3 yang dipantau dan dianalisis Amazon Macie untuk akun Anda

`describe-buckets` Contoh berikut menanyakan metadata untuk semua bucket S3 yang namanya dimulai dengan MY-S3 dan berada di Wilayah saat ini. AWS

```
aws macie2 describe-buckets \  
  --criteria '{"bucketName":{"prefix":"my-S3"}}'
```

## Output:

```
{  
  "buckets": [  
    {  
      "accountId": "123456789012",  
      "allowsUnencryptedObjectUploads": "FALSE",  
      "bucketArn": "arn:aws:s3:::MY-S3-DOC-EXAMPLE-BUCKET1",  
      "bucketCreatedAt": "2020-05-18T19:54:00+00:00",  
      "bucketName": "MY-S3-DOC-EXAMPLE-BUCKET1",  
      "classifiableObjectCount": 13,  
      "classifiableSizeInBytes": 1592088,  
      "jobDetails": {  
        "isDefinedInJob": "TRUE",  
        "isMonitoredByJob": "TRUE",  
        "lastJobId": "08c81dc4a2f3377fae45c9ddaexample",  
        "lastJobRunTime": "2021-04-26T14:55:30.270000+00:00"  
      },  
      "lastAutomatedDiscoveryTime": "2022-12-10T19:11:25.364000+00:00",  
      "lastUpdated": "2022-12-13T07:33:06.337000+00:00",  
      "objectCount": 13,  
      "objectCountByEncryptionType": {  
        "customerManaged": 0,  
        "kmsManaged": 2,  
        "s3Managed": 7,  
        "unencrypted": 4,  
        "unknown": 0  
      },  
      "publicAccess": {  
        "effectivePermission": "NOT_PUBLIC",  
        "permissionConfiguration": {  
          "accountLevelPermissions": {  
            "blockPublicAccess": {  
              "blockPublicAcls": true,  
              "blockPublicPolicy": true,  
              "ignorePublicAcls": true,  
              "restrictPublicBuckets": true  
            }  
          },  
          "bucketLevelPermissions": {
```

```
        "accessControllist": {
            "allowsPublicReadAccess": false,
            "allowsPublicWriteAccess": false
        },
        "blockPublicAccess": {
            "blockPublicAcls": true,
            "blockPublicPolicy": true,
            "ignorePublicAcls": true,
            "restrictPublicBuckets": true
        },
        "bucketPolicy": {
            "allowsPublicReadAccess": false,
            "allowsPublicWriteAccess": false
        }
    }
},
"region": "us-west-2",
"replicationDetails": {
    "replicated": false,
    "replicatedExternally": false,
    "replicationAccounts": []
},
"sensitivityScore": 78,
"serverSideEncryption": {
    "kmsMasterKeyId": null,
    "type": "NONE"
},
"sharedAccess": "NOT_SHARED",
"sizeInBytes": 4549746,
"sizeInBytesCompressed": 0,
"tags": [
    {
        "key": "Division",
        "value": "HR"
    },
    {
        "key": "Team",
        "value": "Recruiting"
    }
],
"unclassifiableObjectCount": {
    "fileType": 0,
    "storageClass": 0,
```

```
        "total": 0
    },
    "unclassifiableObjectSizeInBytes": {
        "fileType": 0,
        "storageClass": 0,
        "total": 0
    },
    "versioning": true
},
{
    "accountId": "123456789012",
    "allowsUnencryptedObjectUploads": "TRUE",
    "bucketArn": "arn:aws:s3:::MY-S3-DOC-EXAMPLE-BUCKET2",
    "bucketCreatedAt": "2020-11-25T18:24:38+00:00",
    "bucketName": "MY-S3-DOC-EXAMPLE-BUCKET2",
    "classifiableObjectCount": 8,
    "classifiableSizeInBytes": 133810,
    "jobDetails": {
        "isDefinedInJob": "TRUE",
        "isMonitoredByJob": "FALSE",
        "lastJobId": "188d4f6044d621771ef7d65f2example",
        "lastJobRunTime": "2021-04-09T19:37:11.511000+00:00"
    },
    "lastAutomatedDiscoveryTime": "2022-12-12T19:11:25.364000+00:00",
    "lastUpdated": "2022-12-13T07:33:06.337000+00:00",
    "objectCount": 8,
    "objectCountByEncryptionType": {
        "customerManaged": 0,
        "kmsManaged": 0,
        "s3Managed": 8,
        "unencrypted": 0,
        "unknown": 0
    },
    "publicAccess": {
        "effectivePermission": "NOT_PUBLIC",
        "permissionConfiguration": {
            "accountLevelPermissions": {
                "blockPublicAccess": {
                    "blockPublicAcls": true,
                    "blockPublicPolicy": true,
                    "ignorePublicAcls": true,
                    "restrictPublicBuckets": true
                }
            }
        }
    },
},
```

```
        "bucketLevelPermissions": {
          "accessControlList": {
            "allowsPublicReadAccess": false,
            "allowsPublicWriteAccess": false
          },
          "blockPublicAccess": {
            "blockPublicAcls": true,
            "blockPublicPolicy": true,
            "ignorePublicAcls": true,
            "restrictPublicBuckets": true
          },
          "bucketPolicy": {
            "allowsPublicReadAccess": false,
            "allowsPublicWriteAccess": false
          }
        }
      },
    },
    "region": "us-west-2",
    "replicationDetails": {
      "replicated": false,
      "replicatedExternally": false,
      "replicationAccounts": []
    },
    "sensitivityScore": 95,
    "serverSideEncryption": {
      "kmsMasterKeyId": null,
      "type": "AES256"
    },
    "sharedAccess": "EXTERNAL",
    "sizeInBytes": 175978,
    "sizeInBytesCompressed": 0,
    "tags": [
      {
        "key": "Division",
        "value": "HR"
      },
      {
        "key": "Team",
        "value": "Recruiting"
      }
    ],
    "unclassifiableObjectCount": {
      "fileType": 3,
```

```
        "storageClass": 0,  
        "total": 3  
    },  
    "unclassifiableObjectSizeInBytes": {  
        "fileType": 2999826,  
        "storageClass": 0,  
        "total": 2999826  
    },  
    "versioning": true  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Memfilter inventaris bucket S3 di Panduan Pengguna Amazon Macie](#).

- Untuk detail API, lihat [DescribeBuckets](#) di Referensi AWS CLI Perintah.

## Contoh Grafana yang Dikelola Amazon menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan Grafana Dikelola Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **list-workspaces**

Contoh kode berikut menunjukkan cara menggunakan `list-workspaces`.



## AWS CLI

Untuk mencantumkan ruang kerja untuk akun di Wilayah yang ditentukan oleh kredensi pengguna `list-workspaces` Contoh berikut mencantumkan ruang kerja Grafana untuk Wilayah akun.

```
aws grafana list-workspaces
```

Output:

```
{
  "workspaces": [
    {
      "authentication": {
        "providers": [
          "AWS_SSO"
        ]
      },
      "created": "2022-04-04T16:20:21.796000-07:00",
      "description": "to test tags",
      "endpoint": "g-949e7b44df.grafana-workspace.us-east-1.amazonaws.com",
      "grafanaVersion": "8.2",
      "id": "g-949e7b44df",
      "modified": "2022-04-04T16:20:21.796000-07:00",
      "name": "testtag2",
      "notificationDestinations": [
        "SNS"
      ],
      "status": "ACTIVE"
    },
    {
      "authentication": {
        "providers": [
          "AWS_SSO"
        ]
      },
      "created": "2022-04-20T10:22:15.115000-07:00",
      "description": "ww",
      "endpoint": "g-bffa51ed1b.grafana-workspace.us-east-1.amazonaws.com",
      "grafanaVersion": "8.2",
      "id": "g-bffa51ed1b",
      "modified": "2022-04-20T10:22:15.115000-07:00",
      "name": "ww",
```

```
        "notificationDestinations": [
            "SNS"
        ],
        "status": "ACTIVE"
    }
]
```

- Untuk detail API, lihat [ListWorkspaces](#) di Referensi AWS CLI Perintah.

## MediaConnect contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with MediaConnect.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-flow-outputs**

Contoh kode berikut menunjukkan cara menggunakan `add-flow-outputs`.

AWS CLI

Untuk menambahkan output ke aliran

`add-flow-outputs` Contoh berikut menambahkan output ke aliran yang ditentukan.

```
aws mediaconnect add-flow-outputs \
```

```
--flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
--outputs Description='NYC
stream',Destination=192.0.2.12,Name=NYC,Port=3333,Protocol=rtp-
fec,SmoothingLatency=100 Description='LA
stream',Destination=203.0.113.9,Name=LA,Port=4444,Protocol=rtp-
fec,SmoothingLatency=100
```

## Output:

```
{
  "Outputs": [
    {
      "Port": 3333,
      "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
      "Name": "NYC",
      "Description": "NYC stream",
      "Destination": "192.0.2.12",
      "Transport": {
        "Protocol": "rtp-fec",
        "SmoothingLatency": 100
      }
    },
    {
      "Port": 4444,
      "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-987655dEF67hiJ89-c34de5fG678h:LA",
      "Name": "LA",
      "Description": "LA stream",
      "Destination": "203.0.113.9",
      "Transport": {
        "Protocol": "rtp-fec",
        "SmoothingLatency": 100
      }
    }
  ],
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan Output ke Alur](#) di MediaConnect Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [AddFlowOutputs](#) di Referensi AWS CLI Perintah.

## create-flow

Contoh kode berikut menunjukkan cara menggunakan `create-flow`.

### AWS CLI

Untuk membuat aliran

`create-flow` Contoh berikut menciptakan aliran dengan konfigurasi yang ditentukan.

```
aws mediaconnect create-flow \  
  --availability-zone us-west-2c \  
  --name ExampleFlow \  
  --source Description='Example source,  
backup', IngestPort=1055, Name=BackupSource, Protocol=rtp, WhitelistCidr=10.24.34.0/23
```

Output:

```
{  
  "Flow": {  
    "FlowArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:ExampleFlow",  
    "AvailabilityZone": "us-west-2c",  
    "EgressIp": "54.245.71.21",  
    "Source": {  
      "IngestPort": 1055,  
      "SourceArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:source:2-3aBC45dEF67hiJ89-c34de5fG678h:BackupSource",  
      "Transport": {  
        "Protocol": "rtp",  
        "MaxBitrate": 80000000  
      },  
      "Description": "Example source, backup",  
      "IngestIp": "54.245.71.21",  
      "WhitelistCidr": "10.24.34.0/23",  
      "Name": "mySource"  
    },  
    "Entitlements": [],  
    "Name": "ExampleFlow",  
    "Outputs": [],  
  },  
}
```

```
    "Status": "STANDBY",
    "Description": "Example source, backup"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Alur](#) di Panduan MediaConnect Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateFlow](#) di Referensi AWS CLI Perintah.

## **delete-flow**

Contoh kode berikut menunjukkan cara menggunakan `delete-flow`.

### AWS CLI

Untuk menghapus aliran

`delete-flow` Contoh berikut menghapus aliran yang ditentukan.

```
aws mediaconnect delete-flow \
  --flow-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Output:

```
{
  "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
  "Status": "DELETING"
}
```

Untuk informasi selengkapnya, lihat [Menghapus Alur](#) di MediaConnect Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeleteFlow](#) di Referensi AWS CLI Perintah.

## **describe-flow**

Contoh kode berikut menunjukkan cara menggunakan `describe-flow`.

## AWS CLI

Untuk melihat detail aliran

`describe-flow` Contoh berikut menampilkan detail alur yang ditentukan, seperti ARN, Availability Zone, status, source, entitlements, dan output.

```
aws mediaconnect describe-flow \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Output:

```
{  
  "Flow": {  
    "EgressIp": "54.201.4.39",  
    "AvailabilityZone": "us-west-2c",  
    "Status": "ACTIVE",  
    "FlowArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",  
    "Entitlements": [  
      {  
        "EntitlementArn": "arn:aws:mediaconnect:us-  
west-2:123456789012:entitlement:1-AaBb11CcDd22EeFf-34DE5fG12AbC:MyEntitlement",  
        "Description": "Assign to this account",  
        "Name": "MyEntitlement",  
        "Subscribers": [  
          "444455556666"  
        ]  
      }  
    ],  
    "Description": "NYC awards show",  
    "Name": "AwardsShow",  
    "Outputs": [  
      {  
        "Port": 2355,  
        "Name": "NYC",  
        "Transport": {  
          "SmoothingLatency": 0,  
          "Protocol": "rtp-fec"  
        },  
        "OutputArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
```

```

        "Destination": "192.0.2.0"
    },
    {
        "Port": 3025,
        "Name": "LA",
        "Transport": {
            "SmoothingLatency": 0,
            "Protocol": "rtp-fec"
        },
        "OutputArn": "arn:aws:mediaconnect:us-
east-1:123456789012:output:2-987655dEF67hiJ89-c34de5fG678h:LA",
        "Destination": "192.0.2.0"
    }
],
"Source": {
    "IngestIp": "54.201.4.39",
    "SourceArn": "arn:aws:mediaconnect:us-
east-1:123456789012:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource",
    "Transport": {
        "MaxBitrate": 80000000,
        "Protocol": "rtp"
    },
    "IngestPort": 1069,
    "Description": "Saturday night show",
    "Name": "ShowSource",
    "WhitelistCidr": "10.24.34.0/23"
}
}
}

```

Untuk informasi selengkapnya, lihat [Melihat Detail Aliran](#) di Panduan MediaConnect Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribeFlow](#) di Referensi AWS CLI Perintah.

## grant-flow-entitlements

Contoh kode berikut menunjukkan cara menggunakan `grant-flow-entitlements`.

### AWS CLI

Untuk memberikan hak pada aliran

`grant-flow-entitlements` Contoh berikut memberikan hak atas alur yang ada yang ditentukan untuk membagikan konten Anda dengan akun lain AWS .

```
aws mediaconnect grant-flow-entitlements \
  --flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
  --entitlements Description='For
AnyCompany',Encryption={"Algorithm=aes128,KeyType=static-
key,RoleArn=arn:aws:iam::111122223333:role/MediaConnect-
ASM,SecretArn=arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1"},Name=AnyCompany_Entitlement,Subscribers=444455556666
Description='For Example Corp',Name=ExampleCorp,Subscribers=777788889999
```

Output:

```
{
  "Entitlements": [
    {
      "Name": "AnyCompany_Entitlement",
      "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement",
      "Subscribers": [
        "444455556666"
      ],
      "Description": "For AnyCompany",
      "Encryption": {
        "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1",
        "Algorithm": "aes128",
        "RoleArn": "arn:aws:iam::111122223333:role/MediaConnect-ASM",
        "KeyType": "static-key"
      }
    },
    {
      "Name": "ExampleCorp",
      "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-3333cccc4444dddd-1111aaaa2222:ExampleCorp",
      "Subscribers": [
        "777788889999"
      ],
      "Description": "For Example Corp"
    }
  ],
}
```



```
"FlowArn": "arn:aws:mediacconnect:us-  
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame"  
}
```

Untuk informasi selengkapnya, lihat [Memberikan Hak pada Alur](#) di Panduan Pengguna AWS Elemental MediaConnect .

- Untuk detail API, lihat [GrantFlowEntitlements](#) di Referensi AWS CLI Perintah.

## list-entitlements

Contoh kode berikut menunjukkan cara menggunakan `list-entitlements`.

### AWS CLI

Untuk melihat daftar hak

`list-entitlements` Contoh berikut menampilkan daftar semua hak yang telah diberikan ke akun.

```
aws mediacconnect list-entitlements
```

Output:

```
{  
  "Entitlements": [  
    {  
      "EntitlementArn": "arn:aws:mediacconnect:us-  
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:MyEntitlement",  
      "EntitlementName": "MyEntitlement"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListEntitlements](#) di Referensi MediaConnect API AWS Elemental.

- Untuk detail API, lihat [ListEntitlements](#) di Referensi AWS CLI Perintah.

## list-flows

Contoh kode berikut menunjukkan cara menggunakan `list-flows`.

## AWS CLI

Untuk melihat daftar alur

`list-flows` Contoh berikut menampilkan daftar aliran.

```
aws mediaconnect list-flows
```

Output:

```
{
  "Flows": [
    {
      "Status": "STANDBY",
      "SourceType": "OWNED",
      "AvailabilityZone": "us-west-2a",
      "Description": "NYC awards show",
      "Name": "AwardsShow",
      "FlowArn": "arn:aws:mediaconnect:us-east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow"
    },
    {
      "Status": "STANDBY",
      "SourceType": "OWNED",
      "AvailabilityZone": "us-west-2c",
      "Description": "LA basketball game",
      "Name": "BasketballGame",
      "FlowArn": "arn:aws:mediaconnect:us-east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Daftar Alur](#) di Panduan MediaConnect Pengguna AWS Elemental.

- Untuk detail API, lihat [ListFlows](#) di Referensi AWS CLI Perintah.

## `list-tags-for-resource`

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk daftar tag untuk sumber MediaConnect daya

`list-tags-for-resource` Contoh berikut menampilkan kunci tag dan nilai-nilai yang terkait dengan MediaConnect sumber daya yang ditentukan.

```
aws mediaconnect list-tags-for-resource \  
  --resource-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame
```

Output:

```
{  
  "Tags": {  
    "region": "west",  
    "stage": "prod"  
  }  
}
```

Untuk informasi selengkapnya, lihat [ListTagsForResource TagResource](#), [UntagResource](#) di Referensi MediaConnect API AWS Elemental.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## remove-flow-output

Contoh kode berikut menunjukkan cara menggunakan `remove-flow-output`.

## AWS CLI

Untuk menghapus output dari aliran

`remove-flow-output` Contoh berikut menghapus output dari aliran yang ditentukan.

```
aws mediaconnect remove-flow-output \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \  
  --output-arn arn:aws:mediaconnect:us-  
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC
```

Output:

```
{
  "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
  "OutputArn": "arn:aws:mediaconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC"
}
```

Untuk informasi selengkapnya, lihat [Menghapus Output dari Alur](#) di MediaConnect Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [RemoveFlowOutput](#) di Referensi AWS CLI Perintah.

## revoke-flow-entitlement

Contoh kode berikut menunjukkan cara menggunakan `revoke-flow-entitlement`.

### AWS CLI

Untuk mencabut hak

`revoke-flow-entitlement` Contoh berikut mencabut hak pada alur yang ditentukan.

```
aws mediaconnect revoke-flow-entitlement \
  --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
  --entitlement-arn arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement
```

Output:

```
{
  "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
  "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement"
}
```

Untuk informasi selengkapnya, lihat [Mencabut Hak](#) di Panduan Pengguna AWS Elemental MediaConnect .

- Untuk detail API, lihat [RevokeFlowEntitlement](#) di Referensi AWS CLI Perintah.

## start-flow

Contoh kode berikut menunjukkan cara menggunakan `start-flow`.

### AWS CLI

Untuk memulai aliran

`start-flow` Contoh berikut memulai aliran yang ditentukan.

```
aws mediaconnect start-flow \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Perintah ini tidak menghasilkan output. Output:

```
{  
  "FlowArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",  
  "Status": "STARTING"  
}
```

Untuk informasi selengkapnya, lihat [Memulai Alur](#) di Panduan MediaConnect Pengguna AWS Elemental.

- Untuk detail API, lihat [StartFlow](#) di Referensi AWS CLI Perintah.

## stop-flow

Contoh kode berikut menunjukkan cara menggunakan `stop-flow`.

### AWS CLI

Untuk menghentikan aliran

`stop-flow` Contoh berikut menghentikan aliran yang ditentukan.

```
aws mediaconnect stop-flow \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Output:

```
{
  "Status": "STOPPING",
  "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow"
}
```

Untuk informasi selengkapnya, lihat [Menghentikan Aliran](#) di Panduan MediaConnect Pengguna AWS Elemental.

- Untuk detail API, lihat [StopFlow](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber MediaConnect daya

`tag-resource` Contoh berikut menambahkan tag dengan nama kunci dan nilai ke MediaConnect sumber daya yang ditentukan.

```
aws mediaconnect tag-resource \
  --resource-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame
  --tags region=west
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ListTagsForResource TagResource](#), [UntagResource](#) di Referensi MediaConnect API AWS Elemental.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber MediaConnect daya

`untag-resource` Contoh berikut menghapus tag dengan nama kunci yang ditentukan dan nilai terkait dari MediaConnect sumber daya.

```
aws mediaconnect untag-resource \  
  --resource-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame \  
  --tag-keys region
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [ListTagsForResource TagResource](#), [UntagResource](#) di Referensi MediaConnect API AWS Elemental.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-flow-entitlement

Contoh kode berikut menunjukkan cara menggunakan `update-flow-entitlement`.

### AWS CLI

Untuk memperbarui hak

`update-flow-entitlement` Contoh berikut memperbarui hak yang ditentukan dengan deskripsi dan pelanggan baru.

```
aws mediaconnect update-flow-entitlement \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \  
  --entitlement-arn arn:aws:mediaconnect:us-  
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement  
 \  
  --description 'For AnyCompany Affiliate' \  
  --subscribers 777788889999
```

Output:

```
{  
  "FlowArn": "arn:aws:mediaconnect:us-  
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",  
  "Entitlement": {  
    "Name": "AnyCompany_Entitlement",
```

```

    "Description": "For AnyCompany Affiliate",
    "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement",
    "Encryption": {
      "KeyType": "static-key",
      "Algorithm": "aes128",
      "RoleArn": "arn:aws:iam::111122223333:role/MediaConnect-ASM",
      "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1"
    },
    "Subscribers": [
      "777788889999"
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Hak](#) di MediaConnect Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [UpdateFlowEntitlement](#) di Referensi AWS CLI Perintah.

## update-flow-output

Contoh kode berikut menunjukkan cara menggunakan `update-flow-output`.

### AWS CLI

Untuk memperbarui output pada aliran

`update-flow-output` Contoh berikut memperbarui output pada aliran yang ditentukan.

```

aws mediacconnect update-flow-output \
  --flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
  --output-arn arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC \
  --port 3331

```

Output:

```

{
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",

```



```

"Output": {
  "Name": "NYC",
  "Port": 3331,
  "Description": "NYC stream",
  "Transport": {
    "Protocol": "rtp-fec",
    "SmoothingLatency": 100
  },
  "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
  "Destination": "192.0.2.12"
}
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Output pada Alur](#) di MediaConnect Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [UpdateFlowOutput](#) di Referensi AWS CLI Perintah.

## update-flow-source

Contoh kode berikut menunjukkan cara menggunakan `update-flow-source`.

### AWS CLI

Untuk memperbarui sumber aliran yang ada

`update-flow-source` Contoh berikut memperbarui sumber aliran yang ada.

```

aws mediacconnect update-flow-source \
  --flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow \
  --source-arn arn:aws:mediacconnect:us-
east-1:111122223333:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource \
  --description 'Friday night show' \
  --ingest-port 3344 \
  --protocol rtp-fec \
  --whitelist-cidr 10.24.34.0/23

```

Output:

```
{
```

```
"FlowArn": "arn:aws:mediaconnect:us-east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
"Source": {
  "IngestIp": "34.210.136.56",
  "WhitelistCidr": "10.24.34.0/23",
  "Transport": {
    "Protocol": "rtp-fec"
  },
  "IngestPort": 3344,
  "Name": "ShowSource",
  "Description": "Friday night show",
  "SourceArn": "arn:aws:mediaconnect:us-east-1:111122223333:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource"
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Sumber Aliran](#) di Panduan MediaConnect Pengguna AWS Elemental.

- Untuk detail API, lihat [UpdateFlowSource](#) di Referensi AWS CLI Perintah.

## MediaConvert contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with MediaConvert.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **cancel-job**

Contoh kode berikut menunjukkan cara menggunakan `cancel-job`.

#### AWS CLI

Untuk membatalkan pekerjaan yang berada dalam antrian

`cancel-job` Contoh berikut membatalkan pekerjaan dengan ID `1234567891234-abc123`. Anda tidak dapat membatalkan pekerjaan yang telah mulai diproses oleh layanan.

```
aws mediaconvert cancel-job \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --id 1234567891234-abc123
```

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Pekerjaan AWS Elemental](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [CancelJob](#) di Referensi AWS CLI Perintah.

### **create-job-template**

Contoh kode berikut menunjukkan cara menggunakan `create-job-template`.

#### AWS CLI

Untuk membuat template pekerjaan

`create-job-template` Contoh berikut membuat template pekerjaan dengan pengaturan transcoding yang ditentukan dalam file `job-template.json` yang berada di sistem Anda.

```
aws mediaconvert create-job-template \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --name JobTemplate1 \  
  --cli-input-json file://~/job-template.json
```

Jika Anda membuat file JSON template pekerjaan Anda dengan menggunakan `get-job-template` dan kemudian memodifikasi file, hapus `JobTemplate` objek, tetapi simpan objek anak Pengaturan di dalamnya. Juga, pastikan untuk menghapus pasangan kunci-nilai berikut: `LastUpdated`, `ArnType`, dan `CreatedAt` Anda dapat menentukan kategori, deskripsi, nama, dan antrian baik di file JSON atau di baris perintah.

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Jika permintaan Anda berhasil, layanan mengembalikan spesifikasi JSON untuk template pekerjaan yang Anda buat.

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS Elemental MediaConvert Job Templates](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateJobTemplate](#) di Referensi AWS CLI Perintah.

## **create-job**

Contoh kode berikut menunjukkan cara menggunakan `create-job`.

### AWS CLI

Untuk membuat pekerjaan

`create-job` Contoh berikut membuat pekerjaan transcoding dengan pengaturan yang ditentukan dalam file `job.json` yang berada di sistem tempat Anda mengirim perintah. Spesifikasi pekerjaan JSON ini mungkin menentukan setiap pengaturan secara individual, referensi template pekerjaan, atau preset output referensi.

```
aws mediaconvert create-job \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --cli-input-json file://~/job.json
```

Anda dapat menggunakan MediaConvert konsol AWS Elemental untuk menghasilkan spesifikasi pekerjaan JSON dengan memilih pengaturan pekerjaan Anda, dan kemudian memilih Tampilkan pekerjaan JSON di bagian bawah bagian Job.

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Jika permintaan Anda berhasil, layanan mengembalikan spesifikasi pekerjaan JSON yang Anda kirim bersama permintaan Anda.

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Pekerjaan AWS Elemental](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateJob](#) di Referensi AWS CLI Perintah.

## **create-preset**

Contoh kode berikut menunjukkan cara menggunakan `create-preset`.

### AWS CLI

Untuk membuat preset keluaran kustom

`create-preset` Contoh berikut membuat preset keluaran kustom berdasarkan pengaturan output yang ditentukan dalam file `preset.json`. Anda dapat menentukan kategori, deskripsi, dan nama baik di file JSON atau di baris perintah.

```
aws mediaconvert create-preset \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --cli-input-json file://~/preset.json
```

Jika Anda membuat file JSON preset Anda dengan menggunakan `get-preset` dan kemudian memodifikasi file output, pastikan bahwa Anda menghapus pasangan kunci-nilai berikut: `LastUpdated`,, dan `Arn Type CreatedAt`

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Untuk informasi selengkapnya, lihat [Bekerja dengan Preset MediaConvert Output AWS Elemental](#) di Panduan Pengguna AWS Elemental MediaConvert .

- Untuk detail API, lihat [CreatePreset](#) di Referensi AWS CLI Perintah.

## **create-queue**

Contoh kode berikut menunjukkan cara menggunakan `create-queue`.

## AWS CLI

Untuk membuat antrian kustom

`create-queue` Contoh berikut membuat antrian transcoding kustom.

```
aws mediaconvert create-queue \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --name Queue1 \  
  --description "Keep this queue empty unless job is urgent."
```

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Output:

```
{  
  "Queue": {  
    "Status": "ACTIVE",  
    "Name": "Queue1",  
    "LastUpdated": 1518034928,  
    "Arn": "arn:aws:mediaconvert:region-name-1:012345678998:queues/Queue1",  
    "Type": "CUSTOM",  
    "CreatedAt": 1518034928,  
    "Description": "Keep this queue empty unless job is urgent."  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Antrian AWS Elemental di Panduan Pengguna AWS MediaConvert Elemental](#).

- Untuk detail API, lihat [CreateQueue](#) di Referensi AWS CLI Perintah.

## **delete-job-template**

Contoh kode berikut menunjukkan cara menggunakan `delete-job-template`.

## AWS CLI

Untuk menghapus template pekerjaan

`delete-job-template` Contoh berikut menghapus template pekerjaan kustom yang ditentukan.

```
aws mediaconvert delete-job-template \  
  --name "DASH Streaming" \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Perintah ini tidak menghasilkan output. Jalankan `aws mediaconvert list-job-templates` untuk mengonfirmasi bahwa template Anda telah dihapus.

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS Elemental MediaConvert Job Templates](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeleteJobTemplate](#) di Referensi AWS CLI Perintah.

## **delete-preset**

Contoh kode berikut menunjukkan cara menggunakan `delete-preset`.

### AWS CLI

Untuk menghapus antrian sesuai permintaan khusus

`delete-preset` Contoh berikut menghapus preset kustom yang ditentukan.

```
aws mediaconvert delete-preset \  
  --name SimpleMP4 \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Perintah ini tidak menghasilkan output. Jalankan `aws mediaconvert list-presets` untuk mengonfirmasi bahwa preset Anda telah dihapus.

Untuk informasi selengkapnya, lihat [Bekerja dengan Preset MediaConvert Output AWS Elemental](#) di Panduan Pengguna AWS Elemental MediaConvert .

- Untuk detail API, lihat [DeletePreset](#) di Referensi AWS CLI Perintah.

## **delete-queue**

Contoh kode berikut menunjukkan cara menggunakan `delete-queue`.

### AWS CLI

Untuk menghapus antrian sesuai permintaan khusus

`delete-queue` Contoh berikut menghapus antrian sesuai permintaan kustom yang ditentukan.

Anda tidak dapat menghapus antrian default Anda. Anda tidak dapat menghapus antrian cadangan yang memiliki paket harga aktif atau yang berisi pekerjaan yang belum diproses.

```
aws mediaconvert delete-queue \  
  --name Customer1 \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Perintah ini tidak menghasilkan output. Jalankan `aws mediaconvert list-queues` untuk mengonfirmasi bahwa antrian Anda telah dihapus.

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Antrian AWS Elemental di Panduan Pengguna AWS MediaConvert Elemental](#).

- Untuk detail API, lihat [DeleteQueue](#) di Referensi AWS CLI Perintah.

## **describe-endpoints**

Contoh kode berikut menunjukkan cara menggunakan `describe-endpoints`.

### AWS CLI

Untuk mendapatkan titik akhir khusus akun Anda

`describe-endpoints` Contoh berikut mengambil titik akhir yang Anda butuhkan untuk mengirim permintaan lain ke layanan.

```
aws mediaconvert describe-endpoints
```

Output:

```
{  
  "Endpoints": [  
    {  
      "Url": "https://abcd1234.mediaconvert.region-name-1.amazonaws.com"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Memulai MediaConvert Menggunakan API di Referensi MediaConvert API AWS Elemental](#).



- Untuk detail API, lihat [DescribeEndpoints](#) di Referensi AWS CLI Perintah.

## get-job-template

Contoh kode berikut menunjukkan cara menggunakan `get-job-template`.

### AWS CLI

Untuk mendapatkan detail untuk template pekerjaan

`get-job-template` Contoh berikut menampilkan definisi JSON dari template pekerjaan kustom yang ditentukan.

```
aws mediaconvert get-job-template \
  --name "DASH Streaming" \
  --endpoint-url https://abcd1234.mediaconvert.us-east-1.amazonaws.com
```

### Output:

```
{
  "JobTemplate": {
    "StatusUpdateInterval": "SECONDS_60",
    "LastUpdated": 1568652998,
    "Description": "Create a DASH streaming ABR stack",
    "CreatedAt": 1568652998,
    "Priority": 0,
    "Name": "DASH Streaming",
    "Settings": {
      ...<truncatedforbrevity>...
    },
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/DASH
Streaming",
    "Type": "CUSTOM"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS Elemental MediaConvert Job Templates](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [GetJobTemplate](#) di Referensi AWS CLI Perintah.

## get-job

Contoh kode berikut menunjukkan cara menggunakan `get-job`.

### AWS CLI

Untuk mendapatkan detail untuk pekerjaan tertentu

Contoh berikut meminta informasi untuk pekerjaan dengan ID `1234567890987-1ab2c3`, yang dalam contoh ini berakhir dengan kesalahan.

```
aws mediaconvert get-job \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --id 1234567890987-1ab2c3
```

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Jika permintaan Anda berhasil, layanan akan mengembalikan file JSON dengan informasi pekerjaan, termasuk pengaturan pekerjaan, kesalahan yang dikembalikan, dan data pekerjaan lainnya, sebagai berikut:

```
{  
  "Job": {  
    "Status": "ERROR",  
    "Queue": "arn:aws:mediaconvert:region-name-1:012345678998:queues/Queue1",  
    "Settings": {  
      ...<truncated for brevity>...  
    },  
    "ErrorMessage": "Unable to open input file [s3://my-input-bucket/file-name.mp4]: [Failed probe/open: [Failed to read data: AssumeRole failed]]",  
    "ErrorCode": 1434,  
    "Role": "arn:aws:iam::012345678998:role/MediaConvertServiceRole",  
    "Arn": "arn:aws:mediaconvert:us-west-1:012345678998:jobs/1234567890987-1ab2c3",  
    "UserMetadata": {},  
    "Timing": {  
      "FinishTime": 1517442131,  
      "SubmitTime": 1517442103,  
      "StartTime": 1517442104  
    },  
    "Id": "1234567890987-1ab2c3",
```

```
    "CreatedAt": 1517442103
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Pekerjaan AWS Elemental](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [GetJob](#) di Referensi AWS CLI Perintah.

## get-preset

Contoh kode berikut menunjukkan cara menggunakan `get-preset`.

### AWS CLI

Untuk mendapatkan detail untuk preset tertentu

`get-preset` Contoh berikut meminta definisi JSON dari preset kustom yang ditentukan.

```
aws mediaconvert get-preset \
  --name SimpleMP4 \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

### Output:

```
{
  "Preset": {
    "Description": "Creates basic MP4 file. No filtering or preprocessing.",
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4",
    "LastUpdated": 1568843141,
    "Name": "SimpleMP4",
    "Settings": {
      "ContainerSettings": {
        "Mp4Settings": {
          "FreeSpaceBox": "EXCLUDE",
          "CslgAtom": "INCLUDE",
          "MoovPlacement": "PROGRESSIVE_DOWNLOAD"
        },
        "Container": "MP4"
      },
      "AudioDescriptions": [
        {
          "LanguageCodeControl": "FOLLOW_INPUT",
```

```
    "AudioTypeControl": "FOLLOW_INPUT",
    "CodecSettings": {
      "AacSettings": {
        "RawFormat": "NONE",
        "CodecProfile": "LC",
        "AudioDescriptionBroadcasterMix": "NORMAL",
        "SampleRate": 48000,
        "Bitrate": 96000,
        "RateControlMode": "CBR",
        "Specification": "MPEG4",
        "CodingMode": "CODING_MODE_2_0"
      },
      "Codec": "AAC"
    }
  ],
  "VideoDescription": {
    "RespondToAfd": "NONE",
    "TimecodeInsertion": "DISABLED",
    "Sharpness": 50,
    "ColorMetadata": "INSERT",
    "CodecSettings": {
      "H264Settings": {
        "FramerateControl": "INITIALIZE_FROM_SOURCE",
        "SpatialAdaptiveQuantization": "ENABLED",
        "Softness": 0,
        "Telecine": "NONE",
        "CodecLevel": "AUTO",
        "QualityTuningLevel": "SINGLE_PASS",
        "UnregisteredSeiTimecode": "DISABLED",
        "Slices": 1,
        "Syntax": "DEFAULT",
        "GopClosedCadence": 1,
        "AdaptiveQuantization": "HIGH",
        "EntropyEncoding": "CABAC",
        "InterlaceMode": "PROGRESSIVE",
        "ParControl": "INITIALIZE_FROM_SOURCE",
        "NumberBFramesBetweenReferenceFrames": 2,
        "GopSizeUnits": "FRAMES",
        "RepeatPps": "DISABLED",
        "CodecProfile": "MAIN",
        "FieldEncoding": "PAFF",
        "GopSize": 90.0,
        "SlowPal": "DISABLED",
```

```

        "SceneChangeDetect": "ENABLED",
        "GopBReference": "DISABLED",
        "RateControlMode": "CBR",
        "FramerateConversionAlgorithm": "DUPLICATE_DROP",
        "FlickerAdaptiveQuantization": "DISABLED",
        "DynamicSubGop": "STATIC",
        "MinIInterval": 0,
        "TemporalAdaptiveQuantization": "ENABLED",
        "Bitrate": 400000,
        "NumberReferenceFrames": 3
    },
    "Codec": "H_264"
},
"AfdSignaling": "NONE",
"AntiAlias": "ENABLED",
"ScalingBehavior": "DEFAULT",
"DropFrameTimecode": "ENABLED"
}
},
"Type": "CUSTOM",
"CreatedAt": 1568841521
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Preset MediaConvert Output AWS Elemental](#) di Panduan Pengguna AWS Elemental MediaConvert .

- Untuk detail API, lihat [GetPreset](#) di Referensi AWS CLI Perintah.

## get-queue

Contoh kode berikut menunjukkan cara menggunakan `get-queue`.

### AWS CLI

Untuk mendapatkan detail untuk antrian

`get-queue` Contoh berikut mengambil rincian antrian kustom yang ditentukan.

```

aws mediaconvert get-queue \
  --name Customer1 \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com

```

**Output:**

```
{
  "Queue": {
    "LastUpdated": 1526428502,
    "Type": "CUSTOM",
    "SubmittedJobsCount": 0,
    "Status": "ACTIVE",
    "PricingPlan": "ON_DEMAND",
    "CreatedAt": 1526428502,
    "ProgressingJobsCount": 0,
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Customer1",
    "Name": "Customer1"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Antrian AWS Elemental di Panduan Pengguna AWS MediaConvert Elemental](#).

- Untuk detail API, lihat [GetQueue](#) di Referensi AWS CLI Perintah.

**list-job-templates**

Contoh kode berikut menunjukkan cara menggunakan `list-job-templates`.

**AWS CLI**

Contoh 1: Untuk daftar template pekerjaan kustom Anda

`list-job-templates` Contoh berikut mencantumkan semua template pekerjaan kustom di Wilayah saat ini. Untuk membuat daftar templat pekerjaan sistem, lihat contoh berikutnya.

```
aws mediaconvert list-job-templates \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

**Output:**

```
{
  "JobTemplates": [
    {
      "Description": "Create a DASH streaming ABR stack",
```

```

    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/DASH
Streaming",
    "Name": "DASH Streaming",
    "LastUpdated": 1568653007,
    "Priority": 0,
    "Settings": {
        ...<truncatedforbrevity>...
    },
    "Type": "CUSTOM",
    "StatusUpdateInterval": "SECONDS_60",
    "CreatedAt": 1568653007
},
{
    "Description": "Create a high-res file",
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/File",
    "Name": "File",
    "LastUpdated": 1568653007,
    "Priority": 0,
    "Settings": {
        ...<truncatedforbrevity>...
    },
    "Type": "CUSTOM",
    "StatusUpdateInterval": "SECONDS_60",
    "CreatedAt": 1568653023
}
]
}

```

Contoh 2: Untuk membuat daftar templat pekerjaan MediaConvert sistem

`list-job-templates` Contoh berikut mencantumkan semua template pekerjaan sistem.

```

aws mediaconvert list-job-templates \
  --endpoint-url https://abcd1234.mediaconvert.us-east-1.amazonaws.com \
  --list-by SYSTEM

```

Output:

```

{
  "JobTemplates": [
    {
      "CreatedAt": 1568321779,

```

```

    "Arn": "arn:aws:mediaconvert:us-east-1:123456789012:jobTemplates/System-
Generic_Mp4_Hev1_Avc_Aac_Sdr_Qvbr",
    "Name": "System-Generic_Mp4_Hev1_Avc_Aac_Sdr_Qvbr",
    "Description": "GENERIC, MP4, AVC + HEV1(HEVC,SDR), AAC, SDR, QVBR",
    "Category": "GENERIC",
    "Settings": {
      "AdAvailOffset": 0,
      "OutputGroups": [
        {
          "Outputs": [
            {
              "Extension": "mp4",
              "Preset": "System-
Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1280x720p_30Hz_5Mbps_Qvbr_Vq9",
              "NameModifier":
"_Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1280x720p_30Hz_5000Kbps_Qvbr_Vq9"
            },
            {
              "Extension": "mp4",
              "Preset": "System-
Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1920x1080p_30Hz_10Mbps_Qvbr_Vq9",
              "NameModifier":
"_Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1920x1080p_30Hz_10000Kbps_Qvbr_Vq9"
            },
            {
              "Extension": "mp4",
              "Preset": "System-
Generic_Sd_Mp4_Avc_Aac_16x9_Sdr_640x360p_30Hz_0.8Mbps_Qvbr_Vq7",
              "NameModifier":
"_Generic_Sd_Mp4_Avc_Aac_16x9_Sdr_640x360p_30Hz_800Kbps_Qvbr_Vq7"
            },
            {
              "Extension": "mp4",
              "Preset": "System-
Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1280x720p_30Hz_4Mbps_Qvbr_Vq9",
              "NameModifier":
"_Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1280x720p_30Hz_4000Kbps_Qvbr_Vq9"
            },
            {
              "Extension": "mp4",
              "Preset": "System-
Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1920x1080p_30Hz_8Mbps_Qvbr_Vq9",
              "NameModifier":
"_Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1920x1080p_30Hz_8000Kbps_Qvbr_Vq9"
            }
          ]
        }
      ]
    }
  }

```



```

        },
        {
            "Extension": "mp4",
            "Preset": "System-
Generic_Uhd_Mp4_Hev1_Aac_16x9_Sdr_3840x2160p_30Hz_12Mbps_Qvbr_Vq9",
            "NameModifier":
            "_Generic_Uhd_Mp4_Hev1_Aac_16x9_Sdr_3840x2160p_30Hz_12000Kbps_Qvbr_Vq9"
        }
    ],
    "OutputGroupSettings": {
        "FileGroupSettings": {

        },
        "Type": "FILE_GROUP_SETTINGS"
    },
    "Name": "File Group"
}
]
},
"Type": "SYSTEM",
"LastUpdated": 1568321779
},
...<truncatedforbrevity>...
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS Elemental MediaConvert Job Templates](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [ListJobTemplates](#) di Referensi AWS CLI Perintah.

## list-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-jobs`.

### AWS CLI

Untuk mendapatkan detail untuk semua pekerjaan di suatu wilayah

Contoh berikut meminta informasi untuk semua pekerjaan Anda di wilayah yang ditentukan.

```

aws mediaconvert list-jobs \
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \

```

```
--region region-name-1
```

Untuk mendapatkan titik akhir khusus akun Anda, gunakan `describe-endpoints`, atau kirim perintah tanpa titik akhir. Layanan mengembalikan kesalahan dan titik akhir Anda.

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Pekerjaan AWS Elemental](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [ListJobs](#) di Referensi AWS CLI Perintah.

## list-presets

Contoh kode berikut menunjukkan cara menggunakan `list-presets`.

### AWS CLI

Contoh 1: Untuk membuat daftar preset keluaran kustom Anda

`list-presets` Contoh berikut mencantumkan preset keluaran kustom Anda. Untuk membuat daftar preset sistem, lihat contoh berikutnya.

```
aws mediaconvert list-presets \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{  
  "Presets": [  
    {  
      "Name": "SimpleMP4",  
      "CreatedAt": 1568841521,  
      "Settings": {  
        .....  
      },  
      "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleMP4",  
      "Type": "CUSTOM",  
      "LastUpdated": 1568843141,  
      "Description": "Creates basic MP4 file. No filtering or preprocessing."  
    },  
    {  
      "Name": "SimpleTS",
```

```

    "CreatedAt": 1568843113,
    "Settings": {
      ... truncated for brevity ...
    },
    "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleTS",
    "Type": "CUSTOM",
    "LastUpdated": 1568843113,
    "Description": "Create a basic transport stream."
  }
]
}

```

## Contoh 2: Untuk daftar preset output sistem

`list-presets` Contoh berikut mencantumkan preset MediaConvert sistem yang tersedia. Untuk membuat daftar preset kustom Anda, lihat contoh sebelumnya.

```

aws mediaconvert list-presets \
  --list-by SYSTEM \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com

```

## Output:

```

{
  "Presets": [
    {
      "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/System-Avc_16x9_1080p_29_97fps_8500kbps",
      "Name": "System-Avc_16x9_1080p_29_97fps_8500kbps",
      "CreatedAt": 1568321789,
      "Description": "Wifi, 1920x1080, 16:9, 29.97fps, 8500kbps",
      "LastUpdated": 1568321789,
      "Type": "SYSTEM",
      "Category": "HLS",
      "Settings": {
        ...<output settings removed for brevity>...
      }
    },
    ...<list of presets shortened for brevity>...
  ]
}

```

```

    "Arn": "arn:aws:mediaconvert:us-east-1:123456789012:presets/System-
Xdcam_HD_1080i_29_97fps_35mpbs",
    "Name": "System-Xdcam_HD_1080i_29_97fps_35mpbs",
    "CreatedAt": 1568321790,
    "Description": "XDCAM MPEG HD, 1920x1080i, 29.97fps, 35mbps",
    "LastUpdated": 1568321790,
    "Type": "SYSTEM",
    "Category": "MXF",
    "Settings": {
      ...<output settings removed for brevity>...
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Preset MediaConvert Output AWS Elemental](#) di Panduan Pengguna AWS Elemental MediaConvert .

- Untuk detail API, lihat [ListPresets](#) di Referensi AWS CLI Perintah.

## list-queues

Contoh kode berikut menunjukkan cara menggunakan `list-queues`.

### AWS CLI

Untuk membuat daftar antrian Anda

`list-queues` Contoh berikut mencantumkan semua MediaConvert antrian Anda.

```

aws mediaconvert list-queues \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com

```

Output:

```

{
  "Queues": [
    {
      "PricingPlan": "ON_DEMAND",
      "Type": "SYSTEM",
      "Status": "ACTIVE",
      "CreatedAt": 1503451595,
      "Name": "Default",

```

```

    "SubmittedJobsCount": 0,
    "ProgressingJobsCount": 0,
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Default",
    "LastUpdated": 1534549158
  },
  {
    "PricingPlan": "ON_DEMAND",
    "Type": "CUSTOM",
    "Status": "ACTIVE",
    "CreatedAt": 1537460025,
    "Name": "Customer1",
    "SubmittedJobsCount": 0,
    "Description": "Jobs we run for our cusotmer.",
    "ProgressingJobsCount": 0,
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Customer1",
    "LastUpdated": 1537460025
  },
  {
    "ProgressingJobsCount": 0,
    "Status": "ACTIVE",
    "Name": "transcode-library",
    "SubmittedJobsCount": 0,
    "LastUpdated": 1564066204,
    "ReservationPlan": {
      "Status": "ACTIVE",
      "ReservedSlots": 1,
      "PurchasedAt": 1564066203,
      "Commitment": "ONE_YEAR",
      "ExpiresAt": 1595688603,
      "RenewalType": "EXPIRE"
    },
    "PricingPlan": "RESERVED",
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/transcode-
library",
    "Type": "CUSTOM",
    "CreatedAt": 1564066204
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Antrian AWS Elemental di Panduan Pengguna AWS MediaConvert Elemental](#).

- Untuk detail API, lihat [ListQueues](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag pada MediaConvert antrian, templat pekerjaan, atau preset keluaran

`list-tags-for-resource` Contoh berikut mencantumkan tag pada preset output yang ditentukan.

```
aws mediaconvert list-tags-for-resource \
  --arn arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4 \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

### Output:

```
{
  "ResourceTags": {
    "Tags": {
      "customer": "zippyVideo"
    },
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4"
  }
}
```

Untuk informasi selengkapnya, lihat [MediaConvert Menandai Antrian AWS Elemen, Template Pekerjaan, dan Preset Output](#) di AWS Panduan Pengguna Elemental. MediaConvert

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## update-job-template

Contoh kode berikut menunjukkan cara menggunakan `update-job-template`.

### AWS CLI

Untuk mengubah template pekerjaan

`update-job-template` Contoh berikut menggantikan definisi JSON dari template pekerjaan kustom yang ditentukan dengan definisi JSON dalam file yang disediakan.

```
aws mediaconvert update-job-template --name File1 --endpoint-url https://
abcd1234.mediaconvert.us-west-2.amazonaws.com -- file: //~/ .json cli-input-json job-template-
update
```

Isi dari job-template-update.json:

```
{
  "Description": "A simple job template that generates a single file output.",
  "Queue": "arn:aws:mediaconvert:us-east-1:012345678998:queues/Default",
  "Name": "SimpleFile",
  "Settings": {
    "OutputGroups": [
      {
        "Name": "File Group",
        "Outputs": [
          {
            "ContainerSettings": {
              "Container": "MP4",
              "Mp4Settings": {
                "CslgAtom": "INCLUDE",
                "FreeSpaceBox": "EXCLUDE",
                "MoovPlacement": "PROGRESSIVE_DOWNLOAD"
              }
            },
            "VideoDescription": {
              "ScalingBehavior": "DEFAULT",
              "TimecodeInsertion": "DISABLED",
              "AntiAlias": "ENABLED",
              "Sharpness": 50,
              "CodecSettings": {
                "Codec": "H_264",
                "H264Settings": {
                  "InterlaceMode": "PROGRESSIVE",
                  "NumberReferenceFrames": 3,
                  "Syntax": "DEFAULT",
                  "Softness": 0,
                  "GopClosedCadence": 1,
                  "GopSize": 90,
                  "Slices": 1,
                  "GopBReference": "DISABLED",
                  "SlowPal": "DISABLED",
                  "SpatialAdaptiveQuantization": "ENABLED",
                  "TemporalAdaptiveQuantization": "ENABLED",
```

```
    "FlickerAdaptiveQuantization": "DISABLED",
    "EntropyEncoding": "CABAC",
    "Bitrate": 400000,
    "FramerateControl": "INITIALIZE_FROM_SOURCE",
    "RateControlMode": "CBR",
    "CodecProfile": "MAIN",
    "Telecine": "NONE",
    "MinIInterval": 0,
    "AdaptiveQuantization": "HIGH",
    "CodecLevel": "AUTO",
    "FieldEncoding": "PAFF",
    "SceneChangeDetect": "ENABLED",
    "QualityTuningLevel": "SINGLE_PASS",
    "FramerateConversionAlgorithm": "DUPLICATE_DROP",
    "UnregisteredSeiTimecode": "DISABLED",
    "GopSizeUnits": "FRAMES",
    "ParControl": "INITIALIZE_FROM_SOURCE",
    "NumberBFramesBetweenReferenceFrames": 2,
    "RepeatPps": "DISABLED",
    "DynamicSubGop": "STATIC"
  }
},
"AfdSignaling": "NONE",
"DropFrameTimecode": "ENABLED",
"RespondToAfd": "NONE",
"ColorMetadata": "INSERT"
},
"AudioDescriptions": [
  {
    "AudioTypeControl": "FOLLOW_INPUT",
    "CodecSettings": {
      "Codec": "AAC",
      "AacSettings": {
        "AudioDescriptionBroadcasterMix": "NORMAL",
        "Bitrate": 96000,
        "RateControlMode": "CBR",
        "CodecProfile": "LC",
        "CodingMode": "CODING_MODE_2_0",
        "RawFormat": "NONE",
        "SampleRate": 48000,
        "Specification": "MPEG4"
      }
    }
  },
  {
    "LanguageCodeControl": "FOLLOW_INPUT"
```



```

        }
      ]
    }
  ],
  "OutputGroupSettings": {
    "Type": "FILE_GROUP_SETTINGS",
    "FileGroupSettings": {}
  }
}
],
"AdAvailOffset": 0
},
"StatusUpdateInterval": "SECONDS_60",
"Priority": 0
}

```

Sistem mengembalikan payload JSON yang Anda kirim dengan permintaan Anda, bahkan ketika permintaan menghasilkan kesalahan. Oleh karena itu, JSON yang dikembalikan belum tentu merupakan definisi baru dari template pekerjaan.

Karena payload JSON bisa panjang, Anda mungkin perlu menggulir ke atas untuk melihat pesan kesalahan apa pun.

Untuk informasi selengkapnya, lihat [Bekerja dengan AWS Elemental MediaConvert Job Templates](#) di MediaConvert Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [UpdateJobTemplate](#) di Referensi AWS CLI Perintah.

## update-preset

Contoh kode berikut menunjukkan cara menggunakan `update-preset`.

### AWS CLI

Untuk mengubah preset

`update-preset` Contoh berikut menggantikan deskripsi untuk preset yang ditentukan.

```

aws mediaconvert update-preset \
--name Customer1 \
--description "New description text."
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com

```

Perintah ini tidak menghasilkan output. Output:

```
{
  "Preset": {
    "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleMP4",
    "Settings": {
      ...<output settings removed for brevity>...
    },
    "Type": "CUSTOM",
    "LastUpdated": 1568938411,
    "Description": "New description text.",
    "Name": "SimpleMP4",
    "CreatedAt": 1568938240
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Preset MediaConvert Output AWS Elemental](#) di Panduan Pengguna AWS Elemental MediaConvert .

- Untuk detail API, lihat [UpdatePreset](#) di Referensi AWS CLI Perintah.

## update-queue

Contoh kode berikut menunjukkan cara menggunakan `update-queue`.

### AWS CLI

Untuk mengubah antrian

`update-queue` Contoh berikut menjeda antrian yang ditentukan, dengan mengubah statusnya menjadi `PAUSED`

```
aws mediaconvert update-queue \
--name Customer1 \
--status PAUSED
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
  "Queue": {
```

```
"LastUpdated": 1568839845,
"Status": "PAUSED",
"ProgressingJobsCount": 0,
"CreatedAt": 1526428516,
"Arn": "arn:aws:mediaconvert:us-west-1:123456789012:queues/Customer1",
"Name": "Customer1",
"SubmittedJobsCount": 0,
"PricingPlan": "ON_DEMAND",
"Type": "CUSTOM"
}
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan MediaConvert Antrian AWS Elemental di Panduan Pengguna AWS MediaConvert Elemental](#).

- Untuk detail API, lihat [UpdateQueue](#) di Referensi AWS CLI Perintah.

## MediaLive contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with MediaLive.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-channel**

Contoh kode berikut menunjukkan cara menggunakan `create-channel`.

## AWS CLI

Untuk membuat saluran

`create-channel` Contoh berikut membuat saluran dengan meneruskan file JSON yang berisi parameter yang ingin Anda tentukan.

Saluran dalam contoh ini menyerap input HLS PULL yang terhubung ke sumber yang berisi video, audio, dan teks yang disematkan. Saluran membuat satu grup keluaran HLS dengan server Akamai sebagai tujuan. Grup output berisi dua output: satu untuk video H.265 dan audio AAC, dan satu untuk keterangan Web-VTT, hanya dalam bahasa Inggris.

JSON untuk saluran contoh ini mencakup parameter minimum yang diperlukan untuk saluran yang menggunakan input HLS PULL dan yang menghasilkan grup keluaran HLS dengan Akamai sebagai tujuan. JSON berisi bagian utama ini:

`InputAttachments`, yang menentukan satu sumber untuk audio, dan satu sumber untuk teks. Itu tidak menentukan pemilih video, yang berarti MediaLive mengekstrak video pertama yang ditemukannya di sumbernya. `Destinations`, yang berisi dua alamat IP (URL) untuk grup keluaran tunggal di saluran ini. Alamat ini memerlukan kata sandi. `EncoderSettings`, yang berisi subbagian. `AudioDescriptions`, yang menentukan bahwa saluran berisi satu aset output audio, yang menggunakan sumber dari `InputAttachments`, dan menghasilkan audio dalam format AAC. `CaptionDescriptions`, yang menentukan bahwa saluran berisi satu aset keluaran teks, yang menggunakan sumber dari `InputAttachments`, dan menghasilkan teks dalam format Web-VTT. `VideoDescriptions`, yang menentukan bahwa saluran berisi satu aset keluaran video, dengan resolusi yang ditentukan. `OutputGroups`, yang menentukan kelompok output. Dalam contoh ini ada satu kelompok bernama `Akamai`. Koneksi dibuat menggunakan HLS PUT. Grup output berisi dua output. Satu output adalah untuk aset video ( bernama `Video_high` ) dan aset audio ( bernama `Audio_EN` ). Satu output adalah untuk aset teks ( bernama `WebVTT_EN` ).

Dalam contoh ini, beberapa parameter tidak mengandung nilai atau mengandung parameter kosong bersarang. Misalnya, `OutputSettings` untuk `Video_and_audio` output berisi beberapa parameter bersarang yang berakhir pada parameter kosong `M3U8Settings`. Parameter ini harus disertakan, tetapi Anda dapat menghilangkan satu, beberapa, atau semua turunannya, yang berarti bahwa anak akan mengambil nilai defaultnya atau menjadi nol.

Semua parameter yang berlaku untuk saluran contoh ini tetapi yang tidak ditentukan dalam file ini akan mengambil nilai default, disetel ke null, atau mengambil nilai unik yang dihasilkan oleh MediaLive.

```
aws medialive create-channel \  
  --cli-input-json file://channel-in-hls-out-hls-akamai.json
```

Isi dari `channel-in-hls-out-hls-akamai.json`:

```
{  
  "Name": "News_West",  
  "RoleArn": "arn:aws:iam::111122223333:role/MediaLiveAccessRole",  
  "InputAttachments": [  
    {  
      "InputAttachmentName": "local_news",  
      "InputId": "1234567",  
      "InputSettings": {  
        "AudioSelectors": [  
          {  
            "Name": "English-Audio",  
            "SelectorSettings": {  
              "AudioLanguageSelection": {  
                "LanguageCode": "EN"  
              }  
            }  
          }  
        ],  
        "CaptionSelectors": [  
          {  
            "LanguageCode": "ENE",  
            "Name": "English_embedded"  
          }  
        ]  
      }  
    }  
  ],  
  "Destinations": [  
    {  
      "Id": "akamai-server-west",  
      "Settings": [  
        {  
          "PasswordParam": "/medialive/examplecorp1",  
          "Url": "http://203.0.113.55/news/news_west",  
          "Username": "examplecorp"  
        },  
        {  
          "PasswordParam": "/medialive/examplecorp2",
```

```
        "Url": "http://203.0.113.82/news/news_west",
        "Username": "examplecorp"
    }
]
],
"EncoderSettings": {
    "AudioDescriptions": [
        {
            "AudioSelectorName": "English-Audio",
            "CodecSettings": {
                "AacSettings": {}
            },
            "Name": "Audio_EN"
        }
    ],
    "CaptionDescriptions": [
        {
            "CaptionSelectorName": "English_embedded",
            "DestinationSettings": {
                "WebvttDestinationSettings": {}
            },
            "Name": "WebVTT_EN"
        }
    ],
    "VideoDescriptions": [
        {
            "Height": 720,
            "Name": "Video_high",
            "Width": 1280
        }
    ],
    "OutputGroups": [
        {
            "Name": "Akamai",
            "OutputGroupSettings": {
                "HlsGroupSettings": {
                    "Destination": {
                        "DestinationRefId": "akamai-server-west"
                    },
                    "HlsCdnSettings": {
                        "HlsBasicPutSettings": {}
                    }
                }
            }
        }
    ]
}
```

```
    },
    "Outputs": [
      {
        "AudioDescriptionNames": [
          "Audio_EN"
        ],
        "OutputName": "Video_and_audio",
        "OutputSettings": {
          "HlsOutputSettings": {
            "HlsSettings": {
              "StandardHlsSettings": {
                "M3u8Settings": {}
              }
            },
            "NameModifier": "_1"
          }
        },
        "VideoDescriptionName": "Video_high"
      },
      {
        "CaptionDescriptionNames": [
          "WebVTT_EN"
        ],
        "OutputName": "Captions-WebVTT",
        "OutputSettings": {
          "HlsOutputSettings": {
            "HlsSettings": {
              "StandardHlsSettings": {
                "M3u8Settings": {}
              }
            },
            "NameModifier": "_2"
          }
        }
      }
    ]
  },
  "TimecodeConfig": {
    "Source": "EMBEDDED"
  }
}
```

## Output:

Output mengulangi kembali isi file JSON, ditambah nilai-nilai berikut. Semua parameter diurutkan menurut abjad.

ARN untuk saluran. Bagian terakhir dari ARN adalah ID saluran unik. `EgressEndpoints` kosong di saluran contoh ini karena hanya digunakan untuk input PUSH. Ketika itu berlaku, itu menunjukkan alamat pada konten MediaLive yang didorong ke `OutputGroups`, `Outputs`. Ini menunjukkan semua parameter untuk grup keluaran dan output, termasuk yang tidak Anda sertakan tetapi relevan dengan saluran ini. Parameter mungkin kosong (mungkin menunjukkan parameter atau fitur dinonaktifkan dalam konfigurasi saluran ini) atau mungkin menampilkan nilai default yang akan berlaku. `LogLevel` diatur ke default (DISABLED). `Tags` diatur ke default (null). `PipelinesRunningCount` dan `State` menunjukkan status saluran saat ini.

Untuk informasi selengkapnya, lihat [Membuat Saluran dari Awal](#) di Panduan MediaLive Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateChannel](#) di Referensi AWS CLI Perintah.

## create-input

Contoh kode berikut menunjukkan cara menggunakan `create-input`.

### AWS CLI

Untuk membuat masukan

`create-input` Contoh berikut membuat HLS PULL input dengan meneruskan file JSON yang berisi parameter yang berlaku untuk jenis input ini. JSON untuk masukan contoh ini menentukan dua sumber (alamat) ke input, untuk mendukung redundansi dalam ingest. Alamat ini memerlukan kata sandi.

```
aws medialive create-input \  
  --cli-input-json file://input-hls-pull-news.json
```

Isi dari `input-hls-pull-news.json`:

```
{  
  "Name": "local_news",  
  "RequestId": "cli000059",  
  "Sources": [  
    {  
      "Destination": "s3",  
      "Format": "HLS",  
      "IngestEndpoint": "s3://mybucket.us-east-1.amazonaws.com",  
      "Protocol": "HLS",  
      "S3Bucket": "mybucket",  
      "S3Prefix": "mybucket-us-east-1",  
      "S3Region": "us-east-1",  
      "S3StorageClass": "STANDARD",  
      "StreamName": "mybucket-us-east-1",  
      "StreamSpecification": "HLS_PULL",  
      "StreamType": "PULL",  
      "StreamTypeOptions": {"Redundancy": "REDUNDANT"},  
      "StreamTypeOptions2": {"Redundancy": "REDUNDANT"},  
      "StreamTypeOptions3": {"Redundancy": "REDUNDANT"}  
    }  
  ]  
}
```



```
    {
      "Url": "https://203.0.113.13/newschannel/anytownusa.m3u8",
      "Username": "examplecorp",
      "PasswordParam": "/medialive/examplecorp1"
    },
    {
      "Url": "https://198.51.100.54/fillervideos/oceanwaves.mp4",
      "Username": "examplecorp",
      "PasswordParam": "examplecorp2"
    }
  ],
  "Type": "URL_PULL"
}
```

### Output:

Output mengulangi kembali isi file JSON, ditambah nilai-nilai berikut. Semua parameter diurutkan menurut abjad.

Arn untuk masukan. Bagian terakhir dari ARN adalah ID input unik. `Attached Channels`, yang selalu kosong untuk input yang baru dibuat. `Destinations`, yang kosong dalam contoh ini karena hanya digunakan dengan input PUSH. `Id` untuk input, sama dengan ID di ARN. `MediaConnectFlows`, yang kosong dalam contoh ini karena hanya digunakan dengan input tipe `MediaConnect`. `SecurityGroups`, yang kosong dalam contoh ini karena hanya digunakan dengan input PUSH. `Stated` dari masukan ini. `Tags`, yang kosong (default untuk parameter ini).

Untuk informasi selengkapnya, lihat [Membuat Input](#) di Panduan MediaLive Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateInput](#) di Referensi AWS CLI Perintah.

## MediaPackage contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with `MediaPackage`.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **create-channel**

Contoh kode berikut menunjukkan cara menggunakan `create-channel`.

## AWS CLI

Untuk membuat saluran

`create-channel` Perintah berikut membuat saluran bernama `sportschannel` di akun saat ini.

```
aws mediapackage create-channel --id sportschannel
```

## Output:

```
{
  "Arn": "arn:aws:mediapackage:us-west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0",
  "HlsIngest": {
    "IngestEndpoints": [
      {
        "Id": "6d345804ec3f46c9b454a91d4a80d0e0",
        "Password": "generatedwebdavpassword1",
        "Url": "https://f31c86aed53b815a.mediapackage.us-west-2.amazonaws.com/in/v2/6d345804ec3f46c9b454a91d4a80d0e0/6d345804ec3f46c9b454a91d4a80d0e0/channel",
        "Username": "generatedwebdavusername1"
      },
      {
        "Id": "2daa32878af24803b24183727211b8ff",
        "Password": "generatedwebdavpassword2",
        "Url": "https://6ebbe7e04c4b0afa.mediapackage.us-west-2.amazonaws.com/in/v2/6d345804ec3f46c9b454a91d4a80d0e0/2daa32878af24803b24183727211b8ff/channel",
        "Username": "generatedwebdavusername2"
      }
    ]
  }
}
```

```
    }
  ]
},
"Id": "sportschannel",
"Tags": {
  "region": "west"
}
}
```

Untuk informasi selengkapnya, lihat [Membuat Saluran](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateChannel](#) di Referensi AWS CLI Perintah.

## create-origin-endpoint

Contoh kode berikut menunjukkan cara menggunakan `create-origin-endpoint`.

### AWS CLI

Untuk membuat titik akhir asal

`create-origin-endpoint` Perintah berikut membuat titik akhir asal bernama `cmaf_sports` dengan pengaturan paket yang disediakan dalam file JSON dan pengaturan titik akhir yang ditentukan.

```
aws mediapackage create-origin-endpoint \
  --channel-id sportschannel \
  --id cmaf_sports \
  --cmf-package file:///file/path/cmafpkg.json --description "cmf output of
sports" \
  --id cmf_sports \
  --manifest-name sports_channel \
  --startover-window-seconds 300 \
  --tags region=west,media=sports \
  --time-delay-seconds 10
```

Output:

```
{
  "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
  "ChannelId": "sportschannel",
```

```

"CmafPackage": {
  "HlsManifests": [
    {
      "AdMarkers": "PASSTHROUGH",
      "Id": "cmaf_sports_endpoint",
      "IncludeIframeOnlyStream": true,
      "ManifestName": "index",
      "PlaylistType": "EVENT",
      "PlaylistWindowSeconds": 300,
      "ProgramDateTimeIntervalSeconds": 300,
      "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
    }
  ],
  "SegmentDurationSeconds": 2,
  "SegmentPrefix": "sportschannel"
},
"Description": "cmaf output of sports",
"Id": "cmaf_sports",
"ManifestName": "sports_channel",
"StartoverWindowSeconds": 300,
"Tags": {
  "region": "west",
  "media": "sports"
},
"TimeDelaySeconds": 10,
"Url": "",
"Whitelist": []
}

```

Untuk informasi selengkapnya, lihat [Membuat Titik Akhir](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateOriginEndpoint](#) di Referensi AWS CLI Perintah.

## delete-channel

Contoh kode berikut menunjukkan cara menggunakan `delete-channel`.

### AWS CLI

Untuk menghapus saluran

`delete-channel` Perintah berikut menghapus saluran bernama `test`.

```
aws mediapackage delete-channel \  
  --id test
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Saluran](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeleteChannel](#) di Referensi AWS CLI Perintah.

## **delete-origin-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `delete-origin-endpoint`.

AWS CLI

Untuk menghapus titik akhir asal

`delete-origin-endpoint` Perintah berikut menghapus titik akhir asal bernama `tester2`

```
aws mediapackage delete-origin-endpoint \  
  --id tester2
```

Untuk informasi selengkapnya, lihat [Menghapus Titik Akhir di Panduan Pengguna AWS Elemental MediaPackage](#).

- Untuk detail API, lihat [DeleteOriginEndpoint](#) di Referensi AWS CLI Perintah.

## **describe-channel**

Contoh kode berikut menunjukkan cara menggunakan `describe-channel`.

AWS CLI

Untuk mendeskripsikan saluran

`describe-channel` Perintah berikut menampilkan semua rincian saluran bernama `test`.

```
aws mediapackage describe-channel \  
  --id test
```

**Output:**

```
{
  "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/584797f1740548c389a273585dd22a63",
  "HlsIngest": {
    "IngestEndpoints": [
      {
        "Id": "584797f1740548c389a273585dd22a63",
        "Password": "webdavgeneratedpassword1",
        "Url": "https://9be9c4405c474882.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
        "Username": "webdavgeneratedusername1"
      },
      {
        "Id": "7d187c8616fd455f88aaa5a9fcf74442",
        "Password": "webdavgeneratedpassword2",
        "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
        "Username": "webdavgeneratedusername2"
      }
    ]
  },
  "Id": "test",
  "Tags": {}
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Saluran](https://docs.aws.amazon.com/mediapackage/latest/ug/channels-view.html) < <https://docs.aws.amazon.com/mediapackage/latest/ug/channels-view.html> > di [MediaPackage Panduan Pengguna AWS Elemental](#)

- Untuk detail API, lihat [DescribeChannel](#) di Referensi AWS CLI Perintah.

**describe-origin-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `describe-origin-endpoint`.

**AWS CLI**

Untuk menggambarkan titik akhir asal

`describe-origin-endpoint` Perintah berikut menampilkan semua rincian dari titik akhir asal bernama `cmaf_sports`.

```
aws mediapackage describe-origin-endpoint \
  --id cmaf_sports
```

Output:

```
{
  "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
  "ChannelId": "sportschannel",
  "CmafPackage": {
    "HlsManifests": [
      {
        "AdMarkers": "NONE",
        "Id": "cmaf_sports_endpoint",
        "IncludeIframeOnlyStream": false,
        "PlaylistType": "EVENT",
        "PlaylistWindowSeconds": 60,
        "ProgramDateTimeIntervalSeconds": 0,
        "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
      }
    ],
    "SegmentDurationSeconds": 2,
    "SegmentPrefix": "sportschannel"
  },
  "Id": "cmaf_sports",
  "ManifestName": "index",
  "StartoverWindowSeconds": 0,
  "Tags": {
    "region": "west",
    "media": "sports"
  },
  "TimeDelaySeconds": 0,
  "Url": "",
  "Whitelist": []
}
```

Untuk informasi selengkapnya, lihat [Melihat Titik Akhir Tunggal](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribeOriginEndpoint](#) di Referensi AWS CLI Perintah.

## list-channels

Contoh kode berikut menunjukkan cara menggunakan `list-channels`.

### AWS CLI

Untuk daftar semua saluran

`list-channels` Perintah berikut mencantumkan semua saluran yang dikonfigurasi pada AWS akun saat ini.

```
aws mediapackage list-channels
```

Output:

```
{
  "Channels": [
    {
      "Arn": "arn:aws:mediapackage:us-west-2:111222333:channels/584797f1740548c389a273585dd22a63",
      "HlsIngest": {
        "IngestEndpoints": [
          {
            "Id": "584797f1740548c389a273585dd22a63",
            "Password": "webdavgeneratedpassword1",
            "Url": "https://9be9c4405c474882.mediapackage.us-west-2.amazonaws.com/in/v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
            "Username": "webdavgeneratedusername1"
          },
          {
            "Id": "7d187c8616fd455f88aaa5a9fcf74442",
            "Password": "webdavgeneratedpassword2",
            "Url": "https://7bf454c57220328d.mediapackage.us-west-2.amazonaws.com/in/v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
            "Username": "webdavgeneratedusername2"
          }
        ]
      }
    }
  ]
}
```



```
    ]
    },
    "Id": "test",
    "Tags": {}
  }
]
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Saluran](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [ListChannels](#) di Referensi AWS CLI Perintah.

## list-origin-endpoints

Contoh kode berikut menunjukkan cara menggunakan `list-origin-endpoints`.

### AWS CLI

Untuk mencantumkan semua titik akhir asal pada saluran

`list-origin-endpoints` Perintah berikut mencantumkan semua titik akhir asal yang dikonfigurasi pada saluran bernama `test`.

```
aws mediapackage list-origin-endpoints \
  --channel-id test
```

### Output:

```
{
  "OriginEndpoints": [
    {
      "Arn": "arn:aws:mediapackage:us-west-2:111222333:origin_endpoints/247cff871f2845d3805129be22f2c0a2",
      "ChannelId": "test",
      "DashPackage": {
        "ManifestLayout": "FULL",
        "ManifestWindowSeconds": 60,
        "MinBufferTimeSeconds": 30,
        "MinUpdatePeriodSeconds": 15,
        "PeriodTriggers": [],
        "Profile": "NONE",
        "SegmentDurationSeconds": 2,

```

```
    "SegmentTemplateFormat": "NUMBER_WITH_TIMELINE",
    "StreamSelection": {
      "MaxVideoBitsPerSecond": 2147483647,
      "MinVideoBitsPerSecond": 0,
      "StreamOrder": "ORIGINAL"
    },
    "SuggestedPresentationDelaySeconds": 25
  },
  "Id": "tester2",
  "ManifestName": "index",
  "StartoverWindowSeconds": 0,
  "Tags": {},
  "TimeDelaySeconds": 0,
  "Url": "https://8343f7014c0ea438.mediapackage.us-west-2.amazonaws.com/
out/v1/247cff871f2845d3805129be22f2c0a2/index.mpd",
  "Whitelist": []
},
{
  "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/869e237f851549e9bcf10e3bc2830839",
  "ChannelId": "test",
  "HlsPackage": {
    "AdMarkers": "NONE",
    "IncludeIframeOnlyStream": false,
    "PlaylistType": "EVENT",
    "PlaylistWindowSeconds": 60,
    "ProgramDateTimeIntervalSeconds": 0,
    "SegmentDurationSeconds": 6,
    "StreamSelection": {
      "MaxVideoBitsPerSecond": 2147483647,
      "MinVideoBitsPerSecond": 0,
      "StreamOrder": "ORIGINAL"
    },
    "UseAudioRenditionGroup": false
  },
  "Id": "tester",
  "ManifestName": "index",
  "StartoverWindowSeconds": 0,
  "Tags": {},
  "TimeDelaySeconds": 0,
  "Url": "https://8343f7014c0ea438.mediapackage.us-west-2.amazonaws.com/
out/v1/869e237f851549e9bcf10e3bc2830839/index.m3u8",
  "Whitelist": []
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Melihat semua Titik Akhir yang Terkait dengan Saluran](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [ListOriginEndpoints](#) di Referensi AWS CLI Perintah.

## **list-tags-for-resource**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang ditetapkan ke sumber daya

`list-tags-for-resource` Perintah berikut mencantumkan tag yang ditetapkan ke sumber daya yang ditentukan.

```
aws mediapackage list-tags-for-resource \
  --resource-arn arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0
```

Output:

```
{
  "Tags": {
    "region": "west"
  }
}
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya di AWS Elemental MediaPackage](#) dalam Panduan Pengguna AWS Elemental MediaPackage .

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **rotate-ingest-endpoint-credentials**

Contoh kode berikut menunjukkan cara menggunakan `rotate-ingest-endpoint-credentials`.

## AWS CLI

Untuk memutar kredensial ingest

`rotate-ingest-endpoint-credentials` Perintah berikut memutar nama pengguna dan kata sandi WebDAV untuk titik akhir ingest yang ditentukan.

```
aws mediapackage rotate-ingest-endpoint-credentials \  
  --id test \  
  --ingest-endpoint-id 584797f1740548c389a273585dd22a63
```

Output:

```
{  
  "Arn": "arn:aws:mediapackage:us-west-2:111222333:channels/584797f1740548c389a273585dd22a63",  
  "HlsIngest": {  
    "IngestEndpoints": [  
      {  
        "Id": "584797f1740548c389a273585dd22a63",  
        "Password": "webdavregeneratedpassword1",  
        "Url": "https://9be9c4405c474882.mediapackage.us-west-2.amazonaws.com/in/v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",  
        "Username": "webdavregeneratedusername1"  
      },  
      {  
        "Id": "7d187c8616fd455f88aaa5a9fcf74442",  
        "Password": "webdavgeneratedpassword2",  
        "Url": "https://7bf454c57220328d.mediapackage.us-west-2.amazonaws.com/in/v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",  
        "Username": "webdavgeneratedusername2"  
      }  
    ]  
  },  
  "Id": "test",  
  "Tags": {}  
}
```

Untuk informasi selengkapnya, lihat [Memutar Kredensial pada URL Input](#) di Panduan Pengguna AWS Elemental MediaPackage .

- Untuk detail API, lihat [RotateIngestEndpointCredentials](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tanda ke sumber daya

`tag-resource` Perintah berikut menambahkan pasangan `region=west` kunci dan nilai ke sumber daya yang ditentukan.

```
aws mediapackage tag-resource \  
  --resource-arn arn:aws:mediapackage:us-  
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0 \  
  --tags region=west
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya di AWS Elemental MediaPackage](#) dalam Panduan Pengguna AWS Elemental MediaPackage .

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Perintah berikut menghapus tag dengan kunci `region` dari saluran yang ditentukan.

```
aws mediapackage untag-resource \  
  --resource-arn arn:aws:mediapackage:us-  
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0 \  
  --tag-keys region
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya di AWS Elemental MediaPackage](#) dalam Panduan Pengguna AWS Elemental MediaPackage .

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-channel

Contoh kode berikut menunjukkan cara menggunakan `update-channel`.

### AWS CLI

Untuk memperbarui saluran

`update-channel` Perintah berikut memperbarui saluran bernama `sportschannel` untuk menyertakan deskripsi `24x7 sports`.

```
aws mediapackage update-channel \  
  --id sportschannel \  
  --description "24x7 sports"
```

### Output:

```
{  
  "Arn": "arn:aws:mediapackage:us-  
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0",  
  "Description": "24x7 sports",  
  "HlsIngest": {  
    "IngestEndpoints": [  
      {  
        "Id": "6d345804ec3f46c9b454a91d4a80d0e0",  
        "Password": "generatedwebdavpassword1",  
        "Url": "https://f31c86aed53b815a.mediapackage.us-  
west-2.amazonaws.com/in/  
v2/6d345804ec3f46c9b454a91d4a80d0e0/6d345804ec3f46c9b454a91d4a80d0e0/channel",  
        "Username": "generatedwebdavusername1"  
      },  
      {  
        "Id": "2daa32878af24803b24183727211b8ff",  
        "Password": "generatedwebdavpassword2",  
        "Url": "https://6ebbe7e04c4b0afa.mediapackage.us-  
west-2.amazonaws.com/in/  
v2/6d345804ec3f46c9b454a91d4a80d0e0/2daa32878af24803b24183727211b8ff/channel",
```

```

        "Username": "generatedwebdavusername2"
      }
    ]
  },
  "Id": "sportschannel",
  "Tags": {}
}

```

Untuk informasi selengkapnya, lihat [Mengedit Saluran](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [UpdateChannel](#) di Referensi AWS CLI Perintah.

## update-origin-endpoint

Contoh kode berikut menunjukkan cara menggunakan `update-origin-endpoint`.

### AWS CLI

Untuk memperbarui titik akhir asal

`update-origin-endpoint` Perintah berikut memperbarui titik akhir asal bernama `cmaf_sports`. Ini mengubah waktu tunda menjadi `0` detik.

```

aws mediapackage update-origin-endpoint \
  --id cmaf_sports \
  --time-delay-seconds 0

```

### Output:

```

{
  "Arn": "arn:aws:mediapackage:us-west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
  "ChannelId": "sportschannel",
  "CmafPackage": {
    "HlsManifests": [
      {
        "AdMarkers": "NONE",
        "Id": "cmaf_sports_endpoint",
        "IncludeIframeOnlyStream": false,
        "PlaylistType": "EVENT",
        "PlaylistWindowSeconds": 60,

```

```
        "ProgramDateTimeIntervalSeconds": 0,
        "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
    }
],
"SegmentDurationSeconds": 2,
"SegmentPrefix": "sportschannel"
},
"Id": "cmaf_sports",
"ManifestName": "index",
"StartoverWindowSeconds": 0,
"Tags": {
    "region": "west",
    "media": "sports"
},
"TimeDelaySeconds": 0,
"Url": "",
"Whitelist": []
}
```

Untuk informasi selengkapnya, lihat [Mengedit Titik Akhir](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [UpdateOriginEndpoint](#) di Referensi AWS CLI Perintah.

## MediaPackage Contoh VOD menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with MediaPackage VOD.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik



- [Tindakan](#)

## Tindakan

### **create-asset**

Contoh kode berikut menunjukkan cara menggunakan `create-asset`.

## AWS CLI

Untuk membuat aset

`create-asset` Contoh berikut membuat aset bernama `Chicken_Asset` di AWS akun saat ini. Aset menyerap file `30sec_chicken.smil` ke `MediaPackage`.

```
aws mediapackage-vod create-asset \  
  --id chicken_asset \  
  --packaging-group-id hls_chicken_gp \  
  --source-role-arn arn:aws:iam::111122223333:role/EMP_Vod \  
  --source-arn arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil
```

Output:

```
{  
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:assets/chicken_asset",  
  "Id": "chicken_asset",  
  "PackagingGroupId": "hls_chicken_gp",  
  "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",  
  "SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod",  
  "EgressEndpoints": [  
    {  
      "PackagingConfigurationId": "New_config_1",  
      "Url": "https://c75ea2668ab49d02bca7ae10ef31c59e.egress.mediapackage-  
vod.us-west-2.amazonaws.com/out/  
v1/6644b55df1744261ab3732a8e5cdaf07/904b06a58c7645e08d57d40d064216ac/  
f5b2e633ff4942228095d164c10074f3/index.m3u8"  
    },  
    {  
      "PackagingConfigurationId": "new_hls",  
      "Url": "https://c75ea2668ab49d02bca7ae10ef31c59e.egress.mediapackage-  
vod.us-west-2.amazonaws.com/out/v1/6644b55df1744261ab3732a8e5cdaf07/  
fe8f1f00a80e424cb4f8da4095835e9e/7370ec57432343af816332356d2bd5c6/string.m3u8"  
    }  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Mengkonsumsi Aset](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [CreateAsset](#) di Referensi AWS CLI Perintah.

## create-packaging-configuration

Contoh kode berikut menunjukkan cara menggunakan `create-packaging-configuration`.

### AWS CLI

Untuk membuat konfigurasi kemasan

`create-packaging-configuration` Contoh berikut membuat konfigurasi kemasan bernama `new_hls` dalam kelompok kemasan bernama `hls_chicken`. Contoh ini menggunakan file pada disk bernama `hls_pc.json` untuk memberikan rincian.

```
aws mediapackage-vod create-packaging-configuration \
  --id new_hls \
  --packaging-group-id hls_chicken \
  --hls-package file://hls_pc.json
```

Isi dari `hls_pc.json`:

```
{
  "HlsManifests":[
    {
      "AdMarkers":"NONE",
      "IncludeIframeOnlyStream":false,
      "ManifestName":"string",
      "ProgramDateTimeIntervalSeconds":60,
      "RepeatExtXKey":true,
      "StreamSelection":{"
        "MaxVideoBitsPerSecond":1000,
        "MinVideoBitsPerSecond":0,
        "StreamOrder":"ORIGINAL"
      }
    }
  ],
```

```

    "SegmentDurationSeconds":6,
    "UseAudioRenditionGroup":false
  }

```

Output:

```

{
  "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/new_hls",
  "Id":"new_hls",
  "PackagingGroupId":"hls_chicken",
  "HlsManifests":{
    "SegmentDurationSeconds":6,
    "UseAudioRenditionGroup":false,
    "HlsMarkers":[
      {
        "AdMarkers":"NONE",
        "IncludeIframeOnlyStream":false,
        "ManifestName":"string",
        "ProgramDateTimeIntervalSeconds":60,
        "RepeatExtXKey":true,
        "StreamSelection":{
          "MaxVideoBitsPerSecond":1000,
          "MinVideoBitsPerSecond":0,
          "StreamOrder":"ORIGINAL"
        }
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat Konfigurasi Kemasan](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [CreatePackagingConfiguration](#) di Referensi AWS CLI Perintah.

## create-packaging-group

Contoh kode berikut menunjukkan cara menggunakan `create-packaging-group`.

### AWS CLI

Untuk membuat grup kemasan

`create-packaging-group` Contoh berikut mencantumkan semua grup kemasan yang dikonfigurasi di AWS akun saat ini.

```
aws mediapackage-vod create-packaging-group \  
  --id hls_chicken
```

Output:

```
{  
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-groups/  
hls_chicken",  
  "Id": "hls_chicken"  
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Kemasan](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [CreatePackagingGroup](#) di Referensi AWS CLI Perintah.

## **delete-asset**

Contoh kode berikut menunjukkan cara menggunakan `delete-asset`.

### AWS CLI

Untuk menghapus aset

`delete-asset` Contoh berikut menghapus aset bernama `30sec_chicken`.

```
aws mediapackage-vod delete-asset \  
  --id 30sec_chicken
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Aset](#) di MediaPackage Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeleteAsset](#) di Referensi AWS CLI Perintah.

## **delete-packaging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-packaging-configuration`.

## AWS CLI

Untuk menghapus konfigurasi kemasan

`delete-packaging-configuration` Contoh berikut menghapus konfigurasi kemasan bernama `CMAF`.

```
aws mediapackage-vod delete-packaging-configuration \  
  --id CMAF
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Konfigurasi Kemasan](#) di *MediaPackage Panduan Pengguna AWS Elemental*.

- Untuk detail API, lihat [DeletePackagingConfiguration](#) di Referensi AWS CLI Perintah.

## `delete-packaging-group`

Contoh kode berikut menunjukkan cara menggunakan `delete-packaging-group`.

## AWS CLI

Untuk menghapus grup kemasan

`delete-packaging-group` Contoh berikut menghapus grup kemasan bernama `Dash_widevine`.

```
aws mediapackage-vod delete-packaging-group \  
  --id Dash_widevine
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Grup Kemasan](#) di *MediaPackage Panduan Pengguna AWS Elemental*.

- Untuk detail API, lihat [DeletePackagingGroup](#) di Referensi AWS CLI Perintah.

## `describe-asset`

Contoh kode berikut menunjukkan cara menggunakan `describe-asset`.

## AWS CLI

Untuk mendeskripsikan aset

`describe-asset` Contoh berikut menampilkan semua rincian aset bernama `30sec_chicken`.

```
aws mediapackage-vod describe-asset \
  --id 30sec_chicken
```

Output:

```
{
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:assets/30sec_chicken",
  "Id": "30sec_chicken",
  "PackagingGroupId": "Packaging_group_1",
  "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
  "SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod",
  "EgressEndpoints": [
    {
      "PackagingConfigurationId": "DASH",
      "Url": "https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/66c25aff456d463aae0855172b3beb27/4ddfda6da17c4c279a1b8401cb
index.mpd"
    },
    {
      "PackagingConfigurationId": "HLS",
      "Url": "https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/6e5bf286a3414254a2bf0d22ae148d7e/06b5875b4d004c3cbdc4da2dc4
index.m3u8"
    },
    {
      "PackagingConfigurationId": "CMAF",
      "Url": "https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/628fb5d8d89e4702958b020af27fde0e/05eb062214064238ad6330a443
index.m3u8"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Aset](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribeAsset](#) di Referensi AWS CLI Perintah.

## describe-packaging-configuration

Contoh kode berikut menunjukkan cara menggunakan `describe-packaging-configuration`.

### AWS CLI

Untuk menggambarkan konfigurasi kemasan

`describe-packaging-configuration` Contoh berikut menampilkan semua rincian konfigurasi kemasan bernama DASH.

```
aws mediapackage-vod describe-packaging-configuration \  
  --id DASH
```

Output:

```
{  
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/  
DASH",  
  "Id": "DASH",  
  "PackagingGroupId": "Packaging_group_1",  
  "DashPackage": [  
    {  
      "SegmentDurationSeconds": "2"  
    },  
    {  
      "DashManifests": {  
        "ManifestName": "index",  
        "MinBufferTimeSeconds": "30",  
        "Profile": "NONE"  
      }  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Konfigurasi Kemasan](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribePackagingConfiguration](#) di Referensi AWS CLI Perintah.

## describe-packaging-group

Contoh kode berikut menunjukkan cara menggunakan `describe-packaging-group`.

### AWS CLI

Untuk menggambarkan kelompok pengemasan

`describe-packaging-group` Contoh berikut menampilkan semua rincian kelompok kemasan bernama `Packaging_group_1`.

```
aws mediapackage-vod describe-packaging-group \
  --id Packaging_group_1
```

Output:

```
{
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-groups/
Packaging_group_1",
  "Id": "Packaging_group_1"
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Grup Kemasan](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribePackagingGroup](#) di Referensi AWS CLI Perintah.

## list-assets

Contoh kode berikut menunjukkan cara menggunakan `list-assets`.

### AWS CLI

Untuk daftar semua aset

`list-assets` Contoh berikut mencantumkan semua aset yang dikonfigurasi dalam AWS akun saat ini.

```
aws mediapackage-vod list-assets
```



**Output:**

```
{
  "Assets": [
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:assets/30sec_chicken",
      "Id": "30sec_chicken",
      "PackagingGroupId": "Packaging_group_1",
      "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
      "SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Aset](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [ListAssets](#) di Referensi AWS CLI Perintah.

**list-packaging-configurations**

Contoh kode berikut menunjukkan cara menggunakan `list-packaging-configurations`.

**AWS CLI**

Untuk membuat daftar semua konfigurasi kemasan

`list-packaging-configurations` Contoh berikut mencantumkan semua konfigurasi kemasan yang dikonfigurasi pada grup kemasan bernama `Packaging_group_1`.

```
aws mediapackage-vod list-packaging-configurations \
  --packaging-group-id Packaging_group_1
```

**Output:**

```
{
  "PackagingConfigurations": [
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/CMAF",
      "Id": "CMAF",
      "PackagingGroupId": "Packaging_group_1",
      "CmafPackage": [
```

```

        {
            "SegmentDurationSeconds":"2"
        },
        {
            "HlsManifests":{
                "AdMarkers":"NONE",
                "RepeatExtXKey":"False",
                "ManifestName":"index",
                "ProgramDateTimeIntervalSeconds":"0",
                "IncludeIframeOnlyStream":"False"
            }
        }
    ]
},
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/DASH",
    "Id":"DASH",
    "PackagingGroupId":"Packaging_group_1",
    "DashPackage":[
        {
            "SegmentDurationSeconds":"2"
        },
        {
            "DashManifests":{
                "ManifestName":"index",
                "MinBufferTimeSeconds":"30",
                "Profile":"NONE"
            }
        }
    ]
},
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/HLS",
    "Id":"HLS",
    "PackagingGroupId":"Packaging_group_1",
    "HlsPackage":[
        {
            "SegmentDurationSeconds":"6",
            "UseAudioRenditionGroup":"False"
        },
        {
            "HlsManifests":{

```

```

        "AdMarkers":"NONE",
        "RepeatExtXKey":"False",
        "ManifestName":"index",
        "ProgramDateTimeIntervalSeconds":"0",
        "IncludeIframeOnlyStream":"False"
    }
}
]
},
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/New_config_0_copy",
    "Id":"New_config_0_copy",
    "PackagingGroupId":"Packaging_group_1",
    "HlsPackage":[
        {
            "SegmentDurationSeconds":"6",
            "UseAudioRenditionGroup":"False"
        },
        {
            "Encryption":{
                "EncryptionMethod":"AWS_128",
                "SpekeKeyProvider":{
                    "RoleArn":"arn:aws:iam:111122223333::role/SPEKERole",
                    "Url":"https://lfgubdvs97.execute-api.us-
west-2.amazonaws.com/EkeStage/copyProtection/",
                    "SystemIds":[
                        "81376844-f976-481e-a84e-cc25d39b0b33"
                    ]
                }
            }
        },
        {
            "HlsManifests":{
                "AdMarkers":"NONE",
                "RepeatExtXKey":"False",
                "ManifestName":"index",
                "ProgramDateTimeIntervalSeconds":"0",
                "IncludeIframeOnlyStream":"False"
            }
        }
    ]
}
]
}
]

```

```
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Konfigurasi Kemasan](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [ListPackagingConfigurations](#) di Referensi AWS CLI Perintah.

## list-packaging-groups

Contoh kode berikut menunjukkan cara menggunakan `list-packaging-groups`.

### AWS CLI

Untuk membuat daftar semua kelompok kemasan

`list-packaging-groups` Contoh berikut mencantumkan semua grup kemasan yang dikonfigurasi di AWS akun saat ini.

```
aws mediapackage-vod list-packaging-groups
```

Output:

```
{
  "PackagingGroups": [
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Dash_widevine",
      "Id": "Dash_widevine"
    },
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Encrypted_HLS",
      "Id": "Encrypted_HLS"
    },
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Packaging_group_1",
      "Id": "Packaging_group_1"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Grup Kemasan](#) di Panduan MediaPackage Pengguna AWS Elemental.

- Untuk detail API, lihat [ListPackagingGroups](#) di Referensi AWS CLI Perintah.

## MediaStore Contoh Data Plane menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With MediaStore Data Plane.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-object**

Contoh kode berikut menunjukkan cara menggunakan `delete-object`.

AWS CLI

Untuk menghapus objek

`delete-object` Contoh berikut menghapus objek yang ditentukan.

```
aws mediastore-data delete-object \  
  --endpoint=https://aaabbbcccddee.data.mediastore.us-west-2.amazonaws.com \  
  --path=/folder_name/README.md
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Objek](#) di MediaStore Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeleteObject](#) di Referensi AWS CLI Perintah.

## describe-object

Contoh kode berikut menunjukkan cara menggunakan `describe-object`.

### AWS CLI

Untuk melihat header untuk objek

`describe-object` Contoh berikut menampilkan header untuk objek di jalur yang ditentukan.

```
aws mediastore-data describe-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path events/baseball/setup.jpg
```

Output:

```
{  
  "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT",  
  "ContentType": "image/jpeg",  
  "ContentLength": "3860266",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f5555555555555555da6d3"  
}
```

Untuk informasi selengkapnya, lihat [Melihat Detail Objek](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [DescribeObject](#) di Referensi AWS CLI Perintah.

## get-object

Contoh kode berikut menunjukkan cara menggunakan `get-object`.

### AWS CLI

Contoh 1: Untuk mengunduh seluruh objek

`get-object` Contoh berikut mengunduh objek yang ditentukan.

```
aws mediastore-data get-object \  
--endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \  
--path events/baseball/setup.jpg setup.jpg
```

Output:

```
{  
  "ContentType": "image/jpeg",  
  "StatusCode": 200,  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eaaaa4dd89ff7f555555555555da6d3",  
  "ContentLength": "3860266",  
  "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT"  
}
```

Contoh 2: Untuk men-download bagian dari objek

`get-object` Contoh berikut mendownload bagian tertentu dari suatu objek.

```
aws mediastore-data get-object \  
--endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \  
--path events/baseball/setup.jpg setup.jpg \  
--range "bytes=0-100"
```

Output:

```
{  
  "StatusCode": 206,  
  "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT",  
  "ContentType": "image/jpeg",  
  "ContentRange": "bytes 0-100/3860266",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eaaaa4dd89ff7f555555555555da6d3",  
  "ContentLength": "101"  
}
```

Untuk informasi selengkapnya, lihat [Mengunduh Objek](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [GetObject](#) di Referensi AWS CLI Perintah.

## list-items

Contoh kode berikut menunjukkan cara menggunakan `list-items`.

### AWS CLI

Contoh 1: Untuk melihat daftar item (objek dan folder) yang disimpan dalam wadah

`list-items` Contoh berikut menampilkan daftar item (objek dan folder) yang disimpan dalam wadah yang ditentukan.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com
```

Output:

```
{  
  "Items": [  
    {  
      "Type": "OBJECT",  
      "ContentLength": 3784,  
      "Name": "setup.jpg",  
      "ETag":  
      "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f5555555555555555da6d3",  
      "ContentType": "image/jpeg",  
      "LastModified": 1563571859.379  
    },  
    {  
      "Type": "FOLDER",  
      "Name": "events"  
    }  
  ]  
}
```

Contoh 2: Untuk melihat daftar item (objek dan folder) yang disimpan dalam folder

`list-items` Contoh berikut menampilkan daftar item (objek dan folder) yang disimpan dalam folder yang ditentukan.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --bucket-name <bucket-name>
```



```
--path events/baseball
```

### Output:

```
{
  "Items": [
    {
      "ETag":
"2aa333bbcc8d8d22d777e999c88d4aa9e4dd89ff7f5555555555555555da6d3",
      "ContentType": "image/jpeg",
      "Type": "OBJECT",
      "ContentLength": 3860266,
      "LastModified": 1563573031.872,
      "Name": "setup.jpg"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Daftar Objek](#) di Panduan MediaStore Pengguna AWS Elemental.

- Untuk detail API, lihat [ListItems](#) di Referensi AWS CLI Perintah.

## put-object

Contoh kode berikut menunjukkan cara menggunakan `put-object`.

### AWS CLI

Contoh 1: Untuk mengunggah objek ke wadah

`put-object` Contoh berikut meng-upload objek ke wadah tertentu.

```
aws mediastore-data put-object \
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \
  --body README.md \
  --path README.md \
  --cache-control "max-age=6, public" \
  --content-type binary/octet-stream
```

### Output:



Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-playback-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-playback-configuration`.

AWS CLI

Untuk menghapus konfigurasi

Berikut ini `delete-playback-configuration` menghapus konfigurasi bernam `campaign_short`.

```
aws mediatailor delete-playback-configuration \  
  --name campaign_short
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Konfigurasi](#) di MediaTailor Panduan Pengguna AWS Elemental.

- Untuk detail API, lihat [DeletePlaybackConfiguration](#) di Referensi AWS CLI Perintah.

### **get-playback-configuration**

Contoh kode berikut menunjukkan cara menggunakan `get-playback-configuration`.

AWS CLI

Untuk mendeskripsikan konfigurasi

Berikut ini `get-playback-configuration` menampilkan semua rincian konfigurasi bernam `west_campaign`.

```
aws mediatailor get-playback-configuration \  
  --name west_campaign
```

### Output:

```
{  
  "AdDecisionServerUrl": "http://your.ads.url",  
  "CdnConfiguration": {},  
  "DashConfiguration": {  
    "ManifestEndpointPrefix":  
    "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/  
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",  
    "MpdLocation": "EMT_DEFAULT",  
    "OriginManifestType": "MULTI_PERIOD"  
  },  
  "HlsConfiguration": {  
    "ManifestEndpointPrefix":  
    "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/  
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/"  
  },  
  "Name": "west_campaign",  
  "PlaybackConfigurationArn": "arn:aws:mediatailor:us-  
west-2:123456789012:playbackConfiguration/west_campaign",  
  "PlaybackEndpointPrefix":  
  "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com",  
  "SessionInitializationEndpointPrefix":  
  "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/  
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",  
  "Tags": {},  
  "VideoContentSourceUrl": "https://8343f7014c0ea438.mediapackage.us-  
west-2.amazonaws.com/out/v1/683f0f2ff7cd43a48902e6dcd5e16dcf/index.m3u8"  
}
```

Untuk informasi selengkapnya, lihat [Melihat Konfigurasi](#) di Panduan MediaTailor Pengguna AWS Elemental.

- Untuk detail API, lihat [GetPlaybackConfiguration](#) di Referensi AWS CLI Perintah.

## list-playback-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-playback-configurations`.

## AWS CLI

Untuk daftar semua konfigurasi

Berikut ini `list-playback-configurations` menampilkan semua detail konfigurasi pada AWS akun saat ini.

```
aws mediatailor list-playback-configurations
```

Output:

```
{
  "Items": [
    {
      "AdDecisionServerUrl": "http://your.ads.url",
      "CdnConfiguration": {},
      "DashConfiguration": {
        "ManifestEndpointPrefix":
          "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
          dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
        "MpdLocation": "EMT_DEFAULT",
        "OriginManifestType": "MULTI_PERIOD"
      },
      "HlsConfiguration": {
        "ManifestEndpointPrefix":
          "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
          master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/"
      },
      "Name": "west_campaign",
      "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
      west-2:123456789012:playbackConfiguration/west_campaign",
      "PlaybackEndpointPrefix":
        "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com",
      "SessionInitializationEndpointPrefix":
        "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
        session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
      "Tags": {},
      "VideoContentSourceUrl": "https://8343f7014c0ea438.mediapackage.us-
      west-2.amazonaws.com/out/v1/683f0f2ff7cd43a48902e6dcd5e16dcf/index.m3u8"
    },
    {
      "AdDecisionServerUrl": "http://your.ads.url",
      "CdnConfiguration": {},

```

```

    "DashConfiguration": {
      "ManifestEndpointPrefix":
        "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediataylor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/",
      "MpdLocation": "DISABLED",
      "OriginManifestType": "MULTI_PERIOD"
    },
    "HlsConfiguration": {
      "ManifestEndpointPrefix":
        "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediataylor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/"
    },
    "Name": "sports_campaign",
    "PlaybackConfigurationArn": "arn:aws:mediataylor:us-
west-2:123456789012:playbackConfiguration/sports_campaign",
    "PlaybackEndpointPrefix":
      "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediataylor.us-west-2.amazonaws.com",
    "SessionInitializationEndpointPrefix":
      "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediataylor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/",
    "SlateAdUrl": "http://s3.bucket/slate_ad.mp4",
    "Tags": {},
    "VideoContentSourceUrl": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/sports_endpoint/
index.m3u8"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat Konfigurasi](https://docs.aws.amazon.com/mediataylor/latest/ug/configurations-view.html) < <https://docs.aws.amazon.com/mediataylor/latest/ug/configurations-view.html> > di Panduan MediaTailor Pengguna AWS Elemental.

- Untuk detail API, lihat [ListPlaybackConfigurations](#) di Referensi AWS CLI Perintah.

## put-playback-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-playback-configuration`.

### AWS CLI

Untuk membuat konfigurasi

Berikut ini `put-playback-configuration` membuat konfigurasi bernama `campaign_short`.

```
aws mediatailor put-playback-configuration \  
  --name campaign_short \  
  --ad-decision-server-url http://your.ads.url \  
  --video-content-source-url http://video.bucket/index.m3u8
```

Output:

```
{  
  "AdDecisionServerUrl": "http://your.ads.url",  
  "CdnConfiguration": {},  
  "DashConfiguration": {  
    "ManifestEndpointPrefix":  
      "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/  
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/",  
    "MpdLocation": "EMT_DEFAULT",  
    "OriginManifestType": "MULTI_PERIOD"  
  },  
  "HlsConfiguration": {  
    "ManifestEndpointPrefix":  
      "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/  
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/"  
  },  
  "Name": "campaign_short",  
  "PlaybackConfigurationArn": "arn:aws:mediatailor:us-  
west-2:123456789012:playbackConfiguration/campaign_short",  
  "PlaybackEndpointPrefix":  
    "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com",  
  "SessionInitializationEndpointPrefix":  
    "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/  
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/",  
  "Tags": {},  
  "VideoContentSourceUrl": "http://video.bucket/index.m3u8"  
}
```

Untuk informasi selengkapnya, lihat [Membuat Konfigurasi](#) di Panduan MediaTailor Pengguna AWS Elemental.

- Untuk detail API, lihat [PutPlaybackConfiguration](#) di Referensi AWS CLI Perintah.

## Contoh MemoryDB menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda bagaimana melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface dengan MemoryDB.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **copy-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `copy-snapshot`.

AWS CLI

Untuk menyalin snapshot

`copy-snapshot` Contoh berikut membuat salinan snapshot.

```
aws memorydb copy-snapshot \  
  --source-snapshot-name my-cluster-snapshot \  
  --target-snapshot-name my-cluster-snapshot-copy
```

Output

```
{  
  "Snapshot": {  
    "Name": "my-cluster-snapshot-copy",
```



```

    "Status": "creating",
    "Source": "manual",
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:snapshot/my-cluster-
snapshot-copy",
    "ClusterConfiguration": {
      "Name": "my-cluster",
      "Description": " ",
      "NodeType": "db.r6g.large",
      "EngineVersion": "6.2",
      "MaintenanceWindow": "wed:03:00-wed:04:00",
      "Port": 6379,
      "ParameterGroupName": "default.memorydb-redis6",
      "SubnetGroupName": "my-sg",
      "VpcId": "vpc-xx2574fc",
      "SnapshotRetentionLimit": 0,
      "SnapshotWindow": "04:30-05:30",
      "NumShards": 2
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Menyalin snapshot](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [CopySnapshot](#) di Referensi AWS CLI Perintah.

## create-acl

Contoh kode berikut menunjukkan cara menggunakan `create-acl`.

### AWS CLI

Untuk membuat ACL

`create-acl` Contoh berikut membuat daftar kontrol Access baru.

```

aws memorydb create-acl \
  --acl-name "new-acl-1" \
  --user-names "my-user"

```

Output:

```
{
```

```

"ACL": {
  "Name": "new-acl-1",
  "Status": "creating",
  "UserNames": [
    "my-user"
  ],
  "MinimumEngineVersion": "6.2",
  "Clusters": [],
  "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/new-acl-1"
}
}

```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [CreateAcl](#) di Referensi AWS CLI Perintah.

## create-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-cluster`.

### AWS CLI

Untuk membuat cluster

`create-cluster` Contoh berikut membuat cluster baru.

```

aws memorydb create-cluster \
  --cluster-name my-new-cluster \
  --node-type db.r6g.large \
  --acl-name my-acl \
  --subnet-group my-sg

```

Output:

```

{
  "Cluster": {
    "Name": "my-new-cluster",
    "Status": "creating",
    "NumberOfShards": 1,
    "AvailabilityMode": "MultiAZ",
    "ClusterEndpoint": {

```

```

        "Port": 6379
    },
    "NodeType": "db.r6g.large",
    "EngineVersion": "6.2",
    "EnginePatchVersion": "6.2.6",
    "ParameterGroupName": "default.memorydb-redis6",
    "ParameterGroupStatus": "in-sync",
    "SubnetGroupName": "my-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:cluster/my-new-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "sat:10:00-sat:11:00",
    "SnapshotWindow": "07:30-08:30",
    "ACLName": "my-acl",
    "AutoMinorVersionUpgrade": true
}
}

```

Untuk informasi selengkapnya, lihat [Mengelola Cluster](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [CreateCluster](#) di Referensi AWS CLI Perintah.

## create-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `create-parameter-group`.

### AWS CLI

Untuk membuat grup parameter

`create-parameter-group` Contoh berikut membuat grup parameter.

```

aws memorydb create-parameter-group \
  --parameter-group-name myRedis6x \
  --family memorydb_redis6 \
  --description "my-parameter-group"

```

Output:

```

{
  "ParameterGroup": {
    "Name": "myredis6x",
    "Family": "memorydb_redis6",

```

```
    "Description": "my-parameter-group",
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:parametergroup/myredis6x"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat grup parameter](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [CreateParameterGroup](#) di Referensi AWS CLI Perintah.

## create-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-snapshot`.

### AWS CLI

Untuk membuat snapshot

`create-snapshot` Contoh berikut membuat snapshot.

```
aws memorydb create-snapshot \
  --cluster-name my-cluster \
  --snapshot-name my-cluster-snapshot
```

Output:

```
{
  "Snapshot": {
    "Name": "my-cluster-snapshot1",
    "Status": "creating",
    "Source": "manual",
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:snapshot/my-cluster-snapshot",
    "ClusterConfiguration": {
      "Name": "my-cluster",
      "Description": "",
      "NodeType": "db.r6g.large",
      "EngineVersion": "6.2",
      "MaintenanceWindow": "wed:03:00-wed:04:00",
      "Port": 6379,
      "ParameterGroupName": "default.memorydb-redis6",
      "SubnetGroupName": "my-sg",
      "VpcId": "vpc-862xxxxc",
      "SnapshotRetentionLimit": 0,

```

```

        "SnapshotWindow": "04:30-05:30",
        "NumShards": 2
    }
}

```

Untuk informasi selengkapnya, lihat [Membuat snapshot manual](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [CreateSnapshot](#) di Referensi AWS CLI Perintah.

## create-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `create-subnet-group`.

### AWS CLI

Untuk membuat grup subnet

`create-subnet-group` Contoh berikut membuat grup subnet.

```

aws memorydb create-subnet-group \
  --subnet-group-name mysubnetgroup \
  --description "my subnet group" \
  --subnet-ids subnet-5623xxxx

```

Output:

```

{
  "SubnetGroup": {
    "Name": "mysubnetgroup",
    "Description": "my subnet group",
    "VpcId": "vpc-86257xxx",
    "Subnets": [
      {
        "Identifier": "subnet-5623xxxx",
        "AvailabilityZone": {
          "Name": "us-east-1a"
        }
      }
    ],
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/mysubnetgroup"
  }
}

```

```
}
}
```

Untuk informasi selengkapnya, lihat [Membuat grup subnet di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [CreateSubnetGroup](#) di Referensi AWS CLI Perintah.

## create-user

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

### AWS CLI

Untuk membuat pengguna

`create-user` Contoh berikut menciptakan pengguna baru.

```
aws memorydb create-user \
  --user-name user-name-1 \
  --access-string "~objects:* ~items:* ~public:*" \
  --authentication-mode \
    Passwords="enterapasswordhere",Type=password
```

Output:

```
{
  "User": {
    "Name": "user-name-1",
    "Status": "active",
    "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
    "ACLNames": [],
    "MinimumEngineVersion": "6.2",
    "Authentication": {
      "Type": "password",
      "PasswordCount": 1
    },
    "ARN": "arn:aws:memorydb:us-west-2:491658xxxxxx:user/user-name-1"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## delete-acl

Contoh kode berikut menunjukkan cara menggunakan `delete-acl`.

### AWS CLI

Untuk menghapus ACL

`delete-acl` Contoh berikut menghapus daftar kontrol Access.

```
aws memorydb delete-acl \  
  --acl-name "new-acl-1"
```

Output:

```
{  
  "ACL": {  
    "Name": "new-acl-1",  
    "Status": "deleting",  
    "UserNames": [  
      "pat"  
    ],  
    "MinimumEngineVersion": "6.2",  
    "Clusters": [],  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/new-acl-1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DeleteAcl](#) di Referensi AWS CLI Perintah.

## delete-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster`.

### AWS CLI

Untuk menghapus klaster

`delete-cluster` Contoh berikut menghapus cluster.

```
aws memorydb delete-cluster \  
  --cluster-name my-new-cluster
```

Output:

```
{  
  "Cluster": {  
    "Name": "my-new-cluster",  
    "Status": "deleting",  
    "NumberOfShards": 1,  
    "ClusterEndpoint": {  
      "Address": "clustercfg.my-new-cluster.xxxxx.memorydb.us-  
east-1.amazonaws.com",  
      "Port": 6379  
    },  
    "NodeType": "db.r6g.large",  
    "EngineVersion": "6.2",  
    "EnginePatchVersion": "6.2.6",  
    "ParameterGroupName": "default.memorydb-redis6",  
    "ParameterGroupStatus": "in-sync",  
    "SubnetGroupName": "my-sg",  
    "TLSEnabled": true,  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-new-cluster",  
    "SnapshotRetentionLimit": 0,  
    "MaintenanceWindow": "sat:10:00-sat:11:00",  
    "SnapshotWindow": "07:30-08:30",  
    "AutoMinorVersionUpgrade": true  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus klaster di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DeleteCluster](#) di Referensi AWS CLI Perintah.

## **delete-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-parameter-group`.

### AWS CLI

Untuk menghapus grup parameter



`delete-parameter-group` Contoh berikut menghapus grup parameter.

```
aws memorydb delete-parameter-group \  
  --parameter-group-name myRedis6x
```

Output:

```
{  
  "ParameterGroup": {  
    "Name": "myredis6x",  
    "Family": "memorydb_redis6",  
    "Description": "my-parameter-group",  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/myredis6x"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus grup parameter di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DeleteParameterGroup](#) di Referensi AWS CLI Perintah.

## **delete-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `delete-snapshot`.

AWS CLI

Untuk menghapus snapshot

`delete-snapshot` Contoh berikut menghapus snapshot.

```
aws memorydb delete-snapshot \  
  --snapshot-name my-cluster-snapshot
```

Output:

```
{  
  "Snapshot": {  
    "Name": "my-cluster-snapshot",  
    "Status": "deleting",  
    "Source": "manual",
```

```

    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:snapshot/my-cluster-
snapshot",
    "ClusterConfiguration": {
      "Name": "my-cluster",
      "Description": "",
      "NodeType": "db.r6g.large",
      "EngineVersion": "6.2",
      "MaintenanceWindow": "wed:03:00-wed:04:00",
      "Port": 6379,
      "ParameterGroupName": "default.memorydb-redis6",
      "SubnetGroupName": "my-sg",
      "VpcId": "vpc-862xxxxc",
      "SnapshotRetentionLimit": 0,
      "SnapshotWindow": "04:30-05:30",
      "NumShards": 2
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Menghapus snapshot](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [DeleteSnapshot](#) di Referensi AWS CLI Perintah.

## delete-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `delete-subnet-group`.

### AWS CLI

Untuk menghapus grup subnet

`delete-subnet-group` Contoh berikut menghapus subnet.

```
aws memorydb delete-subnet-group \
  --subnet-group-name mysubnetgroup
```

Output:

```
{
  "SubnetGroup": {
    "Name": "mysubnetgroup",
    "Description": "my subnet group",
    "VpcId": "vpc-86xxxx4fc",

```

```

    "Subnets": [
      {
        "Identifier": "subnet-56xxx61b",
        "AvailabilityZone": {
          "Name": "us-east-1a"
        }
      }
    ],
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/mysubnetgroup"
  }
}

```

Untuk informasi selengkapnya, lihat [Menghapus grup subnet](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [DeleteSubnetGroup](#) di Referensi AWS CLI Perintah.

## delete-user

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

### AWS CLI

Untuk menghapus pengguna

`delete-user` Contoh berikut menghapus pengguna.

```
aws memorydb delete-user \
  --user-name my-user
```

### Output:

```

{
  "User": {
    "Name": "my-user",
    "Status": "deleting",
    "AccessString": "on ~app:* resetchannels -@all +@read",
    "ACLNames": [
      "my-acl"
    ],
    "MinimumEngineVersion": "6.2",
    "Authentication": {
      "Type": "password",
      "PasswordCount": 1
    }
  }
}

```

```
    },
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## describe-acls

Contoh kode berikut menunjukkan cara menggunakan `describe-acls`.

### AWS CLI

Untuk mengembalikan daftar ACL

Berikut ini `describe-acls` mengembalikan daftar ACL.

```
aws memorydb describe-acls
```

Output:

```
{
  "ACLs": [
    {
      "Name": "open-access",
      "Status": "active",
      "UserNames": [
        "default"
      ],
      "MinimumEngineVersion": "6.2",
      "Clusters": [],
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/open-access"
    },
    {
      "Name": "my-acl",
      "Status": "active",
      "UserNames": [],
      "MinimumEngineVersion": "6.2",
      "Clusters": [
```

```
        "my-cluster"
      ],
      "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxxx:acl/my-acl"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DescribeAcls](#) di Referensi AWS CLI Perintah.

## describe-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-clusters`.

### AWS CLI

Untuk mengembalikan daftar cluster

Berikut ini menjelaskan `clusters`` mengembalikan daftar cluster.

```
aws memorydb describe-clusters
```

Output:

```
{
  "Clusters": [
    {
      "Name": "my-cluster",
      "Status": "available",
      "NumberOfShards": 2,
      "ClusterEndpoint": {
        "Address": "clustercfg.my-cluster.1lru6f.memorydb.us-
east-1.amazonaws.com",
        "Port": 6379
      },
      "NodeType": "db.r6g.large",
      "EngineVersion": "6.2",
      "EnginePatchVersion": "6.2.6",
      "ParameterGroupName": "default.memorydb-redis6",
      "ParameterGroupStatus": "in-sync",
    }
  ]
}
```

```
    "SecurityGroups": [
      {
        "SecurityGroupId": "sg-0a1434xxxxxc9fae",
        "Status": "active"
      }
    ],
    "SubnetGroupName": "pat-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:cluster/my-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "ACLName": "my-acl",
    "AutoMinorVersionUpgrade": true
  }
]
```

Untuk informasi selengkapnya, lihat [Mengelola cluster di Panduan](#) Pengguna MemoryDB.

- Untuk detail API, lihat [DescribeClusters](#) di Referensi AWS CLI Perintah.

## describe-engine-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-engine-versions`.

### AWS CLI

Untuk mengembalikan daftar versi mesin

Berikut `describe-engine-versions` `mengembalikan daftar versi mesin.

```
aws memorydb describe-engine-versions
```

Output:

```
{
  "EngineVersions": [
    {
      "EngineVersion": "6.2",
      "EnginePatchVersion": "6.2.6",
      "ParameterGroupFamily": "memorydb_redis6"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Versi mesin dan peningkatan di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DescribeEngineVersions](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Untuk mengembalikan daftar acara

Berikut ini menjelaskan peristiwa` mengembalikan daftar peristiwa.

```
aws memorydb describe-events
```

Output:

```
{
  "Events": [
    {
      "SourceName": "my-cluster",
      "SourceType": "cluster",
      "Message": "Increase replica count started for replication group my-cluster on 2022-07-22T14:09:01.440Z",
      "Date": "2022-07-22T07:09:01.443000-07:00"
    },
    {
      "SourceName": "my-user",
      "SourceType": "user",
      "Message": "Create user my-user operation completed.",
      "Date": "2022-07-22T07:00:02.975000-07:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memantau peristiwa](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## describe-parameter-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-parameter-groups`.

### AWS CLI

Untuk mengembalikan daftar grup parameter

Berikut `describe-parameter-groups` `mengembalikan daftar kelompok parameter.

```
aws memorydb describe-parameter-groups
```

Output:

```
{
  "ParameterGroups": [
    {
      "Name": "default.memorydb-redis6",
      "Family": "memorydb_redis6",
      "Description": "Default parameter group for memorydb_redis6",
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/default.memorydb-redis6"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi parameter engine menggunakan grup parameter di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DescribeParameterGroups](#) di Referensi AWS CLI Perintah.

## describe-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-parameters`.

### AWS CLI

Untuk mengembalikan daftar parameter

Berikut ini menjelaskan parameter `mengembalikan daftar parameter.

```
aws memorydb describe-parameters
```



## Output:

```
{
  "Parameters": [
    {
      "Name": "acllog-max-len",
      "Value": "128",
      "Description": "The maximum length of the ACL Log",
      "DataType": "integer",
      "AllowedValues": "1-10000",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "activedefrag",
      "Value": "no",
      "Description": "Enabled active memory defragmentation",
      "DataType": "string",
      "AllowedValues": "yes,no",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-cycle-max",
      "Value": "75",
      "Description": "Maximal effort for defrag in CPU percentage",
      "DataType": "integer",
      "AllowedValues": "1-75",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-cycle-min",
      "Value": "5",
      "Description": "Minimal effort for defrag in CPU percentage",
      "DataType": "integer",
      "AllowedValues": "1-75",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-ignore-bytes",
      "Value": "104857600",
      "Description": "Minimum amount of fragmentation waste to start active
defrag",
      "DataType": "integer",
      "AllowedValues": "1048576-",
      "MinimumEngineVersion": "6.2.4"
    }
  ]
}
```

```
    },
    {
      "Name": "active-defrag-max-scan-fields",
      "Value": "1000",
      "Description": "Maximum number of set/hash/zset/list fields that will be
processed from the main dictionary scan",
      "DataType": "integer",
      "AllowedValues": "1-1000000",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-threshold-lower",
      "Value": "10",
      "Description": "Minimum percentage of fragmentation to start active
defrag",
      "DataType": "integer",
      "AllowedValues": "1-100",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-threshold-upper",
      "Value": "100",
      "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
      "DataType": "integer",
      "AllowedValues": "1-100",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-expire-effort",
      "Value": "1",
      "Description": "The amount of effort that redis uses to expire items in
the active expiration job",
      "DataType": "integer",
      "AllowedValues": "1-10",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "activeresharding",
      "Value": "yes",
      "Description": "Apply rehashing or not",
      "DataType": "string",
      "AllowedValues": "yes,no",
      "MinimumEngineVersion": "6.2.4"
    }
  ],
  {
    "Name": "activeresharding",
    "Value": "yes",
    "Description": "Apply rehashing or not",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  }
],
{
  "Name": "activeresharding",
  "Value": "yes",
  "Description": "Apply rehashing or not",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "MinimumEngineVersion": "6.2.4"
}
```

```
  },
  {
    "Name": "client-output-buffer-limit-normal-hard-limit",
    "Value": "0",
    "Description": "Normal client output buffer hard limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-normal-soft-limit",
    "Value": "0",
    "Description": "Normal client output buffer soft limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-normal-soft-seconds",
    "Value": "0",
    "Description": "Normal client output buffer soft limit in seconds",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-pubsub-hard-limit",
    "Value": "33554432",
    "Description": "Pubsub client output buffer hard limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-pubsub-soft-limit",
    "Value": "8388608",
    "Description": "Pubsub client output buffer soft limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-pubsub-soft-seconds",
    "Value": "60",
```

```
    "Description": "Pubsub client output buffer soft limit in seconds",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "hash-max-ziplist-entries",
    "Value": "512",
    "Description": "The maximum number of hash entries in order for the
dataset to be compressed",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "hash-max-ziplist-value",
    "Value": "64",
    "Description": "The threshold of biggest hash entries in order for the
dataset to be compressed",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "hll-sparse-max-bytes",
    "Value": "3000",
    "Description": "HyperLogLog sparse representation bytes limit",
    "DataType": "integer",
    "AllowedValues": "1-16000",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lazyfree-lazy-eviction",
    "Value": "no",
    "Description": "Perform an asynchronous delete on evictions",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lazyfree-lazy-expire",
    "Value": "no",
    "Description": "Perform an asynchronous delete on expired keys",
    "DataType": "string",
```

```
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lazyfree-lazy-server-del",
    "Value": "no",
    "Description": "Perform an asynchronous delete on key updates",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lazyfree-lazy-user-del",
    "Value": "no",
    "Description": "Specifies whether the default behavior of DEL command
acts the same as UNLINK",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lfu-decay-time",
    "Value": "1",
    "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lfu-log-factor",
    "Value": "10",
    "Description": "The log factor for incrementing key counter for LFU
eviction policy",
    "DataType": "integer",
    "AllowedValues": "1-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "list-compress-depth",
    "Value": "0",
    "Description": "Number of quicklist ziplist nodes from each side of
the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
```

```

        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "maxmemory-policy",
        "Value": "noeviction",
        "Description": "Max memory policy",
        "DataType": "string",
        "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "maxmemory-samples",
        "Value": "3",
        "Description": "Max memory samples",
        "DataType": "integer",
        "AllowedValues": "1-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "notify-keyspace-events",
        "Description": "The keyspace events for Redis to notify Pub/Sub clients
about. By default all notifications are disabled",
        "DataType": "string",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "set-max-intset-entries",
        "Value": "512",
        "Description": "The limit in the size of the set in order for the
dataset to be compressed",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "slowlog-log-slower-than",
        "Value": "10000",
        "Description": "The execution time, in microseconds, to exceed in order
for the command to get logged. Note that a negative number disables the slow log,
while a value of zero forces the logging of every command",
        "DataType": "integer",

```

```
    "AllowedValues": "-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "slowlog-max-len",
    "Value": "128",
    "Description": "The length of the slow log. There is no limit to this
length. Just be aware that it will consume memory. You can reclaim memory used by
the slow log with SLOWLOG RESET.",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "stream-node-max-bytes",
    "Value": "4096",
    "Description": "The maximum size of a single node in a stream in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "stream-node-max-entries",
    "Value": "100",
    "Description": "The maximum number of items a single node in a stream
can contain",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "tcp-keepalive",
    "Value": "300",
    "Description": "If non-zero, send ACKs every given number of seconds",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "timeout",
    "Value": "0",
    "Description": "Close connection if client is idle for a given number of
seconds, or never if 0",
    "DataType": "integer",
```

```

        "AllowedValues": "0,20-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "tracking-table-max-keys",
        "Value": "1000000",
        "Description": "The maximum number of keys allowed for the tracking
table for client side caching",
        "DataType": "integer",
        "AllowedValues": "1-1000000000",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "zset-max-ziplist-entries",
        "Value": "128",
        "Description": "The maximum number of sorted set entries in order for
the dataset to be compressed",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "zset-max-ziplist-value",
        "Value": "64",
        "Description": "The threshold of biggest sorted set entries in order for
the dataset to be compressed",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Mengonfigurasi parameter engine menggunakan grup parameter di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DescribeParameters](#) di Referensi AWS CLI Perintah.

## describe-snapshots

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshots`.



## AWS CLI

Untuk mengembalikan daftar snapshot

Berikut ini `describe-snapshots`` mengembalikan daftar snapshot.

```
aws memorydb describe-snapshots
```

Output:

```
{
  "Snapshots": [
    {
      "Name": "my-cluster-snapshot",
      "Status": "available",
      "Source": "manual",
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx2:snapshot/my-cluster-snapshot",
      "ClusterConfiguration": {
        "Name": "my-cluster",
        "Description": " ",
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "MaintenanceWindow": "wed:03:00-wed:04:00",
        "Port": 6379,
        "ParameterGroupName": "default.memorydb-redis6",
        "SubnetGroupName": "my-sg",
        "VpcId": "vpc-862574fc",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "04:30-05:30",
        "NumShards": 2
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Snapshot dan restore di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DescribeSnapshots](#) di Referensi AWS CLI Perintah.

## describe-subnet-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-subnet-groups`.

## AWS CLI

Untuk mengembalikan daftar grup subnet

Berikut `describe-subnet-groups` mengembalikan daftar kelompok subnet.

```
aws memorydb describe-subnet-groups
```

### Output

```
{
  "SubnetGroups": [
    {
      "Name": "my-sg",
      "Description": "pat-sg",
      "VpcId": "vpc-86xxx4fc",
      "Subnets": [
        {
          "Identifier": "subnet-faxx84a6",
          "AvailabilityZone": {
            "Name": "us-east-1b"
          }
        },
        {
          "Identifier": "subnet-56xxf61b",
          "AvailabilityZone": {
            "Name": "us-east-1a"
          }
        }
      ],
      "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:subnetgroup/my-sg"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Subnet dan grup subnet](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [DescribeSubnetGroups](#) di Referensi AWS CLI Perintah.

## describe-users

Contoh kode berikut menunjukkan cara menggunakan `describe-users`.

## AWS CLI

Untuk mengembalikan daftar pengguna

Berikut ini `describe-users` mengembalikan daftar pengguna.

```
aws memorydb describe-users
```

### Output

```
{
  "Users": [
    {
      "Name": "default",
      "Status": "active",
      "AccessString": "on ~* &* +@all",
      "ACLNames": [
        "open-access"
      ],
      "MinimumEngineVersion": "6.0",
      "Authentication": {
        "Type": "no-password"
      },
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/default"
    },
    {
      "Name": "my-user",
      "Status": "active",
      "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
      "ACLNames": [],
      "MinimumEngineVersion": "6.2",
      "Authentication": {
        "Type": "password",
        "PasswordCount": 2
      },
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [DescribeUsers](#) di Referensi AWS CLI Perintah.

## failover-shard

Contoh kode berikut menunjukkan cara menggunakan `failover-shard`.

### AWS CLI

Untuk gagal karena pecahan

Failover-shard` berikut gagal di atas pecahan.

```
aws memorydb failover-shard \  
  --cluster-name my-cluster --shard-name 0001
```

Output:

```
{  
  "Cluster": {  
    "Name": "my-cluster",  
    "Status": "available",  
    "NumberOfShards": 2,  
    "ClusterEndpoint": {  
      "Address": "clustercfg.my-cluster.xxxxxx.memorydb.us-  
east-1.amazonaws.com",  
      "Port": 6379  
    },  
    "NodeType": "db.r6g.large",  
    "EngineVersion": "6.2",  
    "EnginePatchVersion": "6.2.6",  
    "ParameterGroupName": "default.memorydb-redis6",  
    "ParameterGroupStatus": "in-sync",  
    "SecurityGroups": [  
      {  
        "SecurityGroupId": "sg-0a143xxxx45c9fae",  
        "Status": "active"  
      }  
    ],  
    "SubnetGroupName": "my-sg",  
    "TLSEnabled": true,  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",  
    "SnapshotRetentionLimit": 0,  
  }  
}
```

```
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "AutoMinorVersionUpgrade": true
  }
}
```

Untuk informasi selengkapnya, lihat [Meminimalkan waktu henti dengan MultiAZ](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [FailoverShard](#) di Referensi AWS CLI Perintah.

## list-allowed-node-type-updates

Contoh kode berikut menunjukkan cara menggunakan `list-allowed-node-type-updates`.

### AWS CLI

Untuk mengembalikan daftar pembaruan tipe node yang diizinkan

Berikut `list-allowed-node-type -updates` mengembalikan daftar update tipe node yang tersedia.

```
aws memorydb list-allowed-node-type-updates
```

### Output:

```
{
  "Cluster": {
    "Name": "my-cluster",
    "Status": "available",
    "NumberOfShards": 2,
    "ClusterEndpoint": {
      "Address": "clustercfg.my-cluster.xxxxxx.memorydb.us-
east-1.amazonaws.com",
      "Port": 6379
    },
    "NodeType": "db.r6g.large",
    "EngineVersion": "6.2",
    "EnginePatchVersion": "6.2.6",
    "ParameterGroupName": "default.memorydb-redis6",
    "ParameterGroupStatus": "in-sync",
    "SecurityGroups": [
      {
        "SecurityGroupId": "sg-0a143xxxx45c9fae",
```

```
        "Status": "active"
      }
    ],
    "SubnetGroupName": "my-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "AutoMinorVersionUpgrade": true
  }
}
```

Untuk informasi selengkapnya, lihat [Penskalaan](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [ListAllowedNodeTypeUpdates](#) di Referensi AWS CLI Perintah.

## list-tags

Contoh kode berikut menunjukkan cara menggunakan `list-tags`.

### AWS CLI

Untuk mengembalikan daftar tag

Berikut daftar-tag mengembalikan daftar tag.

```
aws memorydb list-tags \
  --resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster
```

Output:

```
{
  "TagList": [
    {
      "Key": "mytag",
      "Value": "myvalue"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [ListTags](#) di Referensi AWS CLI Perintah.

## reset-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `reset-parameter-group`.

### AWS CLI

Untuk mengatur ulang grup parameter

Berikut `reset-parameter-group` `me-reset grup parameter.

```
aws memorydb reset-parameter-group \  
  --parameter-group-name my-parameter-group \  
  --all-parameters
```

Output:

```
{  
  "ParameterGroup": {  
    "Name": "my-parameter-group",  
    "Family": "memorydb_redis6",  
    "Description": "my parameter group",  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/my-parameter-  
group"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi parameter engine menggunakan grup parameter di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [ResetParameterGroup](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

Berikut `tag-resource` `menambahkan tag ke sumber daya.

```
aws memorydb tag-resource \  
  --resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster \  
  --tags Key="mykey",Value="myvalue"
```

Output:

```
{  
  "TagList": [  
    {  
      "Key": "mytag",  
      "Value": "myvalue"  
    },  
    {  
      "Key": "mykey",  
      "Value": "myvalue"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk memperbarui ACL

Berikut `update-acl` memperbarui ACL dengan menambahkan pengguna.

```
aws memorydb untag-resource \  
  --resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster \  
  --tag-keys mykey
```

Output:

```
{  
  "TagList": [  
    ]  
}
```



```
{
  "Key": "mytag",
  "Value": "myvalue"
}
]
```

Untuk informasi selengkapnya, lihat [Menandai sumber daya](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-cluster

Contoh kode berikut menunjukkan cara menggunakan `update-cluster`.

### AWS CLI

Untuk memperbarui klaster

Berikut `update-cluster``` memperbarui kelompok parameter cluster ke. `my-parameter-group`

```
aws memorydb update-cluster \
  --cluster-name my-cluster \
  --parameter-group-name my-parameter-group
```

Output:

```
{
  "Cluster": {
    "Name": "my-cluster",
    "Status": "available",
    "NumberOfShards": 2,
    "AvailabilityMode": "MultiAZ",
    "ClusterEndpoint": {
      "Address": "clustercfg.my-cluster.llru6f.memorydb.us-
east-1.amazonaws.com",
      "Port": 6379
    },
    "NodeType": "db.r6g.large",
    "EngineVersion": "6.2",
    "EnginePatchVersion": "6.2.6",
    "ParameterGroupName": "my-parameter-group",
```

```

    "ParameterGroupStatus": "in-sync",
    "SecurityGroups": [
      {
        "SecurityGroupId": "sg-0a143xxxxxc9fae",
        "Status": "active"
      }
    ],
    "SubnetGroupName": "pat-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "ACLName": "my-acl",
    "AutoMinorVersionUpgrade": true
  }
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi klaster](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [UpdateCluster](#) di Referensi AWS CLI Perintah.

## update-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `update-parameter-group`.

### AWS CLI

Untuk memperbarui grup parameter

Berikut `update-parameter-group` `` memperbarui grup parameter.

```

aws memorydb update-parameter-group \
  --parameter-group-name my-parameter-group \
  --parameter-name-values "ParameterName=activedefrag, ParameterValue=no"

```

Output:

```

{
  "ParameterGroup": {
    "Name": "my-parameter-group",
    "Family": "memorydb_redis6",

```

```
"Description": "my parameter group",
"ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:parametergroup/my-parameter-
group"
}
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi grup parameter di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [UpdateParameterGroup](#) di Referensi AWS CLI Perintah.

## update-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `update-subnet-group`.

### AWS CLI

Untuk memperbarui grup subnet

Berikut ini `update-subnet-group` `memperbarui subnet ID grup subnet.

```
aws memorydb update-subnet-group \
  --subnet-group-name my-sg \
  --subnet-ids subnet-01f29d458f3xxxxxx
```

Output:

```
{
  "SubnetGroup": {
    "Name": "my-sg-1",
    "Description": "my-sg",
    "VpcId": "vpc-09d2cfc01xxxxxxx",
    "Subnets": [
      {
        "Identifier": "subnet-01f29d458fxxxxxx",
        "AvailabilityZone": {
          "Name": "us-east-1a"
        }
      }
    ],
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/my-sg"
  }
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Subnet dan grup subnet](#) di Panduan Pengguna MemoryDB.

- Untuk detail API, lihat [UpdateSubnetGroup](#) di Referensi AWS CLI Perintah.

## update-user

Contoh kode berikut menunjukkan cara menggunakan `update-user`.

### AWS CLI

Untuk memperbarui pengguna

Berikut ini `update-user` memodifikasi string akses pengguna.

```
aws memorydb update-user \  
  --user-name my-user \  
  --access-string "off ~objects:* ~items:* ~public:* resetchannels -@all"
```

Output:

```
{  
  "User": {  
    "Name": "my-user",  
    "Status": "modifying",  
    "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",  
    "ACLNames": [  
      "myt-acl"  
    ],  
    "MinimumEngineVersion": "6.2",  
    "Authentication": {  
      "Type": "password",  
      "PasswordCount": 2  
    },  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengautentikasi pengguna dengan Daftar Kontrol Akses di Panduan Pengguna MemoryDB](#).

- Untuk detail API, lihat [UpdateUser](#) di Referensi AWS CLI Perintah.

## Amazon MSK contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan MSK AWS Command Line Interface with Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-cluster**

Contoh kode berikut menunjukkan cara menggunakan `create-cluster`.

AWS CLI

Untuk membuat cluster MSK Amazon

`create-cluster` Contoh berikut membuat cluster MSK bernama `MessagingCluster` dengan tiga node broker. File JSON bernama `brokernodegroupinfo.json` menentukan tiga subnet di mana Anda ingin Amazon MSK untuk mendistribusikan node broker. Contoh ini tidak menentukan tingkat pemantauan, sehingga cluster mendapatkan `DEFAULT` level.

```
aws kafka create-cluster \  
  --cluster-name "MessagingCluster" \  
  --broker-node-group-info file://brokernodegroupinfo.json \  
  --kafka-version "2.2.1" \  
  --number-of-broker-nodes 3
```

Isi dari `brokernodegroupinfo.json`:

```
{
  "InstanceType": "kafka.m5.xlarge",
  "BrokerAZDistribution": "DEFAULT",
  "ClientSubnets": [
    "subnet-0123456789111abcd",
    "subnet-0123456789222abcd",
    "subnet-0123456789333abcd"
  ]
}
```

Output:

```
{
  "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
  "ClusterName": "MessagingCluster",
  "State": "CREATING"
}
```

Untuk informasi selengkapnya, lihat [Membuat Cluster MSK Amazon](#) di Amazon Managed Streaming for Apache Kafka.

- Untuk detail API, lihat [CreateCluster](#) di Referensi AWS CLI Perintah.

## **create-configuration**

Contoh kode berikut menunjukkan cara menggunakan `create-configuration`.

### AWS CLI

Untuk membuat konfigurasi MSK Amazon kustom

`create-configuration` Contoh berikut membuat konfigurasi MSK kustom dengan properti server yang ditentukan dalam file input.

```
aws kafka create-configuration \
  --name "CustomConfiguration" \
  --description "Topic autocreation enabled; Apache ZooKeeper timeout 2000 ms; Log
rolling 604800000 ms." \
  --kafka-versions "2.2.1" \
```

```
--server-properties file://configuration.txt
```

Isi dari `configuration.txt`:

```
auto.create.topics.enable = true
zookeeper.connection.timeout.ms = 2000
log.roll.ms = 604800000
```

Perintah ini tidak menghasilkan output. Output:

```
{
  "Arn": "arn:aws:kafka:us-west-2:123456789012:configuration/CustomConfiguration/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
  "CreationTime": "2019-10-09T15:26:05.548Z",
  "LatestRevision":
    {
      "CreationTime": "2019-10-09T15:26:05.548Z",
      "Description": "Topic autocreation enabled; Apache ZooKeeper timeout
2000 ms; Log rolling 604800000 ms.",
      "Revision": 1
    },
  "Name": "CustomConfiguration"
}
```

Untuk informasi selengkapnya, lihat [Operasi Konfigurasi MSK Amazon di Panduan](#) Pengembang Amazon Managed Streaming for Apache Kafka.

- Untuk detail API, lihat [CreateConfiguration](#) di Referensi AWS CLI Perintah.

## describe-cluster

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster`.

### AWS CLI

Untuk menggambarkan sebuah cluster

`describe-cluster` Contoh berikut menjelaskan cluster MSK Amazon.

```
aws kafka describe-cluster \
  --cluster-arn arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5
```

## Output:

```
{
  "ClusterInfo": {
    "BrokerNodeGroupInfo": {
      "BrokerAZDistribution": "DEFAULT",
      "ClientSubnets": [
        "subnet-cbfff283",
        "subnet-6746046b"
      ],
      "InstanceType": "kafka.m5.large",
      "SecurityGroups": [
        "sg-f839b688"
      ],
      "StorageInfo": {
        "EbsStorageInfo": {
          "VolumeSize": 100
        }
      }
    },
    "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5",
    "ClusterName": "demo-cluster-1",
    "CreationTime": "2020-07-09T02:31:36.223000+00:00",
    "CurrentBrokerSoftwareInfo": {
      "KafkaVersion": "2.2.1"
    },
    "CurrentVersion": "K3AEGXETSR30VB",
    "EncryptionInfo": {
      "EncryptionAtRest": {
        "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/a7ca56d5-0768-4b64-a670-339a9fbef81c"
      },
      "EncryptionInTransit": {
        "ClientBroker": "TLS_PLAINTEXT",
        "InCluster": true
      }
    },
    "EnhancedMonitoring": "DEFAULT",
    "OpenMonitoring": {
      "Prometheus": {
        "JmxExporter": {
          "EnabledInBroker": false
        }
      }
    }
  }
}
```



```

        "NodeExporter": {
            "EnabledInBroker": false
        }
    },
    "NumberOfBrokerNodes": 2,
    "State": "ACTIVE",
    "Tags": {},
    "ZookeeperConnectString": "z-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-3.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181"
    }
}

```

Untuk informasi selengkapnya, lihat [Mendaftarkan Cluster MSK Amazon di Panduan Pengembang Amazon Managed Streaming for Apache Kafka](#).

- Untuk detail API, lihat [DescribeCluster](#) di Referensi AWS CLI Perintah.

## get-bootstrap-brokers

Contoh kode berikut menunjukkan cara menggunakan `get-bootstrap-brokers`.

### AWS CLI

Untuk mendapatkan broker bootstrap

`get-bootstrap-brokers` Contoh berikut mengambil informasi broker bootstrap untuk cluster MSK Amazon.

```

aws kafka get-bootstrap-brokers \
  --cluster-arn arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5

```

Output:

```

{
  "BootstrapBrokerString": "b-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:9092,b-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:9092",

```

```
"BootstrapBrokerStringTls": "b-1.demo-cluster-1.xuy0sb.c5.kafka.us-east-1.amazonaws.com:9094,b-2.demo-cluster-1.xuy0sb.c5.kafka.us-east-1.amazonaws.com:9094"
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan Broker Bootstrap](#) di Amazon Managed Streaming for Apache Kafka Developer Guide.

- Untuk detail API, lihat [GetBootstrapBrokers](#) di Referensi AWS CLI Perintah.

## list-clusters

Contoh kode berikut menunjukkan cara menggunakan `list-clusters`.

### AWS CLI

Untuk daftar cluster yang tersedia

`list-clusters` Contoh berikut mencantumkan kluster MSK Amazon di akun Anda AWS .

```
aws kafka list-clusters
```

Output:

```
{
  "ClusterInfoList": [
    {
      "BrokerNodeGroupInfo": {
        "BrokerAZDistribution": "DEFAULT",
        "ClientSubnets": [
          "subnet-cbfff283",
          "subnet-6746046b"
        ],
        "InstanceType": "kafka.m5.large",
        "SecurityGroups": [
          "sg-f839b688"
        ],
        "StorageInfo": {
          "EbsStorageInfo": {
            "VolumeSize": 100
          }
        }
      },
    },
  ],
}
```

```

    "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5",
    "ClusterName": "demo-cluster-1",
    "CreationTime": "2020-07-09T02:31:36.223000+00:00",
    "CurrentBrokerSoftwareInfo": {
      "KafkaVersion": "2.2.1"
    },
    "CurrentVersion": "K3AEGXETSR30VB",
    "EncryptionInfo": {
      "EncryptionAtRest": {
        "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/
a7ca56d5-0768-4b64-a670-339a9fbef81c"
      },
      "EncryptionInTransit": {
        "ClientBroker": "TLS_PLAINTEXT",
        "InCluster": true
      }
    },
    "EnhancedMonitoring": "DEFAULT",
    "OpenMonitoring": {
      "Prometheus": {
        "JmxExporter": {
          "EnabledInBroker": false
        },
        "NodeExporter": {
          "EnabledInBroker": false
        }
      }
    },
    "NumberOfBrokerNodes": 2,
    "State": "ACTIVE",
    "Tags": {},
    "ZookeeperConnectString": "z-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-3.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mendaftarkan Cluster MSK Amazon di Panduan Pengembang Amazon Managed Streaming for Apache Kafka](#).

- Untuk detail API, lihat [ListClusters](#) di Referensi AWS CLI Perintah.

## update-broker-storage

Contoh kode berikut menunjukkan cara menggunakan `update-broker-storage`.

### AWS CLI

Untuk memperbarui penyimpanan EBS untuk broker

`update-broker-storage` Contoh berikut memperbarui jumlah penyimpanan EBS untuk semua broker di cluster. Amazon MSK menetapkan jumlah penyimpanan target untuk setiap broker ke jumlah yang ditentukan dalam contoh. Anda bisa mendapatkan versi cluster saat ini dengan menjelaskan cluster atau dengan mencantumkan semua cluster.

```
aws kafka update-broker-storage \  
  --cluster-arn "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2" \  
  --current-version "K21V3IB1VIZYYH" \  
  --target-broker-efs-volume-info "KafkaBrokerNodeId=ALL,VolumeSizeGB=1100"
```

Output mengembalikan ARN untuk operasi ini `update-broker-storage`. Untuk menentukan apakah operasi ini selesai, gunakan `describe-cluster-operation` perintah dengan ARN ini sebagai input.

```
{  
  "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",  
  "ClusterOperationArn": "arn:aws:kafka:us-west-2:123456789012:cluster-  
operation/V123450123/a1b2c3d4-1234-abcd-cdef-22222EXAMPLE-2/a1b2c3d4-abcd-1234-  
bcde-33333EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Penyimpanan EBS untuk Broker di Panduan Pengembang Amazon Managed Streaming for Apache Kafka](#).

- Untuk detail API, lihat [UpdateBrokerStorage](#) di Referensi AWS CLI Perintah.

## update-cluster-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-cluster-configuration`.

## AWS CLI

Untuk memperbarui konfigurasi cluster MSK Amazon

`update-cluster-configuration` Contoh berikut memperbarui konfigurasi cluster MSK yang ada yang ditentukan. Ini menggunakan konfigurasi MSK khusus.

```
aws kafka update-cluster-configuration \  
  --cluster-arn "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2" \  
  --configuration-info file://configuration-info.json \  
  --current-version "K21V3IB1VIZYYH"
```

Isi dari `configuration-info.json`:

```
{  
  "Arn": "arn:aws:kafka:us-west-2:123456789012:configuration/CustomConfiguration/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",  
  "Revision": 1  
}
```

Output mengembalikan ARN untuk operasi ini `update-cluster-configuration`. Untuk menentukan apakah operasi ini selesai, gunakan `describe-cluster-operation` perintah dengan ARN ini sebagai input.

```
{  
  "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",  
  "ClusterOperationArn": "arn:aws:kafka:us-west-2:123456789012:cluster-  
operation/V123450123/a1b2c3d4-1234-abcd-cdef-22222EXAMPLE-2/a1b2c3d4-abcd-1234-  
bcde-33333EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Konfigurasi Cluster MSK Amazon di Panduan Pengembang Amazon Managed Streaming for Apache Kafka](#).

- Untuk detail API, lihat [UpdateClusterConfiguration](#) di Referensi AWS CLI Perintah.

## Contoh Network Manager menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Network Manager.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-customer-gateway**

Contoh kode berikut menunjukkan cara menggunakan `associate-customer-gateway`.

AWS CLI

Untuk mengaitkan gateway pelanggan

`associate-customer-gateway` Contoh berikut mengaitkan gateway pelanggan `cgw-11223344556677889` di jaringan global yang ditentukan dengan perangkat `device-07f6fd08867abc123`.

```
aws networkmanager associate-customer-gateway \
  --customer-gateway-arn arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889 \
  --global-network-id global-network-01231231231231231 \
  --device-id device-07f6fd08867abc123 \
  --region us-west-2
```

Output:

```
{
  "CustomerGatewayAssociation": {
    "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-gateway/cgw-11223344556677889",
    "GlobalNetworkId": "global-network-01231231231231231",
    "DeviceId": "device-07f6fd08867abc123",
    "State": "PENDING"
  }
}
```

Untuk informasi selengkapnya, lihat [Asosiasi Gateway Pelanggan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [AssociateCustomerGateway](#) di Referensi AWS CLI Perintah.

## associate-link

Contoh kode berikut menunjukkan cara menggunakan `associate-link`.

### AWS CLI

Untuk mengaitkan tautan

`associate-link` Contoh berikut mengaitkan tautan `link-11112222aaaabbbb1` dengan perangkat `device-07f6fd08867abc123`. Tautan dan perangkat berada di jaringan global yang ditentukan.

```
aws networkmanager associate-link \
  --global-network-id global-network-01231231231231231 \
  --device-id device-07f6fd08867abc123 \
  --link-id link-11112222aaaabbbb1 \
  --region us-west-2
```

Output:

```
{
  "LinkAssociation": {
    "GlobalNetworkId": "global-network-01231231231231231",
    "DeviceId": "device-07f6fd08867abc123",
    "LinkId": "link-11112222aaaabbbb1",
    "LinkAssociationState": "PENDING"
  }
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Asosiasi Perangkat dan Tautan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [AssociateLink](#) di Referensi AWS CLI Perintah.

## create-core-network

Contoh kode berikut menunjukkan cara menggunakan `create-core-network`.

### AWS CLI

Untuk membuat jaringan inti

`create-core-network` Contoh berikut membuat jaringan inti menggunakan deskripsi opsional dan tag dalam jaringan global AWS Cloud WAN.

```
aws networkmanager create-core-network \  
  --global-network-id global-network-0d59060f16a73bc41\  
  --description "Main headquarters location"\  
  --tags Key=Name,Value="New York City office"
```

Output:

```
{  
  "CoreNetwork": {  
    "GlobalNetworkId": "global-network-0d59060f16a73bc41",  
    "CoreNetworkId": "core-network-0fab62fe438d94db6",  
    "CoreNetworkArn": "arn:aws:networkmanager::987654321012:core-network/core-  
network-0fab62fe438d94db6",  
    "Description": "Main headquarters location",  
    "CreatedAt": "2022-01-10T19:53:59+00:00",  
    "State": "AVAILABLE",  
    "Tags": [  
      {  
        "Key": "Name",  
        "Value": "New York City office"  
      }  
    ]  
  }  
}
```



```
}
```

Untuk informasi selengkapnya, lihat [Jaringan inti](#) di Panduan Pengguna AWS Cloud WAN.

- Untuk detail API, lihat [CreateCoreNetwork](#) di Referensi AWS CLI Perintah.

## create-device

Contoh kode berikut menunjukkan cara menggunakan `create-device`.

### AWS CLI

Untuk membuat perangkat

`create-device` Contoh berikut membuat perangkat dalam jaringan global yang ditentukan. Detail perangkat mencakup deskripsi, jenis, vendor, model, dan nomor seri.

```
aws networkmanager create-device
  --global-network-id global-network-01231231231231231 \
  --description "New York office device" \
  --type "office device" \
  --vendor "anycompany" \
  --model "abcabc" \
  --serial-number "1234" \
  --region us-west-2
```

Output:

```
{
  "Device": {
    "DeviceId": "device-07f6fd08867abc123",
    "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123",
    "GlobalNetworkId": "global-network-01231231231231231",
    "Description": "New York office device",
    "Type": "office device",
    "Vendor": "anycompany",
    "Model": "abcabc",
    "SerialNumber": "1234",
    "CreatedAt": 1575554005.0,
    "State": "PENDING"
  }
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Perangkat](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [CreateDevice](#) di Referensi AWS CLI Perintah.

## **create-global-network**

Contoh kode berikut menunjukkan cara menggunakan `create-global-network`.

### AWS CLI

Untuk membuat jaringan global

`create-global-network` Contoh berikut menciptakan jaringan global baru. Keadaan awal pada penciptaan adalah `PENDING`.

```
aws networkmanager create-global-network
```

Output:

```
{
  "GlobalNetwork": {
    "GlobalNetworkId": "global-network-00a77fc0f722dae74",
    "GlobalNetworkArn": "arn:aws:networkmanager::987654321012:global-network/global-network-00a77fc0f722dae74",
    "CreatedAt": "2022-03-14T20:31:56+00:00",
    "State": "PENDING"
  }
}
```

- Untuk detail API, lihat [CreateGlobalNetwork](#) di Referensi AWS CLI Perintah.

## **create-link**

Contoh kode berikut menunjukkan cara menggunakan `create-link`.

### AWS CLI

Untuk membuat tautan

`create-link` Contoh berikut membuat link dalam jaringan global yang ditentukan. Tautan tersebut mencakup deskripsi dan detail tentang jenis tautan, bandwidth, dan penyedia. ID situs menunjukkan situs yang terkait dengan tautan tersebut.

```
aws networkmanager create-link \  
  --global-network-id global-network-01231231231231231 \  
  --description "VPN Link" \  
  --type "broadband" \  
  --bandwidth UploadSpeed=10,DownloadSpeed=20 \  
  --provider "AnyCompany" \  
  --site-id site-444555aaaabbb11223 \  
  --region us-west-2
```

Output:

```
{  
  "Link": {  
    "LinkId": "link-11112222aaaabbbb1",  
    "LinkArn": "arn:aws:networkmanager::123456789012:link/global-  
network-01231231231231231/link-11112222aaaabbbb1",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "SiteId": "site-444555aaaabbb11223",  
    "Description": "VPN Link",  
    "Type": "broadband",  
    "Bandwidth": {  
      "UploadSpeed": 10,  
      "DownloadSpeed": 20  
    },  
    "Provider": "AnyCompany",  
    "CreatedAt": 1575555811.0,  
    "State": "PENDING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tautan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [CreateLink](#) di Referensi AWS CLI Perintah.

## **create-site**

Contoh kode berikut menunjukkan cara menggunakan `create-site`.

## AWS CLI

Untuk membuat situs

`create-site` Contoh berikut membuat situs di jaringan global yang ditentukan. Detail situs mencakup deskripsi dan informasi lokasi.

```
aws networkmanager create-site \  
  --global-network-id global-network-01231231231231231 \  
  --description "New York head office" \  
  --location Latitude=40.7128,Longitude=-74.0060 \  
  --region us-west-2
```

Output:

```
{  
  "Site": {  
    "SiteId": "site-444555aaabbb11223",  
    "SiteArn": "arn:aws:networkmanager::123456789012:site/global-  
network-01231231231231231/site-444555aaabbb11223",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York head office",  
    "Location": {  
      "Latitude": "40.7128",  
      "Longitude": "-74.0060"  
    },  
    "CreatedAt": 1575554300.0,  
    "State": "PENDING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Situs](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [CreateSite](#) di Referensi AWS CLI Perintah.

## `create-vpc-attachment`

Contoh kode berikut menunjukkan cara menggunakan `create-vpc-attachment`.

## AWS CLI

Untuk membuat lampiran VPC

`create-vpc-attachment` Contoh berikut membuat lampiran VPC dengan dukungan IPv6 dalam jaringan inti.

```
aws networkmanager create-vpc-attachment \  
  --core-network-id core-network-0fab62fe438d94db6 \  
  --vpc-arn arn:aws:ec2:us-east-1:987654321012:vpc/vpc-09f37f69e2786eeb8 \  
  --subnet-arns arn:aws:ec2:us-east-1:987654321012:subnet/subnet-04ca4e010857e7bb7 \  
  --Ipv6Support=true
```

Output:

```
{  
  "VpcAttachment": {  
    "Attachment": {  
      "CoreNetworkId": "core-network-0fab62fe438d94db6",  
      "AttachmentId": "attachment-05e1da6eba87a06e6",  
      "OwnerAccountId": "987654321012",  
      "AttachmentType": "VPC",  
      "State": "CREATING",  
      "EdgeLocation": "us-east-1",  
      "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/  
vpc-09f37f69e2786eeb8",  
      "Tags": [],  
      "CreatedAt": "2022-03-10T20:59:14+00:00",  
      "UpdatedAt": "2022-03-10T20:59:14+00:00"  
    },  
    "SubnetArns": [  
      "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-04ca4e010857e7bb7"  
    ],  
    "Options": {  
      "Ipv6Support": true  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat lampiran](#) di Panduan Pengguna Cloud WAN.

- Untuk detail API, lihat [CreateVpcAttachment](#) di Referensi AWS CLI Perintah.

## delete-attachment

Contoh kode berikut menunjukkan cara menggunakan `delete-attachment`.

### AWS CLI

Untuk menghapus lampiran

`delete-attachment` Contoh berikut menghapus lampiran Connect.

```
aws networkmanager delete-attachment \  
  --attachment-id attachment-01feddaeeae26ab68c
```

Output:

```
{  
  "Attachment": {  
    "CoreNetworkId": "core-network-0f4b0a9d5ee7761d1",  
    "AttachmentId": "attachment-01feddaeeae26ab68c",  
    "OwnerAccountId": "987654321012",  
    "AttachmentType": "CONNECT",  
    "State": "DELETING",  
    "EdgeLocation": "us-east-1",  
    "ResourceArn": "arn:aws:networkmanager::987654321012:attachment/  
attachment-02c3964448fedf5aa",  
    "CreatedAt": "2022-03-15T19:18:41+00:00",  
    "UpdatedAt": "2022-03-15T19:28:59+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus lampiran](#) di Panduan Pengguna Cloud WAN.

- Untuk detail API, lihat [DeleteAttachment](#) di Referensi AWS CLI Perintah.

## delete-bucket-analytics-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-analytics-configuration`.

### AWS CLI

Untuk menghapus konfigurasi analitik untuk bucket

`delete-bucket-analytics-configuration` Contoh berikut menghapus konfigurasi analitik untuk bucket dan ID yang ditentukan.

```
aws s3api delete-bucket-analytics-configuration \  
  --bucket my-bucket \  
  --id 1
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBucketAnalyticsConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-bucket-metrics-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-metrics-configuration`.

### AWS CLI

Untuk menghapus konfigurasi metrik untuk bucket

`delete-bucket-metrics-configuration` Contoh berikut menghapus konfigurasi metrik untuk bucket dan ID yang ditentukan.

```
aws s3api delete-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBucketMetricsConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-core-network**

Contoh kode berikut menunjukkan cara menggunakan `delete-core-network`.

### AWS CLI

Untuk menghapus jaringan inti

`delete-core-network` Contoh berikut menghapus jaringan inti dari jaringan global Cloud WAN.

```
aws networkmanager delete-core-network \  
--core-network-id core-network-0fab62fe438d94db6
```

### Output:

```
{  
  "CoreNetwork": {  
    "GlobalNetworkId": "global-network-0d59060f16a73bc41",  
    "CoreNetworkId": "core-network-0fab62fe438d94db6",  
    "Description": "Main headquarters location",  
    "CreatedAt": "2021-12-09T18:31:11+00:00",  
    "State": "DELETING",  
    "Segments": [  
      {  
        "Name": "dev",  
        "EdgeLocations": [  
          "us-east-1"  
        ],  
        "SharedSegments": []  
      }  
    ],  
    "Edges": [  
      {  
        "EdgeLocation": "us-east-1",  
        "Asn": 64512,  
        "InsideCidrBlocks": []  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Jaringan inti](#) di Panduan Pengguna Cloud WAN.

- Untuk detail API, lihat [DeleteCoreNetwork](#) di Referensi AWS CLI Perintah.

## delete-device

Contoh kode berikut menunjukkan cara menggunakan `delete-device`.

### AWS CLI

Untuk menghapus perangkat



`delete-device` Contoh berikut menghapus perangkat yang ditentukan dari jaringan global yang ditentukan.

```
aws networkmanager delete-device \  
  --global-network-id global-network-01231231231231231 \  
  --device-id device-07f6fd08867abc123 \  
  --region us-west-2
```

Output:

```
{  
  "Device": {  
    "DeviceId": "device-07f6fd08867abc123",  
    "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York office device",  
    "Type": "office device",  
    "Vendor": "anycompany",  
    "Model": "abcabc",  
    "SerialNumber": "1234",  
    "SiteId": "site-444555aaabbb11223",  
    "CreatedAt": 1575554005.0,  
    "State": "DELETING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Perangkat](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [DeleteDevice](#) di Referensi AWS CLI Perintah.

## **delete-global-network**

Contoh kode berikut menunjukkan cara menggunakan `delete-global-network`.

### AWS CLI

Untuk menghapus jaringan global

`delete-global-network` Contoh berikut menghapus jaringan global.

```
aws networkmanager delete-global-network \  
  --global-network-id global-network-052bedddccb193b6b
```

Output:

```
{  
  "GlobalNetwork": {  
    "GlobalNetworkId": "global-network-052bedddccb193b6b",  
    "GlobalNetworkArn": "arn:aws:networkmanager::987654321012:global-network/  
global-network-052bedddccb193b6b",  
    "CreatedAt": "2021-12-09T18:19:12+00:00",  
    "State": "DELETING"  
  }  
}
```

- Untuk detail API, lihat [DeleteGlobalNetwork](#) di Referensi AWS CLI Perintah.

## delete-link

Contoh kode berikut menunjukkan cara menggunakan `delete-link`.

### AWS CLI

Untuk menghapus tautan

`delete-link` Contoh berikut menghapus link yang ditentukan dari jaringan global yang ditentukan.

```
aws networkmanager delete-link \  
  --global-network-id global-network-01231231231231231 \  
  --link-id link-11112222aaaabbbb1 \  
  --region us-west-2
```

Output:

```
{  
  "Link": {  
    "LinkId": "link-11112222aaaabbbb1",  
    "LinkArn": "arn:aws:networkmanager::123456789012:link/global-  
network-01231231231231231/link-11112222aaaabbbb1",  
    "GlobalNetworkId": "global-network-01231231231231231",
```

```
    "SiteId": "site-444555aaabbb11223",
    "Description": "VPN Link",
    "Type": "broadband",
    "Bandwidth": {
      "UploadSpeed": 20,
      "DownloadSpeed": 20
    },
    "Provider": "AnyCompany",
    "CreatedAt": 1575555811.0,
    "State": "DELETING"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tautan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [DeleteLink](#) di Referensi AWS CLI Perintah.

## **delete-public-access-block**

Contoh kode berikut menunjukkan cara menggunakan `delete-public-access-block`.

### AWS CLI

Untuk menghapus konfigurasi blokir akses publik untuk bucket

`delete-public-access-block` Contoh berikut menghapus konfigurasi blokir akses publik pada bucket yang ditentukan.

```
aws s3api delete-public-access-block \
  --bucket my-bucket
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePublicAccessBlock](#) di Referensi AWS CLI Perintah.

## **delete-site**

Contoh kode berikut menunjukkan cara menggunakan `delete-site`.

### AWS CLI

Untuk menghapus situs

`delete-site` Contoh berikut menghapus situs tertentu (`site-444555aaabbb11223`) dalam jaringan global yang ditentukan.

```
aws networkmanager delete-site \  
  --global-network-id global-network-01231231231231231 \  
  --site-id site-444555aaabbb11223 \  
  --region us-west-2
```

Output:

```
{  
  "Site": {  
    "SiteId": "site-444555aaabbb11223",  
    "SiteArn": "arn:aws:networkmanager::123456789012:site/global-  
network-01231231231231231/site-444555aaabbb11223",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York head office",  
    "Location": {  
      "Latitude": "40.7128",  
      "Longitude": "-74.0060"  
    },  
    "CreatedAt": 1575554300.0,  
    "State": "DELETING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Situs](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [DeleteSite](#) di Referensi AWS CLI Perintah.

## **deregister-transit-gateway**

Contoh kode berikut menunjukkan cara menggunakan `deregister-transit-gateway`.

### AWS CLI

Untuk membatalkan pendaftaran gateway transit dari jaringan global

`deregister-transit-gateway` Contoh berikut membatalkan pendaftaran gateway transit yang ditentukan dari jaringan global yang ditentukan.

```
aws networkmanager deregister-transit-gateway \  
  --global-network-id global-network-01231231231231231 \  
  --transit-gateway-arn arn:aws:ec2:us-west-2:123456789012:transit-gateway/  
tgw-123abc05e04123abc \  
  --region us-west-2
```

Output:

```
{  
  "TransitGatewayRegistration": {  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway/  
tgw-123abc05e04123abc",  
    "State": {  
      "Code": "DELETING"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Pendaftaran Gateway Transit](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [DeregisterTransitGateway](#) di Referensi AWS CLI Perintah.

## describe-global-networks

Contoh kode berikut menunjukkan cara menggunakan `describe-global-networks`.

AWS CLI

Untuk menggambarkan jaringan global Anda

`describe-global-networks` Contoh berikut menjelaskan semua jaringan global Anda di akun Anda.

```
aws networkmanager describe-global-networks \  
  --region us-west-2
```

Output:

```
{
```

```

    "GlobalNetworks": [
      {
        "GlobalNetworkId": "global-network-01231231231231231",
        "GlobalNetworkArn": "arn:aws:networkmanager::123456789012:global-
network/global-network-01231231231231231",
        "Description": "Company 1 global network",
        "CreatedAt": 1575553525.0,
        "State": "AVAILABLE"
      }
    ]
  }
}

```

- Untuk detail API, lihat [DescribeGlobalNetworks](#) di Referensi AWS CLI Perintah.

## disassociate-customer-gateway

Contoh kode berikut menunjukkan cara menggunakan `disassociate-customer-gateway`.

### AWS CLI

Untuk memisahkan gateway pelanggan

`disassociate-customer-gateway` Contoh berikut memisahkan gateway pelanggan yang ditentukan (`cgw-11223344556677889`) dari jaringan global yang ditentukan.

```

aws networkmanager disassociate-customer-gateway \
  --global-network-id global-network-01231231231231231 \
  --customer-gateway-arn arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889 \
  --region us-west-2

```

### Output:

```

{
  "CustomerGatewayAssociation": {
    "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889",
    "GlobalNetworkId": "global-network-01231231231231231",
    "DeviceId": "device-07f6fd08867abc123",
    "State": "DELETING"
  }
}

```

Untuk informasi selengkapnya, lihat [Asosiasi Gateway Pelanggan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [DisassociateCustomerGateway](#) di Referensi AWS CLI Perintah.

## disassociate-link

Contoh kode berikut menunjukkan cara menggunakan `disassociate-link`.

### AWS CLI

Untuk memisahkan tautan

`disassociate-link` Contoh berikut memisahkan tautan yang ditentukan dari perangkat `device-07f6fd08867abc123` di jaringan global yang ditentukan.

```
aws networkmanager disassociate-link \
  --global-network-id global-network-01231231231231231 \
  --device-id device-07f6fd08867abc123 \
  --link-id link-11112222aaaabbbb1 \
  --region us-west-2
```

Output:

```
{
  "LinkAssociation": {
    "GlobalNetworkId": "global-network-01231231231231231",
    "DeviceId": "device-07f6fd08867abc123",
    "LinkId": "link-11112222aaaabbbb1",
    "LinkAssociationState": "DELETING"
  }
}
```

Untuk informasi selengkapnya, lihat [Asosiasi Perangkat dan Tautan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [DisassociateLink](#) di Referensi AWS CLI Perintah.

## get-bucket-analytics-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-analytics-configuration`.

## AWS CLI

Untuk mengambil konfigurasi analitik untuk bucket dengan ID tertentu

`get-bucket-analytics-configuration` Contoh berikut menampilkan konfigurasi analitik untuk bucket dan ID yang ditentukan.

```
aws s3api get-bucket-analytics-configuration \  
  --bucket my-bucket \  
  --id 1
```

Output:

```
{  
  "AnalyticsConfiguration": {  
    "StorageClassAnalysis": {},  
    "Id": "1"  
  }  
}
```

- Untuk detail API, lihat [GetBucketAnalyticsConfiguration](#) di Referensi AWS CLI Perintah.

## `get-bucket-metrics-configuration`

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-metrics-configuration`.

## AWS CLI

Untuk mengambil konfigurasi metrik untuk bucket dengan ID tertentu

`get-bucket-metrics-configuration` Contoh berikut menampilkan konfigurasi metrik untuk bucket dan ID yang ditentukan.

```
aws s3api get-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Output:

```
{  
  "MetricsConfiguration": {  
    "Filter": {
```



```
        "Prefix": "logs"
      },
      "Id": "123"
    }
  }
}
```

- Untuk detail API, lihat [GetBucketMetricsConfiguration](#) di Referensi AWS CLI Perintah.

## get-customer-gateway-associations

Contoh kode berikut menunjukkan cara menggunakan `get-customer-gateway-associations`.

### AWS CLI

Untuk mendapatkan asosiasi gateway pelanggan Anda

`get-customer-gateway-associations` Contoh berikut mendapatkan asosiasi gateway pelanggan untuk jaringan global yang ditentukan.

```
aws networkmanager get-customer-gateway-associations \
  --global-network-id global-network-01231231231231231 \
  --region us-west-2
```

### Output:

```
{
  "CustomerGatewayAssociations": [
    {
      "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-
gateway/cgw-11223344556677889",
      "GlobalNetworkId": "global-network-01231231231231231",
      "DeviceId": "device-07f6fd08867abc123",
      "State": "AVAILABLE"
    }
  ]
}
```

- Untuk detail API, lihat [GetCustomerGatewayAssociations](#) di Referensi AWS CLI Perintah.

## get-devices

Contoh kode berikut menunjukkan cara menggunakan `get-devices`.

## AWS CLI

Untuk mendapatkan perangkat Anda

`get-devices` Contoh berikut mendapatkan perangkat dalam jaringan global yang ditentukan.

```
aws networkmanager get-devices \  
  --global-network-id global-network-01231231231231231 \  
  --region us-west-2
```

Output:

```
{  
  "Devices": [  
    {  
      "DeviceId": "device-07f6fd08867abc123",  
      "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123",  
      "GlobalNetworkId": "global-network-01231231231231231",  
      "Description": "NY office device",  
      "Type": "office device",  
      "Vendor": "anycompany",  
      "Model": "abcabc",  
      "SerialNumber": "1234",  
      "CreatedAt": 1575554005.0,  
      "State": "AVAILABLE"  
    }  
  ]  
}
```

- Untuk detail API, lihat [GetDevices](#) di Referensi AWS CLI Perintah.

## `get-link-associations`

Contoh kode berikut menunjukkan cara menggunakan `get-link-associations`.

## AWS CLI

Untuk mendapatkan asosiasi tautan Anda

`get-link-associations` Contoh berikut mendapatkan asosiasi link dalam jaringan global yang ditentukan.

```
aws networkmanager get-link-associations \  
  --global-network-id global-network-01231231231231231 \  
  --region us-west-2
```

Output:

```
{  
  "LinkAssociations": [  
    {  
      "GlobalNetworkId": "global-network-01231231231231231",  
      "DeviceId": "device-07f6fd08867abc123",  
      "LinkId": "link-11112222aaaabbbb1",  
      "LinkAssociationState": "AVAILABLE"  
    }  
  ]  
}
```

- Untuk detail API, lihat [GetLinkAssociations](#) di Referensi AWS CLI Perintah.

## get-links

Contoh kode berikut menunjukkan cara menggunakan `get-links`.

### AWS CLI

Untuk mendapatkan tautan Anda

`get-links` Contoh berikut mendapatkan link dalam jaringan global yang ditentukan.

```
aws networkmanager get-links \  
  --global-network-id global-network-01231231231231231 \  
  --region us-west-2
```

Output:

```
{  
  "Links": [  
    {  
      "LinkId": "link-11112222aaaabbbb1",  
      "LinkArn": "arn:aws:networkmanager::123456789012:link/global-  
network-01231231231231231/link-11112222aaaabbbb1",
```

```

    "GlobalNetworkId": "global-network-01231231231231231",
    "SiteId": "site-444555aaabbb11223",
    "Description": "VPN Link",
    "Type": "broadband",
    "Bandwidth": {
      "UploadSpeed": 10,
      "DownloadSpeed": 20
    },
    "Provider": "AnyCompany",
    "CreatedAt": 1575555811.0,
    "State": "AVAILABLE"
  }
]
}

```

- Untuk detail API, lihat [GetLinks](#) di Referensi AWS CLI Perintah.

## get-object-retention

Contoh kode berikut menunjukkan cara menggunakan `get-object-retention`.

### AWS CLI

Untuk mengambil konfigurasi retensi objek untuk objek

`get-object-retention` Contoh berikut mengambil konfigurasi retensi objek untuk objek tertentu.

```

aws s3api get-object-retention \
  --bucket my-bucket-with-object-lock \
  --key doc1.rtf

```

Output:

```

{
  "Retention": {
    "Mode": "GOVERNANCE",
    "RetainUntilDate": "2025-01-01T00:00:00.000Z"
  }
}

```

- Untuk detail API, lihat [GetObjectRetention](#) di Referensi AWS CLI Perintah.

## get-public-access-block

Contoh kode berikut menunjukkan cara menggunakan `get-public-access-block`.

### AWS CLI

Untuk menyetel atau memodifikasi konfigurasi blokir akses publik untuk bucket

`get-public-access-block` Contoh berikut menampilkan konfigurasi blokir akses publik untuk bucket yang ditentukan.

```
aws s3api get-public-access-block --bucket my-bucket
```

Output:

```
{
  "PublicAccessBlockConfiguration": {
    "IgnorePublicAcls": true,
    "BlockPublicPolicy": true,
    "BlockPublicAcls": true,
    "RestrictPublicBuckets": true
  }
}
```

- Untuk detail API, lihat [GetPublicAccessBlock](#) di Referensi AWS CLI Perintah.

## get-sites

Contoh kode berikut menunjukkan cara menggunakan `get-sites`.

### AWS CLI

Untuk mendapatkan situs Anda

`get-sites` Contoh berikut mendapatkan situs dalam jaringan global yang ditentukan.

```
aws networkmanager get-sites \
  --global-network-id global-network-01231231231231231 \
  --region us-west-2
```

Output:

```
{
  "Sites": [
    {
      "SiteId": "site-444555aaabbb11223",
      "SiteArn": "arn:aws:networkmanager::123456789012:site/global-
network-01231231231231231/site-444555aaabbb11223",
      "GlobalNetworkId": "global-network-01231231231231231",
      "Description": "NY head office",
      "Location": {
        "Latitude": "40.7128",
        "Longitude": "-74.0060"
      },
      "CreatedAt": 1575554528.0,
      "State": "AVAILABLE"
    }
  ]
}
```

- Untuk detail API, lihat [GetSites](#) di Referensi AWS CLI Perintah.

## get-transit-gateway-registrations

Contoh kode berikut menunjukkan cara menggunakan `get-transit-gateway-registrations`.

### AWS CLI

Untuk mendapatkan pendaftaran gateway transit Anda

`get-transit-gateway-registrations` Contoh berikut mendapatkan gateway transit yang terdaftar ke jaringan global yang ditentukan.

```
aws networkmanager get-transit-gateway-registrations \
  --global-network-id global-network-01231231231231231 \
  --region us-west-2
```

### Output:

```
{
  "TransitGatewayRegistrations": [
    {
      "GlobalNetworkId": "global-network-01231231231231231",
```

```

        "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-
gateway/tgw-123abc05e04123abc",
        "State": {
            "Code": "AVAILABLE"
        }
    }
]
}

```

- Untuk detail API, lihat [GetTransitGatewayRegistrations](#) di Referensi AWS CLI Perintah.

## get-vpc-attachment

Contoh kode berikut menunjukkan cara menggunakan `get-vpc-attachment`.

### AWS CLI

Untuk mendapatkan lampiran VPC

`get-vpc-attachment` Contoh berikut mengembalikan informasi tentang lampiran VPC.

```

aws networkmanager get-vpc-attachment \
  --attachment-id attachment-03b7ea450134787da

```

### Output:

```

{
  "VpcAttachment": {
    "Attachment": {
      "CoreNetworkId": "core-network-0522de1b226a5d7b3",
      "AttachmentId": "attachment-03b7ea450134787da",
      "OwnerAccountId": "987654321012",
      "AttachmentType": "VPC",
      "State": "CREATING",
      "EdgeLocation": "us-east-1",
      "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/vpc-a7c4bbda",
      "Tags": [
        {
          "Key": "Name",
          "Value": "DevVPC"
        }
      ],
    },
  },
}

```

```
    "CreatedAt": "2022-03-11T17:48:58+00:00",
    "UpdatedAt": "2022-03-11T17:48:58+00:00"
  },
  "SubnetArns": [
    "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-202cde6c",
    "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-e5022dba",
    "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-2387ae02",
    "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-cda9dfffc"
  ],
  "Options": {
    "Ipv6Support": false
  }
}
```

Untuk informasi selengkapnya, lihat [Lampiran](#) di Panduan Pengguna Cloud WAN.

- Untuk detail API, lihat [GetVpcAttachment](#) di Referensi AWS CLI Perintah.

## list-bucket-analytics-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-bucket-analytics-configurations`.

### AWS CLI

Untuk mengambil daftar konfigurasi analitik untuk bucket

Berikut ini akan `list-bucket-analytics-configurations` mengambil daftar konfigurasi analitik untuk bucket yang ditentukan.

```
aws s3api list-bucket-analytics-configurations \
  --bucket my-bucket
```

Output:

```
{
  "AnalyticsConfigurationList": [
    {
      "StorageClassAnalysis": {},
      "Id": "1"
    }
  ]
}
```



```
  ],  
  "IsTruncated": false  
}
```

- Untuk detail API, lihat [ListBucketAnalyticsConfigurations](#) di Referensi AWS CLI Perintah.

## list-bucket-metrics-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-bucket-metrics-configurations`.

### AWS CLI

Untuk mengambil daftar konfigurasi metrik untuk bucket

`list-bucket-metrics-configurations` Contoh berikut mengambil daftar konfigurasi metrik untuk bucket yang ditentukan.

```
aws s3api list-bucket-metrics-configurations \  
  --bucket my-bucket
```

Output:

```
{  
  "IsTruncated": false,  
  "MetricsConfigurationList": [  
    {  
      "Filter": {  
        "Prefix": "logs"  
      },  
      "Id": "123"  
    },  
    {  
      "Filter": {  
        "Prefix": "tmp"  
      },  
      "Id": "234"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListBucketMetricsConfigurations](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk sumber daya perangkat tertentu (`device-07f6fd08867abc123`).

```
aws networkmanager list-tags-for-resource \
  --resource-arn arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231231/device-07f6fd08867abc123 \
  --region us-west-2
```

Output:

```
{
  "TagList": [
    {
      "Key": "Network",
      "Value": "Northeast"
    }
  ]
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## put-bucket-metrics-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-metrics-configuration`.

### AWS CLI

Untuk menyetel konfigurasi metrik untuk bucket

`put-bucket-metrics-configuration` Contoh berikut menetapkan konfigurasi metrik dengan ID 123 untuk bucket yang ditentukan.

```
aws s3api put-bucket-metrics-configuration \
  --bucket my-bucket \
```

```
--id 123 \  
--metrics-configuration '{"Id": "123", "Filter": {"Prefix": "logs"}}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketMetricsConfiguration](#) di Referensi AWS CLI Perintah.

## put-object-retention

Contoh kode berikut menunjukkan cara menggunakan `put-object-retention`.

### AWS CLI

Untuk mengatur konfigurasi retensi objek untuk objek

`put-object-retention` Contoh berikut menetapkan konfigurasi retensi objek untuk objek yang ditentukan hingga 2025-01-01.

```
aws s3api put-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf \  
  --retention '{ "Mode": "GOVERNANCE", "RetainUntilDate": "2025-01-01T00:00:00" }'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutObjectRetention](#) di Referensi AWS CLI Perintah.

## put-public-access-block

Contoh kode berikut menunjukkan cara menggunakan `put-public-access-block`.

### AWS CLI

Untuk mengatur konfigurasi blokir akses publik untuk bucket

`put-public-access-block` Contoh berikut menetapkan konfigurasi akses publik blok restriktif untuk bucket yang ditentukan.

```
aws s3api put-public-access-block \  
  --bucket my-bucket \  
  --public-access-block-configuration  
  "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutPublicAccessBlock](#) di Referensi AWS CLI Perintah.

## register-transit-gateway

Contoh kode berikut menunjukkan cara menggunakan `register-transit-gateway`.

### AWS CLI

Untuk mendaftarkan gateway transit di jaringan global

`register-transit-gateway` Contoh berikut mendaftarkan gateway transit `tgw-123abc05e04123abc` di jaringan global yang ditentukan.

```
aws networkmanager register-transit-gateway \
  --global-network-id global-network-01231231231231231 \
  --transit-gateway-arn arn:aws:ec2:us-west-2:123456789012:transit-gateway/
tgw-123abc05e04123abc \
  --region us-west-2
```

Output:

```
{
  "TransitGatewayRegistration": {
    "GlobalNetworkId": "global-network-01231231231231231",
    "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway/
tgw-123abc05e04123abc",
    "State": {
      "Code": "PENDING"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Pendaftaran Gateway Transit](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [RegisterTransitGateway](#) di Referensi AWS CLI Perintah.

## reject-attachment

Contoh kode berikut menunjukkan cara menggunakan `reject-attachment`.

## AWS CLI

Untuk menolak lampiran

`reject-attachment` Contoh berikut menolak permintaan lampiran VPC.

```
aws networkmanager reject-attachment \  
  --attachment-id attachment-03b7ea450134787da
```

Output:

```
{  
  "Attachment": {  
    "CoreNetworkId": "core-network-0522de1b226a5d7b3",  
    "AttachmentId": "attachment-03b7ea450134787da",  
    "OwnerAccountId": "987654321012",  
    "AttachmentType": "VPC",  
    "State": "AVAILABLE",  
    "EdgeLocation": "us-east-1",  
    "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/vpc-a7c4bbda",  
    "CreatedAt": "2022-03-11T17:48:58+00:00",  
    "UpdatedAt": "2022-03-11T17:51:25+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Penerimaan lampiran](#) di Panduan Pengguna Cloud WAN.

- Untuk detail API, lihat [RejectAttachment](#) di Referensi AWS CLI Perintah.

## `start-route-analysis`

Contoh kode berikut menunjukkan cara menggunakan `start-route-analysis`.

### AWS CLI

Untuk memulai analisis rute

`start-route-analysis` Contoh berikut memulai analisis antara sumber dan tujuan, termasuk opsional `include-return-path`.

```
aws networkmanager start-route-analysis \  
  --global-network-id global-network-00aa0aaa0b0aaa000 \  
  --source-availability-zone us-east-1a \  
  --destination-availability-zone us-east-1b
```

```
--source TransitGatewayAttachmentArn=arn:aws:ec2:us-east-1:503089527312:transit-
gateway-attachment/tgw-attach-0d4a2d491bf68c093,IpAddress=10.0.0.0 \
--destination TransitGatewayAttachmentArn=arn:aws:ec2:us-
west-1:503089527312:transit-gateway-attachment/tgw-
attach-002577f30bb181742,IpAddress=11.0.0.0 \
--include-return-path
```

### Output:

```
{
  "RouteAnalysis": {
    "GlobalNetworkId": "global-network-00aa0aaa0b0aaa000",
    "OwnerAccountId": "1111222233333",
    "RouteAnalysisId": "a1873de1-273c-470c-1a2bc2345678",
    "StartTimestamp": 1695760154.0,
    "Status": "RUNNING",
    "Source": {
      "TransitGatewayAttachmentArn": "arn:aws:ec2:us-
east-1:111122223333:transit-gateway-attachment/tgw-attach-1234567890abcdef0",
      "TransitGatewayArn": "arn:aws:ec2:us-east-1:111122223333:transit-
gateway/tgw-abcdef01234567890",
      "IpAddress": "10.0.0.0"
    },
    "Destination": {
      "TransitGatewayAttachmentArn": "arn:aws:ec2:us-
west-1:555555555555:transit-gateway-attachment/tgw-attach-021345abcdef6789",
      "TransitGatewayArn": "arn:aws:ec2:us-west-1:111122223333:transit-
gateway/tgw-09876543210fedcba0",
      "IpAddress": "11.0.0.0"
    },
    "IncludeReturnPath": true,
    "UseMiddleboxes": false
  }
}
```

Untuk informasi selengkapnya, lihat [Route Analyzer](#) di Panduan Pengguna AWS Global Networks for Transit Gateways.

- Untuk detail API, lihat [StartRouteAnalysis](#) di Referensi AWS CLI Perintah.

### tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menerapkan tag ke sumber daya

`tag-resource` Contoh berikut menerapkan tag `Network=Northeast` ke perangkat `device-07f6fd08867abc123`.

```
aws networkmanager tag-resource \  
  --resource-arn arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231231/device-07f6fd08867abc123 \  
  --tags Key=Network,Value=Northeast \  
  --region us-west-2
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dengan kunci `Network` dari perangkat `device-07f6fd08867abc123`.

```
aws networkmanager untag-resource \  
  --resource-arn arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231231/device-07f6fd08867abc123 ]  
  --tag-keys Network \  
  --region us-west-2
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-device**

Contoh kode berikut menunjukkan cara menggunakan `update-device`.

## AWS CLI

Untuk memperbarui perangkat

update-device Contoh berikut memperbarui perangkat device-07f6fd08867abc123 dengan menentukan ID situs untuk perangkat.

```
aws networkmanager update-device \  
  --global-network-id global-network-01231231231231231 \  
  --device-id device-07f6fd08867abc123 \  
  --site-id site-444555aaabbb11223 \  
  --region us-west-2
```

Output:

```
{  
  "Device": {  
    "DeviceId": "device-07f6fd08867abc123",  
    "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "NY office device",  
    "Type": "Office device",  
    "Vendor": "anycompany",  
    "Model": "abcabc",  
    "SerialNumber": "1234",  
    "SiteId": "site-444555aaabbb11223",  
    "CreatedAt": 1575554005.0,  
    "State": "UPDATING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Perangkat](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [UpdateDevice](#) di Referensi AWS CLI Perintah.

### update-global-network

Contoh kode berikut menunjukkan cara menggunakan update-global-network.



## AWS CLI

Untuk memperbarui jaringan global

`update-global-network` Contoh berikut memperbarui deskripsi untuk jaringan global `global-network-01231231231231231`.

```
aws networkmanager update-global-network \  
  --global-network-id global-network-01231231231231231 \  
  --description "Head offices" \  
  --region us-west-2
```

Output:

```
{  
  "GlobalNetwork": {  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "GlobalNetworkArn": "arn:aws:networkmanager::123456789012:global-network/  
global-network-01231231231231231",  
    "Description": "Head offices",  
    "CreatedAt": 1575553525.0,  
    "State": "UPDATING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Jaringan Global](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [UpdateGlobalNetwork](#) di Referensi AWS CLI Perintah.

## update-link

Contoh kode berikut menunjukkan cara menggunakan `update-link`.

## AWS CLI

Untuk memperbarui tautan

`update-link` Contoh berikut memperbarui informasi bandwidth untuk `linklink-11112222aaaabbbb1`.

```
aws networkmanager update-link \  
  --link-id linklink-11112222aaaabbbb1
```

```
--global-network-id global-network-01231231231231231 \  
--link-id link-11112222aaaabbbb1 \  
--bandwidth UploadSpeed=20,DownloadSpeed=20 \  
--region us-west-2
```

### Output:

```
{  
  "Link": {  
    "LinkId": "link-11112222aaaabbbb1",  
    "LinkArn": "arn:aws:networkmanager::123456789012:link/global-  
network-01231231231231231/link-11112222aaaabbbb1",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "SiteId": "site-444555aaaabbb11223",  
    "Description": "VPN Link",  
    "Type": "broadband",  
    "Bandwidth": {  
      "UploadSpeed": 20,  
      "DownloadSpeed": 20  
    },  
    "Provider": "AnyCompany",  
    "CreatedAt": 1575555811.0,  
    "State": "UPDATING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Tautan](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [UpdateLink](#) di Referensi AWS CLI Perintah.

## update-site

Contoh kode berikut menunjukkan cara menggunakan `update-site`.

### AWS CLI

Untuk memperbarui situs

`update-site` Contoh berikut memperbarui deskripsi untuk situs `site-444555aaaabbb11223` di jaringan global yang ditentukan.

```
aws networkmanager update-site \  

```

```
--global-network-id global-network-01231231231231231 \  
--site-id site-444555aaabbb11223 \  
--description "New York Office site" \  
--region us-west-2
```

### Output:

```
{  
  "Site": {  
    "SiteId": "site-444555aaabbb11223",  
    "SiteArn": "arn:aws:networkmanager::123456789012:site/global-  
network-01231231231231231/site-444555aaabbb11223",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York Office site",  
    "Location": {  
      "Latitude": "40.7128",  
      "Longitude": "-74.0060"  
    },  
    "CreatedAt": 1575554528.0,  
    "State": "UPDATING"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Situs](#) di Panduan Manajer Jaringan Gateway Transit.

- Untuk detail API, lihat [UpdateSite](#) di Referensi AWS CLI Perintah.

## Contoh Nimble Studio menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Nimble Studio.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **get-eula**

Contoh kode berikut menunjukkan cara menggunakan `get-eula`.

#### AWS CLI

Untuk mendapatkan informasi tentang studio Anda

`get-eula` Contoh berikut mencantumkan informasi tentang EULA.

```
aws nimble get-eula \  
  --eula-id "EULAid"
```

#### Output:

```
{  
  "eula": {  
    "content": "https://www.mozilla.org/en-US/MPL/2.0/",  
    "createdAt": "2021-04-20T16:45:23+00:00",  
    "eulaId": "gJZLygd-Srq_5NNbSfiaLg",  
    "name": "Mozilla-FireFox",  
    "updatedAt": "2021-04-20T16:45:23+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menerima EULA](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [GetEula](#) di Referensi AWS CLI Perintah.

### **get-launch-profile-details**

Contoh kode berikut menunjukkan cara menggunakan `get-launch-profile-details`.

#### AWS CLI

Untuk daftar widget yang tersedia

`get-launch-profile-details` Contoh berikut mencantumkan detail tentang profil peluncuran.

```
aws nimble get-launch-profile-details \  
  --studio-id "StudioID" \  
  --launch-profile-id "LaunchProfileID"
```

Output:

```
{  
  "launchProfile": {  
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/  
yeG71DwNQEiwNTRT7DrV7Q",  
    "createdAt": "2022-01-27T21:18:59+00:00",  
    "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",  
    "description": "The Launch Profile for the Render workers created by  
StudioBuilder.",  
    "ec2SubnetIds": [  
      "subnet-EXAMPLE11111"  
    ],  
    "launchProfileId": "yeG71DwNQEiwNTRT7DrV7Q",  
    "launchProfileProtocolVersions": [  
      "2021-03-31"  
    ],  
    "name": "RenderWorker-Default",  
    "state": "READY",  
    "statusCode": "LAUNCH_PROFILE_CREATED",  
    "statusMessage": "Launch Profile has been created",  
    "streamConfiguration": {  
      "clipboardMode": "ENABLED",  
      "ec2InstanceTypes": [  
        "g4dn.4xlarge",  
        "g4dn.8xlarge"  
      ],  
      "maxSessionLengthInMinutes": 690,  
      "maxStoppedSessionLengthInMinutes": 0,  
      "streamingImageIds": [  
        "Cw_jXnp1QcSSXhE2hkNRoQ",  
        "YGXAqgoWTnCNSV8VP20sHQ"  
      ]  
    },  
    "studioComponentIds": [  
      "_hR_-RaAReS0jAnLakbX7Q",  
      "vQ5w_TbIRayPkAZgcbyYRA",  
    ]  
  }  
}
```

```
    "ZQuMxN99Qfa_Js6ma9TwdA",
    "45Kj0SPPRzK20yvpCuQ6qw"
  ],
  "tags": {
    "resourceArn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/
yeG7lDwNQEiwNTRT7DrV7Q"
  },
  "updatedAt": "2022-01-27T21:19:13+00:00",
  "updatedBy": "AR0A3002NEHCCYRNDDIFT:i-00b98256b04d9e989",
  "validationResults": [
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SUBNET_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
    }
  ]
},
"streamingImages": [
  {
    "arn": "arn:aws:nimble:us-west-2:123456789012:streaming-image/
Cw_jXnp1QcSSXhE2hkNRoQ",
    "description": "Base windows image for NimbleStudio",
    "ec2ImageId": "ami-EXAMPLE11111",
    "eulaIds": [
      "gJZLygd-Srq_5NNbSfiaLg",
```

```

        "ggK2eIw6RQyt8PIee01D3g",
        "a-D9Wc0VQCKUfxAinCDxaw",
        "RvoNmVXiSrS4LhLTb6ybkw",
        "wtp85BcSTa2NZeNRnMKdjw",
        "R1-J0fM5S12hyIiwWIV6hw"
    ],
    "name": "NimbleStudioWindowsStreamImage",
    "owner": "amazon",
    "platform": "WINDOWS",
    "state": "READY",
    "streamingImageId": "Cw_jXnp1QcSSXhE2hkNRoQ",
    "tags": {
        "resourceArn": "arn:aws:nimble:us-west-2:123456789012:streaming-
image/Cw_jXnp1QcSSXhE2hkNRoQ"
    }
},
{
    "arn": "arn:aws:nimble:us-west-2:123456789012:streaming-image/
YGXAqgoWTnCNSV8VP20sHQ",
    "description": "Base linux image for NimbleStudio",
    "ec2ImageId": "ami-EXAMPLE11111",
    "eulaIds": [
        "gJZLygd-Srq_5NNbSfiaLg",
        "ggK2eIw6RQyt8PIee01D3g",
        "a-D9Wc0VQCKUfxAinCDxaw",
        "RvoNmVXiSrS4LhLTb6ybkw",
        "wtp85BcSTa2NZeNRnMKdjw",
        "R1-J0fM5S12hyIiwWIV6hw"
    ],
    "name": "NimbleStudioLinuxStreamImage",
    "owner": "amazon",
    "platform": "LINUX",
    "state": "READY",
    "streamingImageId": "YGXAqgoWTnCNSV8VP20sHQ",
    "tags": {
        "resourceArn": "arn:aws:nimble:us-west-2:123456789012:streaming-
image/YGXAqgoWTnCNSV8VP20sHQ"
    }
}
],
"studioComponentSummaries": [
    {
        "description": "FSx for Windows",
        "name": "FSxWindows",

```

```

        "studioComponentId": "ZQuMxN99Qfa_Js6ma9TwdA",
        "subtype": "AMAZON_FSX_FOR_WINDOWS",
        "type": "SHARED_FILE_SYSTEM"
    },
    {
        "description": "Instance configuration studio component.",
        "name": "InstanceConfiguration",
        "studioComponentId": "vQ5w_TbIRayPkAZgcbyYRA",
        "subtype": "CUSTOM",
        "type": "CUSTOM"
    },
    {
        "name": "ActiveDirectory",
        "studioComponentId": "_hR_-RaAReS0jAnLakbX7Q",
        "subtype": "AWS_MANAGED_MICROSOFT_AD",
        "type": "ACTIVE_DIRECTORY"
    },
    {
        "description": "Render farm running Deadline",
        "name": "RenderFarm",
        "studioComponentId": "45Kj0SPPRzK20yvpCuQ6qw",
        "subtype": "CUSTOM",
        "type": "COMPUTE_FARM"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat profil peluncuran](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [GetLaunchProfileDetails](#) di Referensi AWS CLI Perintah.

## get-launch-profile

Contoh kode berikut menunjukkan cara menggunakan `get-launch-profile`.

### AWS CLI

Untuk daftar widget yang tersedia

`get-launch-profile` Contoh berikut mencantumkan informasi tentang profil peluncuran.

```
aws nimble get-launch-profile \
```



```
--studio-id "StudioID" \  
--launch-profile-id "LaunchProfileID"
```

## Output:

```
{  
  "launchProfile": {  
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/  
yeG7lDwNQEiwNTRT7DrV7Q",  
    "createdAt": "2022-01-27T21:18:59+00:00",  
    "createdBy": "ARO0A3002NEHCCYRNDIIFT:i-EXAMPLE11111",  
    "description": "The Launch Profile for the Render workers created by  
StudioBuilder.",  
    "ec2SubnetIds": [  
      "subnet-EXAMPLE11111"  
    ],  
    "launchProfileId": "yeG7lDwNQEiwNTRT7DrV7Q",  
    "launchProfileProtocolVersions": [  
      "2021-03-31"  
    ],  
    "name": "RenderWorker-Default",  
    "state": "READY",  
    "statusCode": "LAUNCH_PROFILE_CREATED",  
    "statusMessage": "Launch Profile has been created",  
    "streamConfiguration": {  
      "clipboardMode": "ENABLED",  
      "ec2InstanceTypes": [  
        "g4dn.4xlarge",  
        "g4dn.8xlarge"  
      ],  
      "maxSessionLengthInMinutes": 690,  
      "maxStoppedSessionLengthInMinutes": 0,  
      "streamingImageIds": [  
        "Cw_jXnp1QcSSXhE2hkNRoQ",  
        "YGXAqgoWTnCNSV8VP20sHQ"  
      ]  
    },  
    "studioComponentIds": [  
      "_hR_-RaAReS0jAnLakbX7Q",  
      "vQ5w_TbIRayPkAZgcbyYRA",  
      "ZQuMxN99Qfa_Js6ma9TwdA",  
      "45Kj0SPPRzK20yvpCuQ6qw"  
    ],  
  },  
}
```

```
"tags": {},
"updatedAt": "2022-01-27T21:19:13+00:00",
"updatedBy": "AR0A3002NEHCCYRNDDIFT:i-00b98256b04d9e989",
"validationResults": [
  {
    "state": "VALIDATION_SUCCESS",
    "statusCode": "VALIDATION_SUCCESS",
    "statusMessage": "The validation succeeded.",
    "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
  },
  {
    "state": "VALIDATION_SUCCESS",
    "statusCode": "VALIDATION_SUCCESS",
    "statusMessage": "The validation succeeded.",
    "type": "VALIDATE_SUBNET_ASSOCIATION"
  },
  {
    "state": "VALIDATION_SUCCESS",
    "statusCode": "VALIDATION_SUCCESS",
    "statusMessage": "The validation succeeded.",
    "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
  },
  {
    "state": "VALIDATION_SUCCESS",
    "statusCode": "VALIDATION_SUCCESS",
    "statusMessage": "The validation succeeded.",
    "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Membuat profil peluncuran](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [GetLaunchProfile](#) di Referensi AWS CLI Perintah.

## get-studio

Contoh kode berikut menunjukkan cara menggunakan `get-studio`.

## AWS CLI

Untuk mendapatkan informasi tentang studio Anda

`get-studio` Contoh berikut mencantumkan studio di AWS akun Anda.

```
aws nimble get-studio \  
  --studio-id "StudioID"
```

Output:

```
{  
  "studio": {  
    "adminRoleArn": "arn:aws:iam::123456789012:role/studio-admin-role",  
    "arn": "arn:aws:nimble:us-west-2:123456789012:studio/stid-EXAMPLE11111",  
    "createdAt": "2022-01-27T20:29:35+00:00",  
    "displayName": "studio-name",  
    "homeRegion": "us-west-2",  
    "ssoClientId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "state": "READY",  
    "statusCode": "STUDIO_CREATED",  
    "statusMessage": "The studio has been created successfully ",  
    "studioEncryptionConfiguration": {  
      "keyType": "AWS_OWNED_KEY"  
    },  
    "studioId": "us-west-2:stid-EXAMPLE11111",  
    "studioName": "studio-name",  
    "studioUrl": "https://studio-name.nimblestudio.us-west-2.amazonaws.com",  
    "tags": {},  
    "updatedAt": "2022-01-27T20:29:37+00:00",  
    "userRoleArn": "arn:aws:iam::123456789012:role/studio-user-role"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Apa itu Amazon Nimble Studio?](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [GetStudio](#) di Referensi AWS CLI Perintah.

## **list-eula-acceptances**

Contoh kode berikut menunjukkan cara menggunakan `list-eula-acceptances`.

## AWS CLI

Untuk daftar widget yang tersedia

`list-eula-acceptances` Contoh berikut mencantumkan EULA yang diterima di akun Anda AWS .

```
aws nimble list-eula-acceptances \  
  --studio-id "StudioID"
```

Output:

```
{  
  "eulaAcceptances": [  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "V0JlpZQaSx6yHcUuX0qfQw",  
      "eulaId": "R1-J0fM5S12hyIiwWIV6hw"  
    },  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "YY_uDFW-SVibc627qbug0Q",  
      "eulaId": "RvoNmVXiSrS4LhLTb6ybkw"  
    },  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "ov087PnhQ4-MpttiL5uN6Q",  
      "eulaId": "a-D9Wc0VQCKUfxAinCDxaw"  
    },  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "5YeXje4yR0amuTESGvqIAQ",  
      "eulaId": "gJZLygd-Srq_5NNbSfiaLg"  
    },  
    {
```

```

    "acceptedAt": "2022-01-28T17:44:35+00:00",
    "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",
    "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",
    "eulaAcceptanceId": "W1sIn8PtScqeJEn8sxxhgw",
    "eulaId": "ggK2eIw6RQyt8PIee01D3g"
  },
  {
    "acceptedAt": "2022-01-28T17:44:35+00:00",
    "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",
    "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",
    "eulaAcceptanceId": "Zq9KNEQPRMWJ7FolSoQgUA",
    "eulaId": "wtp85BcSTa2NZeNRnMKdjw"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menerima EULA](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [ListEulaAcceptances](#) di Referensi AWS CLI Perintah.

## list-eulas

Contoh kode berikut menunjukkan cara menggunakan `list-eulas`.

### AWS CLI

Untuk daftar widget yang tersedia

`list-eulas` Contoh berikut mencantumkan EULA di akun Anda AWS .

```
aws nimble list-eulas
```

Output:

```

{
  "eulas": [
    {
      "content": "https://www.mozilla.org/en-US/MPL/2.0/",
      "createdAt": "2021-04-20T16:45:23+00:00",
      "eulaId": "gJZLygd-Srq_5NNbSfiaLg",
      "name": "Mozilla-FireFox",
      "updatedAt": "2021-04-20T16:45:23+00:00"
    }
  ]
}

```

```
  },
  {
    "content": "https://www.awsthinkbox.com/end-user-license-agreement",
    "createdAt": "2021-04-20T16:45:24+00:00",
    "eulaId": "RvoNmVXiSrS4LhLTb6ybkw",
    "name": "Thinkbox-Deadline",
    "updatedAt": "2021-04-20T16:45:24+00:00"
  },
  {
    "content": "https://www.videolan.org/legal.html",
    "createdAt": "2021-04-20T16:45:24+00:00",
    "eulaId": "Rl-J0fM5Sl2hyIiwWIV6hw",
    "name": "Videolan-VLC",
    "updatedAt": "2021-04-20T16:45:24+00:00"
  },
  {
    "content": "https://code.visualstudio.com/license",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "ggK2eIw6RQyt8PIee0lD3g",
    "name": "Microsoft-VSCode",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://darbyjohnston.github.io/DJV/legal.html#License",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "wtp85BcSTa2NZeNRnMKdjw",
    "name": "DJV-DJV",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://www.sidefx.com/legal/license-agreement/",
    "createdAt": "2021-04-20T16:45:24+00:00",
    "eulaId": "uu2VDLo-QJeIGWWLBae_UA",
    "name": "SideFX-Houdini",
    "updatedAt": "2021-04-20T16:45:24+00:00"
  },
  {
    "content": "https://www.chaosgroup.com/eula",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "L0HS4P3CRYKVXc2J2L07Vw",
    "name": "ChaosGroup-Vray",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
```

```

    "content": "https://www.foundry.com/eula",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "SAuhfHmmsAeUuq3wsMiMlw",
    "name": "Foundry-Nuke",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://download.blender.org/release/GPL3-license.txt",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "a-D9Wc0VQCKUfxAinCDxaw",
    "name": "BlenderFoundation-Blender",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menerima EULA](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [ListEulas](#) di Referensi AWS CLI Perintah.

## list-launch-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-launch-profiles`.

### AWS CLI

Untuk daftar widget yang tersedia

`list-launch-profiles` Contoh berikut mencantumkan profil peluncuran di AWS akun Anda.

```
aws nimble list-launch-profiles \
  --studio-id "StudioID"
```

Output:

```

{
  "launchProfiles": [
    {
      "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/
yeG7lDwNQEiwNTRT7DrV7Q",
      "createdAt": "2022-01-27T21:18:59+00:00",
      "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",

```

```
"description": "The Launch Profile for the Render workers created by
StudioBuilder.",
"ec2SubnetIds": [
  "subnet-EXAMPLE11111"
],
"launchProfileId": "yeG7lDwNQEiwNTRT7DrV7Q",
"launchProfileProtocolVersions": [
  "2021-03-31"
],
"name": "RenderWorker-Default",
"state": "READY",
"statusCode": "LAUNCH_PROFILE_CREATED",
"statusMessage": "Launch Profile has been created",
"streamConfiguration": {
  "clipboardMode": "ENABLED",
  "ec2InstanceTypes": [
    "g4dn.4xlarge",
    "g4dn.8xlarge"
  ],
  "maxSessionLengthInMinutes": 690,
  "maxStoppedSessionLengthInMinutes": 0,
  "streamingImageIds": [
    "Cw_jXnp1QcSSXhE2hkNRoQ",
    "YGXAqgoWTnCNSV8VP20sHQ"
  ]
},
"studioComponentIds": [
  "_hR_-RaAReS0jAnLakbX7Q",
  "vQ5w_TbIRayPkAZgcbyYRA",
  "ZQuMxN99Qfa_Js6ma9TwdA",
  "45Kj0SPPrzK20yvpCuQ6qw"
],
"tags": {},
"updatedAt": "2022-01-27T21:19:13+00:00",
"updatedBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",
"validationResults": [
  {
    "state": "VALIDATION_SUCCESS",
    "statusCode": "VALIDATION_SUCCESS",
    "statusMessage": "The validation succeeded.",
    "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
  },
  {
    "state": "VALIDATION_SUCCESS",
```



```
        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_SUBNET_ASSOCIATION"
    },
    {
        "state": "VALIDATION_SUCCESS",
        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
    },
    {
        "state": "VALIDATION_SUCCESS",
        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
    }
]
},
{
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/
jDCIm1jRSaa9e44PZ3w7gg",
    "createdAt": "2022-01-27T21:19:26+00:00",
    "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",
    "description": "This Workstation Launch Profile was created by
StudioBuilder",
    "ec2SubnetIds": [
        "subnet-046f4205ae535b2cc"
    ],
    "launchProfileId": "jDCIm1jRSaa9e44PZ3w7gg",
    "launchProfileProtocolVersions": [
        "2021-03-31"
    ],
    "name": "Workstation-Default",
    "state": "READY",
    "statusCode": "LAUNCH_PROFILE_CREATED",
    "statusMessage": "Launch Profile has been created",
    "streamConfiguration": {
        "clipboardMode": "ENABLED",
        "ec2InstanceTypes": [
            "g4dn.4xlarge",
            "g4dn.8xlarge"
        ],
    },
    "maxSessionLengthInMinutes": 690,
    "maxStoppedSessionLengthInMinutes": 0,
```

```
    "streamingImageIds": [
      "Cw_jXnp1QcSSXhE2hkNRoQ",
      "YGXAqgoWTnCNSV8VP20sHQ"
    ]
  },
  "studioComponentIds": [
    "_hR_-RaAReS0jAnLakbX7Q",
    "vQ5w_TbIRayPkAZgcbyYRA",
    "ZQuMxN99Qfa_Js6ma9TwdA",
    "yJSbsHXAQYwk9FXLNusX1Q",
    "45Kj0SPPrzK20yvpCuQ6qw"
  ],
  "tags": {},
  "updatedAt": "2022-01-27T21:19:40+00:00",
  "updatedBy": "AR0A3002NEHCCYRNDDIFT:i-EXAMPLE11111",
  "validationResults": [
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SUBNET_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Membuat profil peluncuran](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [ListLaunchProfiles](#) di Referensi AWS CLI Perintah.

## list-studio-components

Contoh kode berikut menunjukkan cara menggunakan `list-studio-components`.

### AWS CLI

Untuk daftar widget yang tersedia

`list-studio-components` Contoh berikut mencantumkan komponen studio di AWS akun Anda.

```
aws nimble list-studio-components \
  --studio-id "StudioID"
```

Output:

```
{
  "studioComponents": [
    {
      "arn": "arn:aws:nimble:us-west-2:123456789012:studio-component/
ZQuMxN99Qfa_Js6ma9TwdA",
      "configuration": {
        "sharedFileSystemConfiguration": {
          "fileSystemId": "fs-EXAMPLE11111",
          "linuxMountPoint": "/mnt/fsxshare",
          "shareName": "share",
          "windowsMountDrive": "Z"
        }
      },
      "createdAt": "2022-01-27T21:15:34+00:00",
      "createdBy": "AROA3002NEHCCYRNDIIFT:i-EXAMPLE11111",
      "description": "FSx for Windows",
      "ec2SecurityGroupIds": [
        "sg-EXAMPLE11111"
      ],
    }
  ]
}
```

```

        "name": "FSxWindows",
        "state": "READY",
        "statusCode": "STUDIO_COMPONENT_CREATED",
        "statusMessage": "Studio Component has been created",
        "studioComponentId": "ZQuMxN99Qfa_Js6ma9TwdA",
        "subtype": "AMAZON_FSX_FOR_WINDOWS",
        "tags": {},
        "type": "SHARED_FILE_SYSTEM",
        "updatedAt": "2022-01-27T21:15:35+00:00",
        "updatedBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111"
    },
    ...
}

```

Untuk informasi selengkapnya, lihat [Cara StudioBuilder bekerja dengan Amazon Nimble Studio di Panduan Pengguna Amazon Nimble Studio](#).

- Untuk detail API, lihat [ListStudioComponents](#) di Referensi AWS CLI Perintah.

## list-studio-members

Contoh kode berikut menunjukkan cara menggunakan `list-studio-members`.

### AWS CLI

Untuk daftar widget yang tersedia

`list-studio-members` Contoh berikut mencantumkan anggota studio yang tersedia di AWS akun Anda.

```
aws nimble list-studio-members \
  --studio-id "StudioID"
```

Output:

```
{
  "members": [
    {
      "identityStoreId": "d-EXAMPLE11111",
      "persona": "ADMINISTRATOR",
      "principalId": "EXAMPLE11111-e9fd012a-94ad-4f16-9866-c69a63ab6486"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Menambahkan pengguna studio](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [ListStudioMembers](#) di Referensi AWS CLI Perintah.

## list-studios

Contoh kode berikut menunjukkan cara menggunakan `list-studios`.

### AWS CLI

Untuk membuat daftar studio Anda

`list-studios` Contoh berikut mencantumkan studio di AWS akun Anda.

```
aws nimble list-studios
```

Output:

```
{
  "studios": [
    {
      "adminRoleArn": "arn:aws:iam::123456789012:role/studio-admin-role",
      "arn": "arn:aws:nimble:us-west-2:123456789012:studio/studio-id",
      "createdAt": "2022-01-27T20:29:35+00:00",
      "displayName": "studio-name",
      "homeRegion": "us-west-2",
      "ssoClientId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "state": "READY",
      "statusCode": "STUDIO_CREATED",
      "statusMessage": "The studio has been created successfully ",
      "studioEncryptionConfiguration": {
        "keyType": "AWS_OWNED_KEY"
      },
      "studioId": "us-west-2:studio-id",
      "studioName": "studio-name",
      "studioUrl": "https://studio-name.nimblestudio.us-west-2.amazonaws.com",
      "tags": {},
      "updatedAt": "2022-01-27T20:29:37+00:00",
```

```
        "userRoleArn": "arn:aws:iam::123456789012:role/studio-user-role"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Apa itu Amazon Nimble Studio?](#) di Panduan Pengguna Amazon Nimble Studio.

- Untuk detail API, lihat [ListStudios](#) di Referensi AWS CLI Perintah.

## OpenSearch Contoh layanan menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan OpenSearch Layanan AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-elasticsearch-domain**

Contoh kode berikut menunjukkan cara menggunakan `create-elasticsearch-domain`.

AWS CLI

Untuk membuat domain Amazon Elasticsearch Service

`create-elasticsearch-domain` Perintah berikut membuat domain Amazon Elasticsearch Service baru dalam VPC dan membatasi akses ke satu pengguna. Amazon ES menyimpulkan ID VPC dari subnet dan ID grup keamanan yang ditentukan.

```
aws es create-elasticsearch-domain \  
  --domain-name vpc-cli-example \  
  --elasticsearch-version 6.2 \  
  --elasticsearch-cluster-config  
InstanceType=m4.large.elasticsearch,InstanceCount=1 \  
  --efs-options EBSEnabled=true,VolumeType=standard,VolumeSize=10 \  
  --access-policies '{"Version": "2012-10-17", "Statement": [ { "Effect":  
"Allow", "Principal": {"AWS": "arn:aws:iam::123456789012:root" }, "Action": "es:*",  
"Resource": "arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example/*" } ] }' \  
  --vpc-options SubnetIds=subnet-1a2a3a4a,SecurityGroupIds=sg-2a3a4a5a
```

### Output:

```
{  
  "DomainStatus": {  
    "ElasticsearchClusterConfig": {  
      "DedicatedMasterEnabled": false,  
      "InstanceCount": 1,  
      "ZoneAwarenessEnabled": false,  
      "InstanceType": "m4.large.elasticsearch"  
    },  
    "DomainId": "123456789012/vpc-cli-example",  
    "CognitoOptions": {  
      "Enabled": false  
    },  
    "VPCOptions": {  
      "SubnetIds": [  
        "subnet-1a2a3a4a"  
      ],  
      "VPCId": "vpc-3a4a5a6a",  
      "SecurityGroupIds": [  
        "sg-2a3a4a5a"  
      ],  
      "AvailabilityZones": [  
        "us-west-1c"  
      ]  
    },  
    "Created": true,  
    "Deleted": false,  
    "EBSOptions": {  
      "VolumeSize": 10,  
      "VolumeType": "standard",  
      "EBSEnabled": true  
    }  
  }  
}
```

```

    },
    "Processing": true,
    "DomainName": "vpc-cli-example",
    "SnapshotOptions": {
      "AutomatedSnapshotStartHour": 0
    },
    },
    "ElasticsearchVersion": "6.2",
    "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example/*\"}]}",
    "AdvancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    },
    "EncryptionAtRestOptions": {
      "Enabled": false
    },
    },
    "ARN": "arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Domain Layanan Amazon Elasticsearch di Panduan Pengembang Amazon Elasticsearch Service](#).

- Untuk detail API, lihat [CreateElasticsearchDomain](#) di Referensi AWS CLI Perintah.

## describe-elasticsearch-domain-config

Contoh kode berikut menunjukkan cara menggunakan `describe-elasticsearch-domain-config`.

### AWS CLI

Untuk mendapatkan detail konfigurasi domain

`describe-elasticsearch-domain-config` Contoh berikut memberikan rincian konfigurasi untuk domain tertentu, bersama dengan informasi status untuk setiap komponen domain individu.

```
aws es describe-elasticsearch-domain-config \
  --domain-name cli-example
```

Output:



```
{
  "DomainConfig": {
    "ElasticsearchVersion": {
      "Options": "7.4",
      "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
      }
    },
    "ElasticsearchClusterConfig": {
      "Options": {
        "InstanceType": "c5.large.elasticsearch",
        "InstanceCount": 1,
        "DedicatedMasterEnabled": true,
        "ZoneAwarenessEnabled": false,
        "DedicatedMasterType": "c5.large.elasticsearch",
        "DedicatedMasterCount": 3,
        "WarmEnabled": true,
        "WarmType": "ultrawarm1.medium.elasticsearch",
        "WarmCount": 2
      },
      "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
      }
    },
    "EBSOptions": {
      "Options": {
        "EBSEnabled": true,
        "VolumeType": "gp2",
        "VolumeSize": 10
      },
      "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
```

```
        "PendingDeletion": false
    }
},
"AccessPolicies": {
    "Options": [{"Version": "2012-10-17", "Statement": [{"Effect": "Allow", "Principal": {"AWS": "*"}, "Action": "es:*", "Resource": "arn:aws:es:us-east-1:123456789012:domain/cli-example/*"}]}],
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
    }
},
"SnapshotOptions": {
    "Options": {
        "AutomatedSnapshotStartHour": 0
    },
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
    }
},
"VPCOptions": {
    "Options": {},
    "Status": {
        "CreationDate": 1591210426.162,
        "UpdateDate": 1591210426.162,
        "UpdateVersion": 18,
        "State": "Active",
        "PendingDeletion": false
    }
},
"CognitoOptions": {
    "Options": {
        "Enabled": false
    },
    "Status": {
        "CreationDate": 1591210426.163,
        "UpdateDate": 1591210426.163,
```

```
        "UpdateVersion": 18,  
        "State": "Active",  
        "PendingDeletion": false  
    }  
},  
"EncryptionAtRestOptions": {  
    "Options": {  
        "Enabled": true,  
        "KmsKeyId": "arn:aws:kms:us-  
east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"  
    },  
    "Status": {  
        "CreationDate": 1589395034.946,  
        "UpdateDate": 1589395827.325,  
        "UpdateVersion": 8,  
        "State": "Active",  
        "PendingDeletion": false  
    }  
},  
"NodeToNodeEncryptionOptions": {  
    "Options": {  
        "Enabled": true  
    },  
    "Status": {  
        "CreationDate": 1589395034.946,  
        "UpdateDate": 1589395827.325,  
        "UpdateVersion": 8,  
        "State": "Active",  
        "PendingDeletion": false  
    }  
},  
"AdvancedOptions": {  
    "Options": {  
        "rest.action.multi.allow_explicit_index": "true"  
    },  
    "Status": {  
        "CreationDate": 1589395034.946,  
        "UpdateDate": 1589395827.325,  
        "UpdateVersion": 8,  
        "State": "Active",  
        "PendingDeletion": false  
    }  
},  
"LogPublishingOptions": {
```

```
    "Options": {},
    "Status": {
      "CreationDate": 1591210426.164,
      "UpdateDate": 1591210426.164,
      "UpdateVersion": 18,
      "State": "Active",
      "PendingDeletion": false
    }
  },
  "DomainEndpointOptions": {
    "Options": {
      "EnforceHTTPS": true,
      "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },
    "Status": {
      "CreationDate": 1589395034.946,
      "UpdateDate": 1589395827.325,
      "UpdateVersion": 8,
      "State": "Active",
      "PendingDeletion": false
    }
  },
  "AdvancedSecurityOptions": {
    "Options": {
      "Enabled": true,
      "InternalUserDatabaseEnabled": true
    },
    "Status": {
      "CreationDate": 1589395034.946,
      "UpdateDate": 1589827485.577,
      "UpdateVersion": 14,
      "State": "Active",
      "PendingDeletion": false
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Domain Layanan Amazon Elasticsearch di Panduan Pengembang Amazon Elasticsearch Service](#).

- Untuk detail API, lihat [DescribeElasticsearchDomainConfig](#) di Referensi AWS CLI Perintah.

## describe-elasticsearch-domain

Contoh kode berikut menunjukkan cara menggunakan `describe-elasticsearch-domain`.

### AWS CLI

Untuk mendapatkan detail untuk satu domain

`describe-elasticsearch-domain` Contoh berikut memberikan rincian konfigurasi untuk domain tertentu.

```
aws es describe-elasticsearch-domain \
  --domain-name cli-example
```

Output:

```
{
  "DomainStatus": {
    "DomainId": "123456789012/cli-example",
    "DomainName": "cli-example",
    "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example",
    "Created": true,
    "Deleted": false,
    "Endpoint": "search-cli-example-1a2a3a4a5a6a7a8a9a0a.us-
east-1.es.amazonaws.com",
    "Processing": false,
    "UpgradeProcessing": false,
    "ElasticsearchVersion": "7.4",
    "ElasticsearchClusterConfig": {
      "InstanceType": "c5.large.elasticsearch",
      "InstanceCount": 1,
      "DedicatedMasterEnabled": true,
      "ZoneAwarenessEnabled": false,
      "DedicatedMasterType": "c5.large.elasticsearch",
      "DedicatedMasterCount": 3,
      "WarmEnabled": true,
      "WarmType": "ultrawarm1.medium.elasticsearch",
      "WarmCount": 2
    },
    "EBSOptions": {
      "EBSEnabled": true,
      "VolumeType": "gp2",
      "VolumeSize": 10
    }
  }
}
```

```

    },
    "AccessPolicies": [{"Version": "2012-10-17", "Statement": [{"Effect": "Allow", "Principal": {"AWS": "*"}, "Action": "es:*", "Resource": "arn:aws:es:us-east-1:123456789012:domain/cli-example/*"}]}],
    "SnapshotOptions": {
      "AutomatedSnapshotStartHour": 0
    },
    "CognitoOptions": {
      "Enabled": false
    },
    "EncryptionAtRestOptions": {
      "Enabled": true,
      "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
    },
    "NodeToNodeEncryptionOptions": {
      "Enabled": true
    },
    "AdvancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    },
    "ServiceSoftwareOptions": {
      "CurrentVersion": "R20200522",
      "NewVersion": "",
      "UpdateAvailable": false,
      "Cancellable": false,
      "UpdateStatus": "COMPLETED",
      "Description": "There is no software update available for this domain.",
      "AutomatedUpdateDate": 0.0
    },
    "DomainEndpointOptions": {
      "EnforceHTTPS": true,
      "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },
    "AdvancedSecurityOptions": {
      "Enabled": true,
      "InternalUserDatabaseEnabled": true
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Domain Layanan Amazon Elasticsearch di Panduan Pengembang Amazon Elasticsearch Service](#).

- Untuk detail API, lihat [DescribeElasticsearchDomain](#) di Referensi AWS CLI Perintah.

## describe-elasticsearch-domains

Contoh kode berikut menunjukkan cara menggunakan `describe-elasticsearch-domains`.

### AWS CLI

Untuk mendapatkan detail untuk satu atau beberapa domain

`describe-elasticsearch-domains` Contoh berikut memberikan rincian konfigurasi untuk satu atau beberapa domain.

```
aws es describe-elasticsearch-domains \  
  --domain-names cli-example-1 cli-example-2
```

### Output:

```
{  
  "DomainStatusList": [  
    {  
      "DomainId": "123456789012/cli-example-1",  
      "DomainName": "cli-example-1",  
      "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example-1",  
      "Created": true,  
      "Deleted": false,  
      "Endpoint": "search-cli-example-1-1a2a3a4a5a6a7a8a9a0a.us-  
east-1.es.amazonaws.com",  
      "Processing": false,  
      "UpgradeProcessing": false,  
      "ElasticsearchVersion": "7.4",  
      "ElasticsearchClusterConfig": {  
        "InstanceType": "c5.large.elasticsearch",  
        "InstanceCount": 1,  
        "DedicatedMasterEnabled": true,  
        "ZoneAwarenessEnabled": false,  
        "DedicatedMasterType": "c5.large.elasticsearch",  
        "DedicatedMasterCount": 3,  
        "WarmEnabled": true,  
        "WarmType": "ultrawarm1.medium.elasticsearch",  
        "WarmCount": 2  
      },  
      "EBSOptions": {  
        "EBSEnabled": true,  
      }  
    }  
  ]  
}
```

```

        "VolumeType": "gp2",
        "VolumeSize": 10
    },
    "AccessPolicies": "{ \"Version\": \"2012-10-17\", \"Statement\": [{ \"Effect\": \"Allow\", \"Principal\": { \"AWS\": \"*\" }, \"Action\": \"es:*\", \"Resource\": \"arn:aws:es:us-east-1:123456789012:domain/cli-example-1/*\" } ] }",
    "SnapshotOptions": {
        "AutomatedSnapshotStartHour": 0
    },
    "CognitoOptions": {
        "Enabled": false
    },
    "EncryptionAtRestOptions": {
        "Enabled": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
    },
    "NodeToNodeEncryptionOptions": {
        "Enabled": true
    },
    "AdvancedOptions": {
        "rest.action.multi.allow_explicit_index": "true"
    },
    "ServiceSoftwareOptions": {
        "CurrentVersion": "R20200522",
        "NewVersion": "",
        "UpdateAvailable": false,
        "Cancellable": false,
        "UpdateStatus": "COMPLETED",
        "Description": "There is no software update available for this domain.",
        "AutomatedUpdateDate": 0.0
    },
    "DomainEndpointOptions": {
        "EnforceHTTPS": true,
        "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },
    "AdvancedSecurityOptions": {
        "Enabled": true,
        "InternalUserDatabaseEnabled": true
    }
},
{
    "DomainId": "123456789012/cli-example-2",

```



```
"DomainName": "cli-example-2",
"ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example-2",
"Created": true,
"Deleted": false,
"Processing": true,
"UpgradeProcessing": false,
"ElasticsearchVersion": "7.4",
"ElasticsearchClusterConfig": {
  "InstanceType": "r5.large.elasticsearch",
  "InstanceCount": 1,
  "DedicatedMasterEnabled": false,
  "ZoneAwarenessEnabled": false,
  "WarmEnabled": false
},
"EBSOptions": {
  "EBSEnabled": true,
  "VolumeType": "gp2",
  "VolumeSize": 10
},
"AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/cli-example-2/*\"}]}",
"SnapshotOptions": {
  "AutomatedSnapshotStartHour": 0
},
"CognitoOptions": {
  "Enabled": false
},
"EncryptionAtRestOptions": {
  "Enabled": false
},
"NodeToNodeEncryptionOptions": {
  "Enabled": false
},
"AdvancedOptions": {
  "rest.action.multi.allow_explicit_index": "true"
},
"ServiceSoftwareOptions": {
  "CurrentVersion": "",
  "NewVersion": "",
  "UpdateAvailable": false,
  "Cancellable": false,
  "UpdateStatus": "COMPLETED",
```

```

        "Description": "There is no software update available for this
domain.",
        "AutomatedUpdateDate": 0.0
    },
    "DomainEndpointOptions": {
        "EnforceHTTPS": false,
        "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },
    "AdvancedSecurityOptions": {
        "Enabled": false,
        "InternalUserDatabaseEnabled": false
    }
}
]
}

```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Domain Layanan Amazon Elasticsearch di Panduan Pengembang Amazon Elasticsearch Service](#).

- Untuk detail API, lihat [DescribeElasticsearchDomains](#) di Referensi AWS CLI Perintah.

## describe-reserved-elasticsearch-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-elasticsearch-instances`.

### AWS CLI

Untuk melihat semua instance yang dicadangkan

`describe-elasticsearch-domains` Contoh berikut memberikan ringkasan semua instance yang telah Anda pesan di suatu wilayah.

```
aws es describe-reserved-elasticsearch-instances
```

Output:

```

{
  "ReservedElasticsearchInstances": [{
    "FixedPrice": 100.0,
    "ReservedElasticsearchInstanceOfferingId":
    "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",

```

```
"ReservationName": "my-reservation",
"PaymentOption": "PARTIAL_UPFRONT",
"UsagePrice": 0.0,
"ReservedElasticsearchInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a",
"RecurringCharges": [{
  "RecurringChargeAmount": 0.603,
  "RecurringChargeFrequency": "Hourly"
}],
"State": "payment-pending",
"StartTime": 1522872571.229,
"ElasticsearchInstanceCount": 3,
"Duration": 31536000,
"ElasticsearchInstanceType": "m4.2xlarge.elasticsearch",
"CurrencyCode": "USD"
}]
}
```

Untuk informasi selengkapnya, lihat [Instans Cadangan](#) di Panduan Pengembang Amazon Elasticsearch Service.

- Untuk detail API, lihat [DescribeReservedElasticsearchInstances](#) di Referensi AWS CLI Perintah.

## list-domain-names

Contoh kode berikut menunjukkan cara menggunakan `list-domain-names`.

### AWS CLI

Untuk daftar semua domain

`list-domain-names` Contoh berikut memberikan ringkasan singkat dari semua domain di wilayah tersebut.

```
aws es list-domain-names
```

Output:

```
{
  "DomainNames": [{
    "DomainName": "cli-example-1"
  },
  {
    "DomainName": "cli-example-2"
  }
]
```

```
}  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan Mengelola Domain Layanan Amazon Elasticsearch di Panduan Pengembang Amazon Elasticsearch Service](#).

- Untuk detail API, lihat [ListDomainNames](#) di Referensi AWS CLI Perintah.

## AWS OpsWorks contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS OpsWorks.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **assign-instance**

Contoh kode berikut menunjukkan cara menggunakan `assign-instance`.

### AWS CLI

Untuk menetapkan instance terdaftar ke lapisan

Contoh berikut menetapkan instance terdaftar ke lapisan kustom.

```
aws opsworks --region us-east-1 assign-instance --instance-id 4d6d1710-ded9-42a1-  
b08e-b043ad7af1e2 --layer-ids 26cf1d32-6876-42fa-bbf1-9cad0bff938
```

Keluaran: Tidak ada.

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menetapkan Instans Terdaftar ke Lapisan di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [AssignInstance](#) di Referensi AWS CLI Perintah.

## **assign-volume**

Contoh kode berikut menunjukkan cara menggunakan `assign-volume`.

AWS CLI

Untuk menetapkan volume terdaftar ke sebuah instans

Contoh berikut menetapkan volume Amazon Elastic Block Store (Amazon EBS) terdaftar ke sebuah instans. Volume diidentifikasi oleh ID volumenya, yang merupakan GUID yang AWS OpsWorks ditetapkan saat Anda mendaftarkan volume dengan tumpukan, bukan ID volume Amazon Elastic Compute Cloud (Amazon EC2). Sebelum Anda menjalankan `assign-volume`, Anda harus terlebih dahulu menjalankan `update-volume` untuk menetapkan titik mount ke volume.

```
aws opsworks --region us-east-1 assign-volume --instance-id 4d6d1710-ded9-42a1-b08e-b043ad7af1e2 --volume-id 26cf1d32-6876-42fa-bbf1-9cad0bfff938
```

Keluaran: Tidak ada.

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menetapkan Volume Amazon EBS ke Instans di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [AssignVolume](#) di Referensi AWS CLI Perintah.

## **associate-elastic-ip**

Contoh kode berikut menunjukkan cara menggunakan `associate-elastic-ip`.

AWS CLI

Untuk mengaitkan alamat IP Elastis dengan sebuah instance

Contoh berikut mengaitkan alamat IP Elastis dengan instance tertentu.

```
aws opsworks --region us-east-1 associate-elastic-ip --instance-id
dfe18b02-5327-493d-91a4-c5c0c448927f --elastic-ip 54.148.130.96
```

Keluaran: Tidak ada.

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Manajemen Sumber Daya di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [AssociateElasticIp](#) di Referensi AWS CLI Perintah.

## **attach-elastic-load-balancer**

Contoh kode berikut menunjukkan cara menggunakan `attach-elastic-load-balancer`.

AWS CLI

Untuk melampirkan penyeimbang beban ke lapisan

Contoh berikut melampirkan penyeimbang beban, diidentifikasi dengan namanya, ke lapisan tertentu.

```
aws opsworks --region us-east-1 attach-elastic-load-balancer --elastic-load-
balancer-name Java-LB --layer-id 888c5645-09a5-4d0e-95a8-812ef1db76a4
```

Keluaran: Tidak ada.

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Elastic Load Balancing di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [AttachElasticLoadBalancer](#) di Referensi AWS CLI Perintah.

## **create-app**

Contoh kode berikut menunjukkan cara menggunakan `create-app`.

## AWS CLI

### Contoh 1: Untuk membuat aplikasi

Contoh berikut membuat aplikasi PHP bernama SimplePhpApp dari kode yang disimpan dalam repositori. GitHub Perintah menggunakan bentuk singkatan dari definisi sumber aplikasi.

```
aws opsworks create-app \  
  --region us-east-1 \  
  --stack-id f6673d70-32e6-4425-8999-265dd002fec7 \  
  --name SimplePHPApp \  
  --type php \  
  --app-source Type=git,Url=git://github.com/amazonwebservices/opsworks-demo-php-simple-app.git,Revision=version1
```

### Output:

```
{  
  "AppId": "6cf5163c-a951-444f-a8f7-3716be75f2a2"  
}
```

### Contoh 2: Untuk membuat aplikasi dengan database terlampir

Contoh berikut membuat aplikasi JSP dari kode yang disimpan dalam arsip.zip di bucket S3 publik. Ini melampirkan instance RDS DB untuk berfungsi sebagai penyimpanan data aplikasi. Sumber aplikasi dan database didefinisikan dalam file JSON terpisah yang ada di direktori tempat Anda menjalankan perintah.

```
aws opsworks create-app \  
  --region us-east-1 \  
  --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8 \  
  --name SimpleJSP \  
  --type java \  
  --app-source file://appsource.json \  
  --data-sources file://datasource.json
```

Informasi sumber aplikasi ada di `appsource.json` dan berisi yang berikut ini.

```
{  
  "Type": "archive",  
  "Url": "https://s3.amazonaws.com/opsworks-demo-assets/simplejsp.zip"
```

```
}
```

Informasi sumber database ada di `datasource.json` dan berisi yang berikut ini.

```
[
  {
    "Type": "RdsDbInstance",
    "Arn": "arn:aws:rds:us-west-2:123456789012:db:clitestdb",
    "DatabaseName": "mydb"
  }
]
```

Catatan: Untuk instance RDS DB, Anda harus terlebih dahulu menggunakan `register-rds-db-instance` untuk mendaftarkan instance dengan tumpukan. Untuk instance MySQL App Server, setel ke `Type OpsworksMySQLInstance` Contoh ini dibuat oleh AWS OpsWorks, sehingga mereka tidak harus terdaftar.

Output:

```
{
  "AppId": "26a61ead-d201-47e3-b55c-2a7c666942f8"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan Aplikasi](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateApp](#) di Referensi AWS CLI Perintah.

## create-deployment

Contoh kode berikut menunjukkan cara menggunakan `create-deployment`.

### AWS CLI

Contoh 1: Untuk menyebarkan aplikasi dan menjalankan perintah tumpukan

Contoh berikut menunjukkan cara menggunakan `create-deployment` perintah untuk menyebarkan aplikasi dan menjalankan perintah tumpukan. Perhatikan bahwa karakter quote (") dalam objek JSON yang menentukan perintah semua didahului oleh karakter escape (\). Tanpa karakter escape, perintah mungkin mengembalikan kesalahan JSON yang tidak valid.

`create-deployment` Contoh berikut menerapkan aplikasi ke tumpukan tertentu.



```
aws opsworks create-deployment \  
  --stack-id cfb7e082-ad1d-4599-8e81-de1c39ab45bf \  
  --app-id 307be5c8-d55d-47b5-bd6e-7bd417c6c7eb \  
  --command "{\"Name\":\"deploy\"}"
```

Output:

```
{  
  "DeploymentId": "5746c781-df7f-4c87-84a7-65a119880560"  
}
```

## Contoh 2: Menerapkan Aplikasi Rails dan Memigrasi Database

`create-deployment` Perintah berikut menerapkan aplikasi Ruby on Rails ke tumpukan tertentu dan memigrasikan database.

```
aws opsworks create-deployment \  
  --stack-id cfb7e082-ad1d-4599-8e81-de1c39ab45bf \  
  --app-id 307be5c8-d55d-47b5-bd6e-7bd417c6c7eb \  
  --command "{\"Name\":\"deploy\", \"Args\":{\"migrate\":[\"true\"]}]}"
```

Output:

```
{  
  "DeploymentId": "5746c781-df7f-4c87-84a7-65a119880560"  
}
```

Untuk informasi selengkapnya tentang penerapan, lihat [Menerapkan Aplikasi](#) di AWS OpsWorks Panduan Pengguna.

## Contoh 3: Jalankan Resep

`create-deployment` Perintah berikut menjalankan resep kustom `phpapp::appsetup`, pada instance dalam tumpukan tertentu.

```
aws opsworks create-deployment \  
  --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb \  
  --command "{\"Name\":\"execute_recipes\", \"Args\":{\"recipes\":[  
    [\"phpapp::appsetup\"]}]}"
```

Output:

```
{
  "DeploymentId": "5cbaa7b9-4e09-4e53-aa1b-314fbd106038"
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Stack](#) di Panduan AWS OpsWorks Pengguna.

#### Contoh 4: Instal Dependensi

`create-deployment` Perintah berikut menginstal dependensi, seperti paket atau permata Ruby, pada instance dalam tumpukan tertentu.

```
aws opsworks create-deployment \
  --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb \
  --command "{\"Name\":\"install_dependencies\"}"
```

Output:

```
{
  "DeploymentId": "aef5b255-8604-4928-81b3-9b0187f962ff"
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Stack](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateDeployment](#) di Referensi AWS CLI Perintah.

## create-instance

Contoh kode berikut menunjukkan cara menggunakan `create-instance`.

### AWS CLI

Untuk membuat sebuah instance

`create-instance` Perintah berikut membuat instance Amazon Linux m1.large bernama `myinstance1` dalam tumpukan tertentu. Instance ditugaskan ke satu lapisan.

```
aws opsworks --region us-east-1 create-instance --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --layer-ids 5c8c272a-f2d5-42e3-8245-5bf3927cb65b --hostname myinstance1 --instance-type m1.large --os "Amazon Linux"
```

Untuk menggunakan nama yang dibuat secara otomatis `get-hostname-suggestion`, panggil, yang menghasilkan nama host berdasarkan tema yang Anda tentukan saat Anda membuat tumpukan. Kemudian berikan nama itu ke argumen nama host.

Output:

```
{
  "InstanceId": "5f9adeaa-c94c-42c6-aeef-28a5376002cd"
}
```

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Menambahkan Instance ke Layer](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateInstance](#) di Referensi AWS CLI Perintah.

## **create-layer**

Contoh kode berikut menunjukkan cara menggunakan `create-layer`.

AWS CLI

Untuk membuat layer

`create-layer` Perintah berikut membuat lapisan PHP App Server bernama `MyPHPLayer` dalam tumpukan tertentu.

```
aws opsworks create-layer --region us-east-1 --stack-id
f6673d70-32e6-4425-8999-265dd002fec7 --type php-app --name MyPHPLayer --shortname
myphplayer
```

Output:

```
{
  "LayerId": "0b212672-6b4b-40e4-8a34-5a943cf2e07a"
}
```

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Cara Membuat Layer](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateLayer](#) di Referensi AWS CLI Perintah.

## create-server

Contoh kode berikut menunjukkan cara menggunakan `create-server`.

### AWS CLI

Untuk membuat server

`create-server` Contoh berikut membuat server Chef Automate baru bernama `automate-06` di wilayah default Anda. Perhatikan bahwa default digunakan untuk sebagian besar pengaturan lain, seperti jumlah cadangan yang harus disimpan, dan waktu mulai pemeliharaan dan pencadangan. Sebelum Anda menjalankan `create-server` perintah, selesaikan prasyarat dalam [Memulai untuk Chef Automate di Panduan Pengguna AWS Ossworks AWS OpsWorks for Chef Automate](#).

```
aws opsworks-cm create-server \  
  --engine "ChefAutomate" \  
  --instance-profile-arn "arn:aws:iam::012345678901:instance-profile/aws-opsworks-  
cm-ec2-role" \  
  --instance-type "t2.medium" \  
  --server-name "automate-06" \  
  --service-role-arn "arn:aws:iam::012345678901:role/aws-opsworks-cm-service-role"
```

Output:

```
{  
  "Server": {  
    "AssociatePublicIpAddress": true,  
    "BackupRetentionCount": 10,  
    "CreatedAt": 2019-12-29T13:38:47.520Z,  
    "DisableAutomatedBackup": FALSE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "ChefAutomate",  
    "EngineAttributes": [  
      {  
        "Name": "CHEF_AUTOMATE_ADMIN_PASSWORD",  
        "Value": "1Example1"  
      }  
    ],  
    "EngineModel": "Single",  
    "EngineVersion": "2019-08",
```

```
    "InstanceProfileArn": "arn:aws:iam::012345678901:instance-profile/aws-opsworks-cm-ec2-role",
    "InstanceType": "t2.medium",
    "PreferredBackupWindow": "Sun:02:00",
    "PreferredMaintenanceWindow": "00:00",
    "SecurityGroupIds": [ "sg-12345678" ],
    "ServerArn": "arn:aws:iam::012345678901:instance/automate-06-1010V4UU2WRM2",
    "ServerName": "automate-06",
    "ServiceRoleArn": "arn:aws:iam::012345678901:role/aws-opsworks-cm-service-role",
    "Status": "CREATING",
    "SubnetIds": [ "subnet-12345678" ]
  }
}
```

Untuk informasi selengkapnya, lihat [CreateServer](#) di Referensi API AWS OpsWorks untuk Chef Automate.

- Untuk detail API, lihat [CreateServer](#) di Referensi AWS CLI Perintah.

## create-stack

Contoh kode berikut menunjukkan cara menggunakan `create-stack`.

### AWS CLI

Untuk membuat tumpukan

`create-stack` Perintah berikut membuat tumpukan bernama CLI Stack.

```
aws opsworks create-stack --name "CLI Stack" --stack-region "us-east-1" --service-role-arn arn:aws:iam::123456789012:role/aws-opsworks-service-role --default-instance-profile-arn arn:aws:iam::123456789012:instance-profile/aws-opsworks-ec2-role --region us-east-1
```

Parameter `service-role-arn` dan `default-instance-profile-arn` diperlukan. Anda biasanya menggunakan yang AWS OpsWorks menciptakan untuk Anda ketika Anda membuat tumpukan pertama Anda. Untuk mendapatkan Nama Sumber Daya Amazon (ARN) untuk akun Anda, buka konsol IAM, pilih Roles di panel navigasi, pilih peran atau profil, dan pilih tab.

### Summary

Output:

```
{
  "StackId": "f6673d70-32e6-4425-8999-265dd002fec7"
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Membuat Tumpukan Baru](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateStack](#) di Referensi AWS CLI Perintah.

## create-user-profile

Contoh kode berikut menunjukkan cara menggunakan `create-user-profile`.

### AWS CLI

Untuk membuat profil pengguna

Anda mengimpor pengguna AWS Identity and Access Manager (IAM) AWS OpsWorks dengan menelepon `create-user-profile` untuk membuat profil pengguna. Contoh berikut membuat profil pengguna untuk pengguna `cli-user-test` IAM, yang diidentifikasi oleh Amazon Resource Name (ARN). Contoh ini memberikan pengguna nama pengguna SSH `myusername` dan memungkinkan manajemen diri, yang memungkinkan pengguna untuk menentukan kunci publik SSH.

```
aws opsworks --region us-east-1 create-user-profile --iam-user-arn
arn:aws:iam::123456789102:user/cli-user-test --ssh-username myusername --allow-
self-management
```

### Output:

```
{
  "IamUserArn": "arn:aws:iam::123456789102:user/cli-user-test"
}
```

Tip: Perintah ini mengimpor pengguna IAM ke dalam AWS OpsWorks, tetapi hanya dengan izin yang diberikan oleh kebijakan terlampir. Anda dapat memberikan izin per tumpukan AWS OpsWorks dengan menggunakan perintah `set-permissions`

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mengimpor Pengguna ke AWS OpsWorks dalam Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateUserProfile](#) di Referensi AWS CLI Perintah.

## **delete-app**

Contoh kode berikut menunjukkan cara menggunakan `delete-app`.

### AWS CLI

Untuk menghapus aplikasi

Contoh berikut menghapus aplikasi tertentu, yang diidentifikasi oleh ID aplikasinya. Anda dapat memperoleh ID aplikasi dengan membuka halaman detail aplikasi di AWS OpsWorks konsol atau dengan menjalankan `describe-apps` perintah.

```
aws opsworks delete-app --region us-east-1 --app-id 577943b9-2ec1-4baf-
a7bf-1d347601edc5
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Aplikasi di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DeleteApp](#) di Referensi AWS CLI Perintah.

## **delete-instance**

Contoh kode berikut menunjukkan cara menggunakan `delete-instance`.

### AWS CLI

Untuk menghapus sebuah instance

`delete-instance` Contoh berikut menghapus contoh tertentu, yang diidentifikasi oleh ID instance-nya. Anda dapat menemukan ID instance dengan membuka halaman detail instance di AWS OpsWorks konsol, atau dengan menjalankan `describe-instances` perintah.

Jika instans online, Anda harus terlebih dahulu menghentikan instance dengan menelepon `stop-instance`, dan kemudian Anda harus menunggu sampai instance berhenti. Jalankan `describe-instances` untuk memeriksa status instance.

Untuk menghapus volume Amazon EBS instans atau alamat IP Elastis, tambahkan `--delete-elastic-ip` argumen `--delete-volumes` atau masing-masing.

```
aws opsworks delete-instance \  
  --region us-east-1 \  
  --instance-id 3a21cfac-4a1f-4ce2-a921-b2cfba6f7771
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus AWS OpsWorks Instans](#) di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DeleteInstance](#) di Referensi AWS CLI Perintah.

## **delete-layer**

Contoh kode berikut menunjukkan cara menggunakan `delete-layer`.

### AWS CLI

Untuk menghapus lapisan

Contoh berikut menghapus lapisan tertentu, yang diidentifikasi oleh ID layer nya. Anda dapat memperoleh ID layer dengan membuka halaman detail layer di AWS OpsWorks konsol atau dengan menjalankan `describe-layers` perintah.

Catatan: Sebelum menghapus layer, Anda harus menggunakan `delete-instance` untuk menghapus semua instance layer.

```
aws opsworks delete-layer --region us-east-1 --layer-id a919454e-b816-4598-  
b29a-5796afb498ed
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Menghapus AWS OpsWorks Instans](#) di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DeleteLayer](#) di Referensi AWS CLI Perintah.



## delete-stack

Contoh kode berikut menunjukkan cara menggunakan `delete-stack`.

### AWS CLI

Untuk menghapus tumpukan

Contoh berikut menghapus tumpukan tertentu, yang diidentifikasi oleh ID tumpukannya. Anda dapat memperoleh ID tumpukan dengan mengklik Pengaturan Tumpukan di AWS OpsWorks konsol atau dengan menjalankan `describe-stacks` perintah.

Catatan: Sebelum menghapus lapisan, Anda harus menggunakan `delete-app`, `delete-instance`, dan `delete-layer` menghapus semua aplikasi, instance, dan lapisan tumpukan.

```
aws opsworks delete-stack --region us-east-1 --stack-id
154a9d89-7e9e-433b-8de8-617e53756c84
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mematikan Tumpukan di Panduan AWS OpsWorks Pengguna](#).

- Untuk detail API, lihat [DeleteStack](#) di Referensi AWS CLI Perintah.

## delete-user-profile

Contoh kode berikut menunjukkan cara menggunakan `delete-user-profile`.

### AWS CLI

Untuk menghapus profil pengguna dan menghapus pengguna IAM dari AWS OpsWorks

Contoh berikut menghapus profil pengguna untuk pengguna AWS Identity and Access Management (IAM) yang ditentukan, yang diidentifikasi oleh Amazon Resource Name (ARN). Operasi menghapus pengguna dari AWS OpsWorks, tetapi tidak menghapus pengguna IAM. Anda harus menggunakan konsol IAM, CLI, atau API untuk tugas itu.

```
aws opsworks --region us-east-1 delete-user-profile --iam-user-arn
arn:aws:iam::123456789102:user/cli-user-test
```

Keluaran: Tidak ada.

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mengimpor Pengguna ke AWS OpsWorks dalam Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DeleteUserProfile](#) di Referensi AWS CLI Perintah.

## **deregister-elastic-ip**

Contoh kode berikut menunjukkan cara menggunakan `deregister-elastic-ip`.

AWS CLI

Untuk membatalkan pendaftaran alamat IP Elastis dari tumpukan

Contoh berikut membatalkan pendaftaran alamat IP Elastis, yang diidentifikasi oleh alamat IP-nya, dari tumpukannya.

```
aws opsworks deregister-elastic-ip --region us-east-1 --elastic-ip 54.148.130.96
```

Keluaran: Tidak ada.

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Membatalkan pendaftaran Alamat IP Elastis di Panduan Pengguna AWS OpsWorks

- Untuk detail API, lihat [DeregisterElasticIp](#) di Referensi AWS CLI Perintah.

## **deregister-instance**

Contoh kode berikut menunjukkan cara menggunakan `deregister-instance`.

AWS CLI

Untuk membatalkan pendaftaran instance terdaftar dari tumpukan

`deregister-instance` Perintah berikut membatalkan pendaftaran instance terdaftar dari tumpukannya.

```
aws opsworks --region us-east-1 deregister-instance --instance-id 4d6d1710-  
ded9-42a1-b08e-b043ad7af1e2
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Membatalkan pendaftaran Instans Terdaftar di Panduan Pengguna.AWS OpsWorks](#)

- Untuk detail API, lihat [DeregisterInstance](#) di Referensi AWS CLI Perintah.

## deregister-rds-db-instance

Contoh kode berikut menunjukkan cara menggunakan `deregister-rds-db-instance`.

### AWS CLI

Untuk membatalkan pendaftaran instans Amazon RDS DB dari tumpukan

Contoh berikut membatalkan pendaftaran instans RDS DB, diidentifikasi oleh ARN, dari tumpukannya.

```
aws opsworks deregister-rds-db-instance --region us-east-1 --rds-db-instance-arn  
arn:aws:rds:us-west-2:123456789012:db:clitestdb
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [membatalkan pendaftaran Instans Amazon RDS di Panduan Pengguna ASW. OpsWorks](#)

ID contoh: clitestdb Nama pengguna master: cliuser Master PWD: some23! pwd DB Nama: mydb  
aws opsworks deregister-rds-db-instance --region us-east-1 -- arn:aws:rds:us-west-  
2:645732743964:db:clitestdb rds-db-instance-arn

- Untuk detail API, lihat [DeregisterRdsDbInstance](#) di Referensi AWS CLI Perintah.

## deregister-volume

Contoh kode berikut menunjukkan cara menggunakan `deregister-volume`.

## AWS CLI

Untuk membatalkan pendaftaran volume Amazon EBS

Contoh berikut membatalkan pendaftaran volume EBS dari tumpukannya. Volume diidentifikasi oleh ID volumenya, yang merupakan GUID yang AWS OpsWorks ditetapkan saat Anda mendaftarkan volume dengan tumpukan, bukan ID volume EC2.

```
aws opsworks deregister-volume --region us-east-1 --volume-id 5c48ef52-3144-4bf5-beaa-fda4deb23d4d
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat membatalkan pendaftaran Volume Amazon EBS di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DeregisterVolume](#) di Referensi AWS CLI Perintah.

## describe-apps

Contoh kode berikut menunjukkan cara menggunakan `describe-apps`.

## AWS CLI

Untuk mendeskripsikan aplikasi

`describe-apps` Perintah berikut menjelaskan aplikasi dalam tumpukan tertentu.

```
aws opsworks describe-apps \
  --stack-id 38ee91e2-abdc-4208-a107-0b7168b3cc7a \
  --region us-east-1
```

Output:

```
{
  "Apps": [
    {
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
      "AppSource": {
        "Url": "https://s3-us-west-2.amazonaws.com/opsworks-demo-assets/simplejsp.zip",
```

```
    "Type": "archive"
  },
  "Name": "SimpleJSP",
  "EnableSsl": false,
  "SslConfiguration": {},
  "AppId": "da1decc1-0dff-43ea-ad7c-bb667cd87c8b",
  "Attributes": {
    "RailsEnv": null,
    "AutoBundleOnDeploy": "true",
    "DocumentRoot": "ROOT"
  },
  "Shortname": "simplejsp",
  "Type": "other",
  "CreatedAt": "2013-08-01T21:46:54+00:00"
}
]
}
```

Untuk informasi selengkapnya, lihat Aplikasi di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeApps](#) di Referensi AWS CLI Perintah.

## describe-commands

Contoh kode berikut menunjukkan cara menggunakan `describe-commands`.

### AWS CLI

Untuk menggambarkan perintah

`describe-commands` Perintah berikut menjelaskan perintah dalam contoh tertentu.

```
aws opsworks describe-commands \
  --instance-id 8c2673b9-3fe5-420d-9cfa-78d875ee7687 \
  --region us-east-1
```

Output:

```
{
  "Commands": [
    {
      "Status": "successful",
```

```

    "CompletedAt": "2013-07-25T18:57:47+00:00",
    "InstanceId": "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
    "DeploymentId": "6ed0df4c-9ef7-4812-8dac-d54a05be1029",
    "AcknowledgedAt": "2013-07-25T18:57:41+00:00",
    "LogUrl": "https://s3.amazonaws.com/<bucket-name>/logs/008c1a91-
ec59-4d51-971d-3adff54b00cc?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
&Expires=1375394373&Signature=HkXil6UuNfxTCC37EPQAa462E1E%3D&response-cache-
control=private&response-content-encoding=gzip&response-content-type=text%2Fplain",
    "Type": "undeploy",
    "CommandId": "008c1a91-ec59-4d51-971d-3adff54b00cc",
    "CreatedAt": "2013-07-25T18:57:34+00:00",
    "ExitCode": 0
  },
  {
    "Status": "successful",
    "CompletedAt": "2013-07-25T18:55:40+00:00",
    "InstanceId": "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
    "DeploymentId": "19d3121e-d949-4ff2-9f9d-94eac087862a",
    "AcknowledgedAt": "2013-07-25T18:55:32+00:00",
    "LogUrl": "https://s3.amazonaws.com/<bucket-name>/
logs/899d3d64-0384-47b6-a586-33433aad117c?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
&Expires=1375394373&Signature=xMsJvtLuUqWmsr8s%2FAjVru0BtRs%3D&response-cache-
control=private&response-content-encoding=gzip&response-content-type=text%2Fplain",
    "Type": "deploy",
    "CommandId": "899d3d64-0384-47b6-a586-33433aad117c",
    "CreatedAt": "2013-07-25T18:55:29+00:00",
    "ExitCode": 0
  }
]
}

```

Untuk informasi selengkapnya, lihat [Peristiwa AWS OpsWorks Siklus Hidup](#) di [AWS OpsWorks Panduan Pengguna](#).

- Untuk detail API, lihat [DescribeCommands](#) di [Referensi AWS CLI Perintah](#).

## describe-deployments

Contoh kode berikut menunjukkan cara menggunakan `describe-deployments`.

### AWS CLI

Untuk menggambarkan penerapan

`describe-deployments` Perintah berikut menjelaskan penerapan dalam tumpukan tertentu.

```
aws opsworks --region us-east-1 describe-deployments --stack-id 38ee91e2-abdc-4208-a107-0b7168b3cc7a
```

Output:

```
{
  "Deployments": [
    {
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
      "Status": "successful",
      "CompletedAt": "2013-07-25T18:57:49+00:00",
      "DeploymentId": "6ed0df4c-9ef7-4812-8dac-d54a05be1029",
      "Command": {
        "Args": {},
        "Name": "undeploy"
      },
      "CreatedAt": "2013-07-25T18:57:34+00:00",
      "Duration": 15,
      "InstanceIds": [
        "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
        "9e588a25-35b2-4804-bd43-488f85ebe5b7"
      ]
    },
    {
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
      "Status": "successful",
      "CompletedAt": "2013-07-25T18:56:41+00:00",
      "IamUserArn": "arn:aws:iam::123456789012:user/someuser",
      "DeploymentId": "19d3121e-d949-4ff2-9f9d-94eac087862a",
      "Command": {
        "Args": {},
        "Name": "deploy"
      },
      "InstanceIds": [
        "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
        "9e588a25-35b2-4804-bd43-488f85ebe5b7"
      ],
      "Duration": 72,
      "CreatedAt": "2013-07-25T18:55:29+00:00"
    }
  ]
}
```

```
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menerapkan Aplikasi di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeDeployments](#) di Referensi AWS CLI Perintah.

## describe-elastic-ips

Contoh kode berikut menunjukkan cara menggunakan `describe-elastic-ips`.

### AWS CLI

Untuk menggambarkan contoh IP Elastis

`describe-elastic-ips` Perintah berikut menjelaskan alamat IP Elastis dalam contoh tertentu.

```
aws opsworks --region us-east-1 describe-elastic-ips --instance-id b62f3e04-
e9eb-436c-a91f-d9e9a396b7b0
```

### Output:

```
{
  "ElasticIps": [
    {
      "Ip": "192.0.2.0",
      "Domain": "standard",
      "Region": "us-west-2"
    }
  ]
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Contoh di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeElasticIps](#) di Referensi AWS CLI Perintah.

## describe-elastic-load-balancers

Contoh kode berikut menunjukkan cara menggunakan `describe-elastic-load-balancers`.



## AWS CLI

Untuk menggambarkan penyeimbang beban elastis tumpukan

`describe-elastic-load-balancers` Perintah berikut menjelaskan penyeimbang beban tumpukan tertentu.

```
aws opsworks --region us-west-2 describe-elastic-load-balancers --stack-id
6f4660e5-37a6-4e42-bfa0-1358ebd9c182
```

Keluaran: Tumpukan khusus ini memiliki satu penyeimbang beban.

```
{
  "ElasticLoadBalancers": [
    {
      "SubnetIds": [
        "subnet-60e4ea04",
        "subnet-66e1c110"
      ],
      "Ec2InstanceIds": [],
      "ElasticLoadBalancerName": "my-balancer",
      "Region": "us-west-2",
      "LayerId": "344973cb-bf2b-4cd0-8d93-51cd819bab04",
      "AvailabilityZones": [
        "us-west-2a",
        "us-west-2b"
      ],
      "VpcId": "vpc-b319f9d4",
      "StackId": "6f4660e5-37a6-4e42-bfa0-1358ebd9c182",
      "DnsName": "my-balancer-2094040179.us-west-2.elb.amazonaws.com"
    }
  ]
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Aplikasi di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeElasticLoadBalancers](#) di Referensi AWS CLI Perintah.

## **describe-instances**

Contoh kode berikut menunjukkan cara menggunakan `describe-instances`.

## AWS CLI

Untuk menggambarkan contoh

`describe-instances` Perintah berikut menjelaskan contoh dalam tumpukan tertentu:

```
aws opsworks --region us-east-1 describe-instances --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Output: Contoh output berikut adalah untuk tumpukan dengan dua instance. Yang pertama adalah instans EC2 terdaftar, dan yang kedua dibuat oleh AWS OpsWorks.

```
{
  "Instances": [
    {
      "StackId": "71c7ca72-55ae-4b6a-8ee1-a8dcdded3fa0f",
      "PrivateDns": "ip-10-31-39-66.us-west-2.compute.internal",
      "LayerIds": [
        "26cf1d32-6876-42fa-bbf1-9cadc0bfff938"
      ],
      "EbsOptimized": false,
      "ReportedOs": {
        "Version": "14.04",
        "Name": "ubuntu",
        "Family": "debian"
      },
      "Status": "online",
      "InstanceId": "4d6d1710-ded9-42a1-b08e-b043ad7af1e2",
      "SshKeyName": "US-West-2",
      "InfrastructureClass": "ec2",
      "RootDeviceVolumeId": "vol-d08ec6c1",
      "SubnetId": "subnet-b8de0ddd",
      "InstanceType": "t1.micro",
      "CreatedAt": "2015-02-24T20:52:49+00:00",
      "AmiId": "ami-35501205",
      "Hostname": "ip-192-0-2-0",
      "Ec2InstanceId": "i-5cd23551",
      "PublicDns": "ec2-192-0-2-0.us-west-2.compute.amazonaws.com",
      "SecurityGroupIds": [
        "sg-c4d3f0a1"
      ],
      "Architecture": "x86_64",
      "RootDeviceType": "ebs",
```

```
"InstallUpdatesOnBoot": true,
"Os": "Custom",
"VirtualizationType": "paravirtual",
"AvailabilityZone": "us-west-2a",
"PrivateIp": "10.31.39.66",
"PublicIp": "192.0.2.06",
"RegisteredBy": "arn:aws:iam::123456789102:user/AWS/OpsWorks/OpsWorks-
EC2Register-i-5cd23551"
},
{
  "StackId": "71c7ca72-55ae-4b6a-8ee1-a8dcdded3fa0f",
  "PrivateDns": "ip-10-31-39-158.us-west-2.compute.internal",
  "SshHostRsaKeyFingerprint": "69:6b:7b:8b:72:f3:ed:23:01:00:05:bc:9f:a4:60:c1",
  "LayerIds": [
    "26cf1d32-6876-42fa-bbf1-9cad0bfff938"
  ],
  "EbsOptimized": false,
  "ReportedOs": {},
  "Status": "booting",
  "InstanceId": "9b137a0d-2f5d-4cc0-9704-13da4b31fdcb",
  "SshKeyName": "US-West-2",
  "InfrastructureClass": "ec2",
  "RootDeviceVolumeId": "vol-e09dd5f1",
  "SubnetId": "subnet-b8de0ddd",
  "InstanceProfileArn": "arn:aws:iam::123456789102:instance-profile/aws-
opsworks-ec2-role",
  "InstanceType": "c3.large",
  "CreatedAt": "2015-02-24T21:29:33+00:00",
  "AmiId": "ami-9fc29baf",
  "SshHostDsaKeyFingerprint": "fc:87:95:c3:f5:e1:3b:9f:d2:06:6e:62:9a:35:27:e8",
  "Ec2InstanceId": "i-8d2dca80",
  "PublicDns": "ec2-192-0-2-1.us-west-2.compute.amazonaws.com",
  "SecurityGroupIds": [
    "sg-b022add5",
    "sg-b122add4"
  ],
  "Architecture": "x86_64",
  "RootDeviceType": "ebs",
  "InstallUpdatesOnBoot": true,
  "Os": "Amazon Linux 2014.09",
  "VirtualizationType": "paravirtual",
  "AvailabilityZone": "us-west-2a",
  "Hostname": "custom11",
  "PrivateIp": "10.31.39.158",
```

```
    "PublicIp": "192.0.2.0"  
  }  
]  
}
```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Contoh di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeInstances](#) di Referensi AWS CLI Perintah.

## describe-layers

Contoh kode berikut menunjukkan cara menggunakan `describe-layers`.

### AWS CLI

Untuk menggambarkan lapisan tumpukan

`describe-layers` Perintah berikut menjelaskan lapisan dalam tumpukan tertentu:

```
aws opsworks --region us-east-1 describe-layers --stack-id 38ee91e2-abdc-4208-  
a107-0b7168b3cc7a
```

### Output:

```
{  
  "Layers": [  
    {  
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",  
      "Type": "db-master",  
      "DefaultSecurityGroupNames": [  
        "AWS-OpsWorks-DB-Master-Server"  
      ],  
      "Name": "MySQL",  
      "Packages": [],  
      "DefaultRecipes": {  
        "Undeploy": [],  
        "Setup": [  
          "opsworks_initial_setup",  
          "ssh_host_keys",  
          "ssh_users",  
          "mysql::client",  
        ]  
      }  
    }  
  ]  
}
```

```
        "dependencies",
        "ebs",
        "opsworks_ganglia::client",
        "mysql::server",
        "dependencies",
        "deploy::mysql"
    ],
    "Configure": [
        "opsworks_ganglia::configure-client",
        "ssh_users",
        "agent_version",
        "deploy::mysql"
    ],
    "Shutdown": [
        "opsworks_shutdown::default",
        "mysql::stop"
    ],
    "Deploy": [
        "deploy::default",
        "deploy::mysql"
    ]
  ],
  "CustomRecipes": {
    "Undeploy": [],
    "Setup": [],
    "Configure": [],
    "Shutdown": [],
    "Deploy": []
  },
  "EnableAutoHealing": false,
  "LayerId": "41a20847-d594-4325-8447-171821916b73",
  "Attributes": {
    "MysqlRootPasswordUbiquitous": "true",
    "RubygemsVersion": null,
    "RailsStack": null,
    "HaproxyHealthCheckMethod": null,
    "RubyVersion": null,
    "BundlerVersion": null,
    "HaproxyStatsPassword": null,
    "PassengerVersion": null,
    "MemcachedMemory": null,
    "EnableHaproxyStats": null,
    "ManageBundler": null,
    "NodejsVersion": null,
```

```

        "HaproxyHealthCheckUrl": null,
        "MysqlRootPassword": "*****FILTERED*****",
        "GangliaPassword": null,
        "GangliaUser": null,
        "HaproxyStatsUrl": null,
        "GangliaUrl": null,
        "HaproxyStatsUser": null
    },
    "Shortname": "db-master",
    "AutoAssignElasticIps": false,
    "CustomSecurityGroupIds": [],
    "CreatedAt": "2013-07-25T18:11:19+00:00",
    "VolumeConfigurations": [
        {
            "MountPoint": "/vol/mysql",
            "Size": 10,
            "NumberOfDisks": 1
        }
    ]
},
{
    "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
    "Type": "custom",
    "DefaultSecurityGroupNames": [
        "AWS-OpsWorks-Custom-Server"
    ],
    "Name": "TomCustom",
    "Packages": [],
    "DefaultRecipes": {
        "Undeploy": [],
        "Setup": [
            "opsworks_initial_setup",
            "ssh_host_keys",
            "ssh_users",
            "mysql::client",
            "dependencies",
            "ebs",
            "opsworks_ganglia::client"
        ],
        "Configure": [
            "opsworks_ganglia::configure-client",
            "ssh_users",
            "agent_version"
        ]
    ]
},

```

```
    "Shutdown": [
      "opsworks_shutdown::default"
    ],
    "Deploy": [
      "deploy::default"
    ]
  },
  "CustomRecipes": {
    "Undeploy": [],
    "Setup": [
      "tomcat::setup"
    ],
    "Configure": [
      "tomcat::configure"
    ],
    "Shutdown": [],
    "Deploy": [
      "tomcat::deploy"
    ]
  },
  "EnableAutoHealing": true,
  "LayerId": "e6cbcd29-d223-40fc-8243-2eb213377440",
  "Attributes": {
    "MysqlRootPasswordUbiquitous": null,
    "RubygemsVersion": null,
    "RailsStack": null,
    "HaproxyHealthCheckMethod": null,
    "RubyVersion": null,
    "BundlerVersion": null,
    "HaproxyStatsPassword": null,
    "PassengerVersion": null,
    "MemcachedMemory": null,
    "EnableHaproxyStats": null,
    "ManageBundler": null,
    "NodejsVersion": null,
    "HaproxyHealthCheckUrl": null,
    "MysqlRootPassword": null,
    "GangliaPassword": null,
    "GangliaUser": null,
    "HaproxyStatsUrl": null,
    "GangliaUrl": null,
    "HaproxyStatsUser": null
  },
  "Shortname": "tomcustom",
```

```
    "AutoAssignElasticIps": false,  
    "CustomSecurityGroupIds": [],  
    "CreatedAt": "2013-07-25T18:12:53+00:00",  
    "VolumeConfigurations": []  
  }  
]  
}
```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Lapisan di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeLayers](#) di Referensi AWS CLI Perintah.

## describe-load-based-auto-scaling

Contoh kode berikut menunjukkan cara menggunakan `describe-load-based-auto-scaling`.

### AWS CLI

Untuk menggambarkan konfigurasi penskalaan berbasis beban lapisan

Contoh berikut menjelaskan konfigurasi penskalaan berbasis beban lapisan tertentu. Lapisan diidentifikasi oleh ID lapisannya, yang dapat Anda temukan di halaman detail layer atau dengan menjalankannya `describe-layers`.

```
aws opsworks describe-load-based-auto-scaling --region us-east-1 --layer-ids  
6bec29c9-c866-41a0-aba5-fa3e374ce2a1
```

Output: Lapisan contoh memiliki instance berbasis beban tunggal.

```
{  
  "LoadBasedAutoScalingConfigurations": [  
    {  
      "DownScaling": {  
        "IgnoreMetricsTime": 10,  
        "ThresholdsWaitTime": 10,  
        "InstanceCount": 1,  
        "CpuThreshold": 30.0  
      },  
      "Enable": true,  
      "UpScaling": {
```



```
    "IgnoreMetricsTime": 5,  
    "ThresholdsWaitTime": 5,  
    "InstanceCount": 1,  
    "CpuThreshold": 80.0  
  },  
  "LayerId": "6bec29c9-c866-41a0-aba5-fa3e374ce2a1"  
}  
]  
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Cara Kerja Penskalaan Berbasis Beban Otomatis di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DescribeLoadBasedAutoScaling](#) di Referensi AWS CLI Perintah.

### **describe-my-user-profile**

Contoh kode berikut menunjukkan cara menggunakan `describe-my-user-profile`.

#### AWS CLI

Untuk mendapatkan profil pengguna

Contoh berikut menunjukkan cara mendapatkan profil pengguna AWS Identity and Access Management (IAM) and Access Management yang menjalankan perintah.

```
aws opsworks --region us-east-1 describe-my-user-profile
```

Keluaran: Untuk singkatnya, sebagian besar kunci publik SSH pengguna diganti dengan elipsis (...).

```
{  
  "UserProfile": {  
    "IamUserArn": "arn:aws:iam::123456789012:user/myusername",  
    "SshPublicKey": "ssh-rsa AAAAB3NzaC1yc2EAAAABJQ...3LQ4aX9jpxQw== rsa-  
key-20141104",  
    "Name": "myusername",  
    "SshUsername": "myusername"  
  }  
}
```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mengimpor Pengguna ke AWS OpsWorks dalam Panduan AWS OpsWorks Pengguna](#).

- Untuk detail API, lihat [DescribeMyUserProfile](#) di Referensi AWS CLI Perintah.

## describe-permissions

Contoh kode berikut menunjukkan cara menggunakan `describe-permissions`.

### AWS CLI

Untuk mendapatkan tingkat izin per tumpukan AWS OpsWorks pengguna

Contoh berikut menunjukkan cara mendapatkan tingkat izin pengguna AWS Identity and Access Management (IAM) and Access Management (IAM) pada tumpukan tertentu.

```
aws opsworks --region us-east-1 describe-permissions --iam-user-arn
arn:aws:iam::123456789012:user/cli-user-test --stack-id d72553d4-8727-448c-9b00-
f024f0ba1b06
```

### Output:

```
{
  "Permissions": [
    {
      "StackId": "d72553d4-8727-448c-9b00-f024f0ba1b06",
      "IamUserArn": "arn:aws:iam::123456789012:user/cli-user-test",
      "Level": "manage",
      "AllowSudo": true,
      "AllowSsh": true
    }
  ]
}
```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Memberikan Tingkat Izin Per-Tumpukan di Panduan Pengguna.AWS OpsWorks](#)

- Untuk detail API, lihat [DescribePermissions](#) di Referensi AWS CLI Perintah.

## describe-raid-arrays

Contoh kode berikut menunjukkan cara menggunakan `describe-raid-arrays`.

### AWS CLI

Untuk menggambarkan array RAID

Contoh berikut menjelaskan array RAID yang dilampirkan ke instance dalam tumpukan tertentu.

```
aws opsworks --region us-east-1 describe-raid-arrays --stack-id
d72553d4-8727-448c-9b00-f024f0ba1b06
```

Keluaran: Berikut ini adalah output untuk tumpukan dengan satu array RAID.

```
{
  "RaidArrays": [
    {
      "StackId": "d72553d4-8727-448c-9b00-f024f0ba1b06",
      "AvailabilityZone": "us-west-2a",
      "Name": "Created for php-app1",
      "NumberOfDisks": 2,
      "InstanceId": "9f14adbc-ced5-43b6-bf01-e7d0db6cf2f7",
      "RaidLevel": 0,
      "VolumeType": "standard",
      "RaidArrayId": "f2d4e470-5972-4676-b1b8-bae41ec3e51c",
      "Device": "/dev/md0",
      "MountPoint": "/mnt/workspace",
      "CreatedAt": "2015-02-26T23:53:09+00:00",
      "Size": 100
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Volume EBS di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeRaidArrays](#) di Referensi AWS CLI Perintah.

## describe-rds-db-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-rds-db-instances`.

## AWS CLI

Untuk menggambarkan instans Amazon RDS yang terdaftar di tumpukan

Contoh berikut menjelaskan instans Amazon RDS yang terdaftar dengan tumpukan tertentu.

```
aws opsworks --region us-east-1 describe-rds-db-instances --stack-id
d72553d4-8727-448c-9b00-f024f0ba1b06
```

Keluaran: Berikut ini adalah output untuk tumpukan dengan satu instance RDS terdaftar.

```
{
  "RdsDbInstances": [
    {
      "Engine": "mysql",
      "StackId": "d72553d4-8727-448c-9b00-f024f0ba1b06",
      "MissingOnRds": false,
      "Region": "us-west-2",
      "RdsDbInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:clitestdb",
      "DbPassword": "*****FILTERED*****",
      "Address": "clitestdb.cd1qlk5uwd0k.us-west-2.rds.amazonaws.com",
      "DbUser": "cliuser",
      "DbInstanceIdentifier": "clitestdb"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Manajemen Sumber Daya di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeRdsDbInstances](#) di Referensi AWS CLI Perintah.

## describe-stack-provisioning-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-provisioning-parameters`.

### AWS CLI

Untuk mengembalikan parameter penyediaan untuk tumpukan

`describe-stack-provisioning-parameters` Contoh berikut mengembalikan parameter penyediaan untuk tumpukan tertentu. Parameter penyediaan mencakup pengaturan seperti lokasi

instalasi agen dan kunci publik yang OpsWorks digunakan untuk mengelola agen pada instance dalam tumpukan.

```
aws opsworks describe-stack-provisioning-parameters \
  --stack-id 62744d97-6faf-4ecb-969b-a086fEXAMPLE
```

Output:

```
{
  "AgentInstallerUrl": "https://opsworks-instance-agent-us-
west-2.s3.amazonaws.com/ID_number/opsworks-agent-installer.tgz",
  "Parameters": {
    "agent_installer_base_url": "https://opsworks-instance-agent-us-
west-2.s3.amazonaws.com",
    "agent_installer_tgz": "opsworks-agent-installer.tgz",
    "assets_download_bucket": "opsworks-instance-assets-us-
west-2.s3.amazonaws.com",
    "charlie_public_key": "-----BEGIN PUBLIC KEY-----PUBLIC_KEY_EXAMPLE\n-----
END PUBLIC KEY-----",
    "instance_service_endpoint": "opsworks-instance-service.us-
west-2.amazonaws.com",
    "instance_service_port": "443",
    "instance_service_region": "us-west-2",
    "instance_service_ssl_verify_peer": "true",
    "instance_service_use_ssl": "true",
    "ops_works_endpoint": "opsworks.us-west-2.amazonaws.com",
    "ops_works_port": "443",
    "ops_works_region": "us-west-2",
    "ops_works_ssl_verify_peer": "true",
    "ops_works_use_ssl": "true",
    "verbose": "false",
    "wait_between_runs": "30"
  }
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Stack](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeStackProvisioningParameters](#) di Referensi AWS CLI Perintah.

## describe-stack-summary

Contoh kode berikut menunjukkan cara menggunakan `describe-stack-summary`.

### AWS CLI

Untuk menggambarkan konfigurasi tumpukan

`describe-stack-summary` Perintah berikut mengembalikan ringkasan konfigurasi stack yang ditentukan.

```
aws opsworks --region us-east-1 describe-stack-summary --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Output:

```
{
  "StackSummary": {
    "StackId": "8c428b08-a1a1-46ce-a5f8-feddc43771b8",
    "InstancesCount": {
      "Booting": 1
    },
    "Name": "CLITest",
    "AppsCount": 1,
    "LayersCount": 1,
    "Arn": "arn:aws:opsworks:us-west-2:123456789012:stack/8c428b08-a1a1-46ce-a5f8-feddc43771b8/"
  }
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Tumpukan di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeStackSummary](#) di Referensi AWS CLI Perintah.

## describe-stacks

Contoh kode berikut menunjukkan cara menggunakan `describe-stacks`.

### AWS CLI

Untuk menggambarkan tumpukan

`describe-stacks` Perintah berikut menjelaskan tumpukan akun.

```
aws opsworks --region us-east-1 describe-stacks
```

Output:

```
{
  "Stacks": [
    {
      "ServiceRoleArn": "arn:aws:iam::444455556666:role/aws-opsworks-service-role",
      "StackId": "aeb7523e-7c8b-49d4-b866-03aae9d4fbcf",
      "DefaultRootDeviceType": "instance-store",
      "Name": "TomStack-sd",
      "ConfigurationManager": {
        "Version": "11.4",
        "Name": "Chef"
      },
      "UseCustomCookbooks": true,
      "CustomJson": "{\n  \"tomcat\": {\n    \"base_version\": 7,\n    \"java_opts\n\": \"-Djava.awt.headless=true -Xmx256m\"\n  },\n  \"datasources\": {\n    \"ROOT\":\n  \"jdbc/mydb\"\n  }\n}",
      "Region": "us-east-1",
      "DefaultInstanceProfileArn": "arn:aws:iam::444455556666:instance-profile/aws-opsworks-ec2-role",
      "CustomCookbooksSource": {
        "Url": "git://github.com/example-repo/tomcustom.git",
        "Type": "git"
      },
      "DefaultAvailabilityZone": "us-east-1a",
      "HostnameTheme": "Layer_Dependent",
      "Attributes": {
        "Color": "rgb(45, 114, 184)"
      },
      "DefaultOs": "Amazon Linux",
      "CreatedAt": "2013-08-01T22:53:42+00:00"
    },
    {
      "ServiceRoleArn": "arn:aws:iam::444455556666:role/aws-opsworks-service-role",
      "StackId": "40738975-da59-4c5b-9789-3e422f2cf099",
      "DefaultRootDeviceType": "instance-store",
      "Name": "MyStack",
      "ConfigurationManager": {
        "Version": "11.4",
```

```
    "Name": "Chef"
  },
  "UseCustomCookbooks": false,
  "Region": "us-east-1",
  "DefaultInstanceProfileArn": "arn:aws:iam::444455556666:instance-profile/aws-opsworks-ec2-role",
  "CustomCookbooksSource": {},
  "DefaultAvailabilityZone": "us-east-1a",
  "HostnameTheme": "Layer_Dependent",
  "Attributes": {
    "Color": "rgb(45, 114, 184)"
  },
  "DefaultOs": "Amazon Linux",
  "CreatedAt": "2013-10-25T19:24:30+00:00"
}
]
}
```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Tumpukan di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeStacks](#) di Referensi AWS CLI Perintah.

## describe-timebased-auto-scaling

Contoh kode berikut menunjukkan cara menggunakan `describe-timebased-auto-scaling`.

### AWS CLI

Untuk menggambarkan konfigurasi penskalaan berbasis waktu dari sebuah instance

Contoh berikut menjelaskan konfigurasi penskalaan berbasis waktu instance tertentu. Instance diidentifikasi oleh ID instance-nya, yang dapat Anda temukan di halaman detail instance atau dengan `describe-instances` menjalankannya.

```
aws opsworks describe-time-based-auto-scaling --region us-east-1 --instance-ids
701f2ffe-5d8e-4187-b140-77b75f55de8d
```

Output: Contoh ini memiliki instance berbasis waktu tunggal.

```
{
```



```
"TimeBasedAutoScalingConfigurations": [  
  {  
    "InstanceId": "701f2ffe-5d8e-4187-b140-77b75f55de8d",  
    "AutoScalingSchedule": {  
      "Monday": {  
        "11": "on",  
        "10": "on",  
        "13": "on",  
        "12": "on"  
      },  
      "Tuesday": {  
        "11": "on",  
        "10": "on",  
        "13": "on",  
        "12": "on"  
      }  
    }  
  }  
]
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Cara Kerja Penskalaan Berbasis Waktu Otomatis di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DescribeTimebasedAutoScaling](#) di Referensi AWS CLI Perintah.

## describe-user-profiles

Contoh kode berikut menunjukkan cara menggunakan `describe-user-profiles`.

### AWS CLI

Untuk menggambarkan profil pengguna

`describe-user-profiles` Perintah berikut menjelaskan profil pengguna akun.

```
aws opsworks --region us-east-1 describe-user-profiles
```

Output:

```
{
```

```
"UserProfiles": [  
  {  
    "IamUserArn": "arn:aws:iam::123456789012:user/someuser",  
    "SshPublicKey": "ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEak0uP7i80q3Cko...",  
    "AllowSelfManagement": true,  
    "Name": "someuser",  
    "SshUsername": "someuser"  
  },  
  {  
    "IamUserArn": "arn:aws:iam::123456789012:user/cli-user-test",  
    "AllowSelfManagement": true,  
    "Name": "cli-user-test",  
    "SshUsername": "myusername"  
  }  
]  
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mengelola AWS OpsWorks Pengguna](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeUserProfiles](#) di Referensi AWS CLI Perintah.

## describe-volumes

Contoh kode berikut menunjukkan cara menggunakan `describe-volumes`.

### AWS CLI

Untuk menggambarkan volume tumpukan

Contoh berikut menjelaskan volume EBS stack.

```
aws opsworks --region us-east-1 describe-volumes --stack-id 8c428b08-a1a1-46ce-a5f8-  
feddc43771b8
```

Output:

```
{  
  "Volumes": [  
    {
```

```
"Status": "in-use",
"AvailabilityZone": "us-west-2a",
"Name": "CLITest",
"InstanceId": "dfe18b02-5327-493d-91a4-c5c0c448927f",
"VolumeType": "standard",
"VolumeId": "56b66fbd-e1a1-4aff-9227-70f77118d4c5",
"Device": "/dev/sdi",
"Ec2VolumeId": "vol-295c1638",
"MountPoint": "/mnt/myvolume",
"Size": 1
}
]
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Manajemen Sumber Daya di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeVolumes](#) di Referensi AWS CLI Perintah.

## detach-elastic-load-balancer

Contoh kode berikut menunjukkan cara menggunakan `detach-elastic-load-balancer`.

### AWS CLI

Untuk melepaskan penyeimbang beban dari lapisannya

Contoh berikut melepaskan penyeimbang beban, diidentifikasi dengan namanya, dari lapisannya.

```
aws opsworks --region us-east-1 detach-elastic-load-balancer --elastic-load-balancer-name Java-LB --layer-id 888c5645-09a5-4d0e-95a8-812ef1db76a4
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Elastic Load Balancing di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DetachElasticLoadBalancer](#) di Referensi AWS CLI Perintah.

## **disassociate-elastic-ip**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-elastic-ip`.

### AWS CLI

Untuk memisahkan alamat IP Elastis dari sebuah instance

Contoh berikut memisahkan alamat IP Elastis dari instance tertentu.

```
aws opsworks --region us-east-1 disassociate-elastic-ip --elastic-ip 54.148.130.96
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Manajemen Sumber Daya di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DisassociateElasticIp](#) di Referensi AWS CLI Perintah.

## **get-hostname-suggestion**

Contoh kode berikut menunjukkan cara menggunakan `get-hostname-suggestion`.

### AWS CLI

Untuk mendapatkan nama host berikutnya untuk sebuah layer

Contoh berikut mendapatkan nama host yang dihasilkan berikutnya untuk lapisan tertentu. Lapisan yang digunakan untuk contoh ini adalah lapisan Java Application Server dengan satu contoh. Tema hostname stack adalah default, `Layer_Dependent`.

```
aws opsworks --region us-east-1 get-hostname-suggestion --layer-id  
888c5645-09a5-4d0e-95a8-812ef1db76a4
```

Output:

```
{  
  "Hostname": "java-app2",  
  "LayerId": "888c5645-09a5-4d0e-95a8-812ef1db76a4"
```

```
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Membuat Tumpukan Baru](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [GetHostnameSuggestion](#) di Referensi AWS CLI Perintah.

## **reboot-instance**

Contoh kode berikut menunjukkan cara menggunakan `reboot-instance`.

### AWS CLI

Untuk me-reboot sebuah instance

Contoh berikut reboot sebuah instance.

```
aws opsworks --region us-east-1 reboot-instance --instance-id
dfe18b02-5327-493d-91a4-c5c0c448927f
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mem-boot Ulang Instance](#) di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [RebootInstance](#) di Referensi AWS CLI Perintah.

## **register-elastic-ip**

Contoh kode berikut menunjukkan cara menggunakan `register-elastic-ip`.

### AWS CLI

Untuk mendaftarkan alamat IP Elastis dengan tumpukan

Contoh berikut mendaftarkan alamat IP Elastis, diidentifikasi oleh alamat IP-nya, dengan tumpukan tertentu.

Catatan: Alamat IP Elastis harus berada di wilayah yang sama dengan tumpukan.

```
aws opsworks register-elastic-ip --region us-east-1 --stack-id
d72553d4-8727-448c-9b00-f024f0ba1b06 --elastic-ip 54.148.130.96
```

## Keluaran

```
{
  "ElasticIp": "54.148.130.96"
}
```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mendaftarkan Alamat IP Elastis dengan Tumpukan di Panduan OpsWorks Pengguna.

- Untuk detail API, lihat [RegisterElasticIp](#) di Referensi AWS CLI Perintah.

## register-rds-db-instance

Contoh kode berikut menunjukkan cara menggunakan `register-rds-db-instance`.

### AWS CLI

Untuk mendaftarkan instans Amazon RDS dengan tumpukan

Contoh berikut mendaftarkan instans Amazon RDS DB, diidentifikasi oleh Amazon Resource Name (ARN), dengan tumpukan tertentu. Ini juga menentukan nama pengguna dan kata sandi master instance. Perhatikan bahwa AWS OpsWorks tidak memvalidasi salah satu dari nilai-nilai ini. Jika salah satu salah, aplikasi Anda tidak akan dapat terhubung ke database.

```
aws opsworks register-rds-db-instance --region us-east-1 --stack-id
d72553d4-8727-448c-9b00-f024f0ba1b06 --rds-db-instance-arn arn:aws:rds:us-
west-2:123456789012:db:clitestdb --db-user cliuser --db-password some23!pwd
```

Keluaran: Tidak ada.

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mendaftarkan Instans Amazon RDS dengan Tumpukan di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [RegisterRdsDbInstance](#) di Referensi AWS CLI Perintah.

## register-volume

Contoh kode berikut menunjukkan cara menggunakan `register-volume`.

### AWS CLI

Untuk mendaftarkan volume Amazon EBS dengan tumpukan

Contoh berikut mendaftarkan volume Amazon EBS, diidentifikasi oleh ID volumenya, dengan tumpukan tertentu.

```
aws opsworks register-volume --region us-east-1 --stack-id d72553d4-8727-448c-9b00-f024f0ba1b06 --ec-2-volume-id vol-295c1638
```

Output:

```
{
  "VolumeId": "ee08039c-7cb7-469f-be10-40fb7f0c05e8"
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mendaftarkan Volume Amazon EBS dengan Tumpukan](#) di [Panduan AWS OpsWorks Pengguna](#).

- Untuk detail API, lihat [RegisterVolume](#) di Referensi AWS CLI Perintah.

## register

Contoh kode berikut menunjukkan cara menggunakan `register`.

### AWS CLI

Untuk mendaftarkan instance dengan tumpukan

Contoh berikut menunjukkan berbagai cara untuk mendaftarkan instance dengan tumpukan yang dibuat di luar AWS Opsworks. Anda dapat menjalankan `register` dari instance yang akan didaftarkan, atau dari workstation terpisah. Untuk informasi selengkapnya, lihat [Mendaftarkan Amazon EC2 dan Instans Lokal](#) di [Panduan Pengguna AWS OpsWorks](#)

Catatan: Untuk singkatnya, contoh menghilangkan argumen. `region`

Untuk mendaftarkan instans Amazon EC2

Untuk menunjukkan bahwa Anda mendaftarkan instans EC2, atur `--infrastructure-class` argumen ke `ec2`.

Contoh berikut mendaftarkan instance EC2 dengan tumpukan tertentu dari workstation terpisah. Instans diidentifikasi oleh ID EC2-nya, `i-12345678`. Contoh menggunakan nama pengguna SSH default workstation dan mencoba masuk ke instance menggunakan teknik otentikasi yang tidak memerlukan kata sandi, seperti kunci SSH pribadi default. Jika gagal, `register` kueri untuk kata sandi.

```
aws opsworks register --infrastructure-class=ec2 --stack-id 935450cc-61e0-4b03-
a3e0-160ac817d2bb i-12345678
```

Contoh berikut mendaftarkan instance EC2 dengan tumpukan khusus dari workstation terpisah. Ini menggunakan `--ssh-private-key` argumen `--ssh-username` dan untuk secara eksplisit menentukan nama pengguna SSH dan file kunci pribadi yang digunakan perintah untuk masuk ke instance. `ec2-user` adalah nama pengguna standar untuk instans Amazon Linux. Gunakan `ubuntu` untuk instance Ubuntu.

```
aws opsworks register --infrastructure-class=ec2 --stack-id 935450cc-61e0-4b03-
a3e0-160ac817d2bb --ssh-username ec2-user --ssh-private-key ssh_private_key
i-12345678
```

Contoh berikut mendaftarkan instance EC2 yang menjalankan perintah. `register` Masuk ke instance dengan SSH dan jalankan `register` dengan `--local` argumen alih-alih ID instance atau nama host.

```
aws opsworks register --infrastructure-class ec2 --stack-id 935450cc-61e0-4b03-
a3e0-160ac817d2bb --local
```

Untuk mendaftarkan instans lokal

Untuk menunjukkan bahwa Anda mendaftarkan instans lokal, setel `--infrastructure-class` argumen ke `on-premises`.

Contoh berikut mendaftarkan instance lokal yang ada dengan tumpukan tertentu dari workstation terpisah. Instance diidentifikasi oleh alamat IP-nya, `192.0.2.3`. Contoh menggunakan nama pengguna SSH default workstation dan mencoba masuk ke instance menggunakan teknik otentikasi yang tidak memerlukan kata sandi, seperti kunci SSH pribadi default. Jika gagal, `register` kueri untuk kata sandi.



```
aws opsworks register --infrastructure-class on-premises --stack-id
935450cc-61e0-4b03-a3e0-160ac817d2bb 192.0.2.3
```

Contoh berikut mendaftarkan instance lokal dengan tumpukan tertentu dari workstation terpisah. Instance diidentifikasi dengan nama hostnya, `host1`. `--override-...` Argumen langsung AWS OpsWorks untuk ditampilkan `webserver1` sebagai nama host dan `192.0.2.3` dan `10.0.0.2` sebagai alamat IP publik dan pribadi instance, masing-masing.

```
aws opsworks register --infrastructure-class on-premises --stack-id
935450cc-61e0-4b03-a3e0-160ac817d2bb --override-hostname webserver1 --override-
public-ip 192.0.2.3 --override-private-ip 10.0.0.2 host1
```

Contoh berikut mendaftarkan instance lokal dengan tumpukan tertentu dari workstation terpisah. Instance diidentifikasi oleh alamat IP-nya. `register` masuk ke instance menggunakan nama pengguna SSH dan file kunci pribadi yang ditentukan.

```
aws opsworks register --infrastructure-class on-premises --stack-id
935450cc-61e0-4b03-a3e0-160ac817d2bb --ssh-username admin --ssh-private-key
ssh_private_key 192.0.2.3
```

Contoh berikut mendaftarkan instance lokal yang ada dengan tumpukan tertentu dari workstation terpisah. Perintah masuk ke instance menggunakan string perintah SSH kustom yang menentukan kata sandi SSH dan alamat IP instans.

```
aws opsworks register --infrastructure-class on-premises --stack-id
935450cc-61e0-4b03-a3e0-160ac817d2bb --override-ssh "sshpass -p 'mypassword' ssh
your-user@192.0.2.3"
```

Contoh berikut mendaftarkan instance lokal yang menjalankan perintah. `register` Masuk ke instance dengan SSH dan jalankan `register` dengan `--local` argumen alih-alih ID instance atau nama host.

```
aws opsworks register --infrastructure-class on-premises --stack-id
935450cc-61e0-4b03-a3e0-160ac817d2bb --local
```

Keluaran: Berikut ini adalah output khas untuk mendaftarkan instans EC2.

```
Warning: Permanently added '52.11.41.206' (ECDSA) to the list of known hosts.
```

```

% Total      % Received % Xferd  Average Speed   Time    Time       Time  Current
           Dload  Upload  Total      Spent      Left     Speed
100 6403k  100 6403k    0      0  2121k      0  0:00:03  0:00:03  --:--:-- 2121k
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Initializing AWS OpsWorks
environment
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Running on Ubuntu
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Checking if OS is supported
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Running on supported OS
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Setup motd
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Executing: ln -sf --backup /etc/
motd.opsworks-static /etc/motd
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Enabling multiverse repositories
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Customizing APT environment
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Installing system packages
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Executing: dpkg --configure -a
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Executing with retry: apt-get
update
[Tue, 24 Feb 2015 20:49:13 +0000] opsworks-init: Executing: apt-get install -y ruby
ruby-dev libicu-dev libssl-dev libxslt-dev libxml2-dev libyaml-dev monit
[Tue, 24 Feb 2015 20:50:13 +0000] opsworks-init: Using assets bucket from
environment: 'opsworks-instance-assets-us-east-1.s3.amazonaws.com'.
[Tue, 24 Feb 2015 20:50:13 +0000] opsworks-init: Installing Ruby for the agent
[Tue, 24 Feb 2015 20:50:13 +0000] opsworks-init: Executing: /tmp/opsworks-
agent-installer.YgGq8wF3UUre6yDy/opsworks-agent-installer/opsworks-agent/bin/
installer_wrapper.sh -r -R opsworks-instance-assets-us-east-1.s3.amazonaws.com
[Tue, 24 Feb 2015 20:50:44 +0000] opsworks-init: Starting the installer
Instance successfully registered. Instance ID: 4d6d1710-ded9-42a1-b08e-b043ad7af1e2
Connection to 52.11.41.206 closed.

```

## Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mendaftarkan Instance dengan AWS OpsWorks Stack di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [Mendaftar](#) di Referensi AWS CLI Perintah.

## set-load-based-auto-scaling

Contoh kode berikut menunjukkan cara menggunakan `set-load-based-auto-scaling`.

### AWS CLI

Untuk mengatur konfigurasi penskalaan berbasis beban untuk lapisan

Contoh berikut memungkinkan penskalaan berbasis beban untuk lapisan tertentu dan menetapkan konfigurasi untuk lapisan itu. Anda harus menggunakan `create-instance` untuk menambahkan instance berbasis beban ke layer.

```
aws opsworks --region us-east-1 set-load-based-auto-scaling --layer-id
523569ae-2faf-47ac-b39e-f4c4b381f36d --enable --up-scaling file://upscale.json --
down-scaling file://downscale.json
```

Contoh menempatkan pengaturan ambang upscaling dalam file terpisah di direktori kerja bernama `upscale.json`, yang berisi berikut ini.

```
{
  "InstanceCount": 2,
  "ThresholdsWaitTime": 3,
  "IgnoreMetricsTime": 3,
  "CpuThreshold": 85,
  "MemoryThreshold": 85,
  "LoadThreshold": 85
}
```

Contoh ini menempatkan pengaturan ambang downscaling dalam file terpisah di direktori kerja bernama `downscale.json`, yang berisi berikut ini.

```
{
  "InstanceCount": 2,
  "ThresholdsWaitTime": 3,
  "IgnoreMetricsTime": 3,
  "CpuThreshold": 35,
  "MemoryThreshold": 30,
  "LoadThreshold": 30
}
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Menggunakan Penskalaan Berbasis Beban Otomatis di AWS OpsWorks Panduan Pengguna](#).

- Untuk detail API, lihat [SetLoadBasedAutoScaling](#) di Referensi AWS CLI Perintah.

## set-permission

Contoh kode berikut menunjukkan cara menggunakan `set-permission`.

### AWS CLI

Untuk memberikan tingkat izin per tumpukan AWS OpsWorks

Saat Anda mengimpor pengguna AWS Identity and Access Management (IAM) and Access Management (IAM) AWS OpsWorks dengan menelepon `create-user-profile`, pengguna hanya memiliki izin yang diberikan oleh kebijakan IAM terlampir. Anda dapat memberikan AWS OpsWorks izin dengan mengubah kebijakan pengguna. Namun, seringkali lebih mudah untuk mengimpor pengguna dan kemudian menggunakan `set-permission` perintah untuk memberikan pengguna salah satu tingkat izin standar untuk setiap tumpukan yang pengguna akan memerlukan akses.

Contoh berikut memberikan izin untuk tumpukan tertentu untuk pengguna, yang diidentifikasi oleh Amazon Resource Name (ARN). Contoh ini memberi pengguna tingkat izin Kelola, dengan hak istimewa `sudo` dan `SSH` pada instance tumpukan.

```
aws opsworks set-permission --region us-east-1 --stack-id 71c7ca72-55ae-4b6a-8ee1-
a8dcdded3fa0f --level manage --iam-user-arn arn:aws:iam::123456789102:user/cli-user-
test --allow-ssh --allow-sudo
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Memberikan Izin Per-Tumpukan AWS OpsWorks Pengguna](#) di [Panduan Pengguna AWS OpsWorks](#).

- Untuk detail API, lihat [SetPermission](#) di Referensi AWS CLI Perintah.

## set-time-based-auto-scaling

Contoh kode berikut menunjukkan cara menggunakan `set-time-based-auto-scaling`.

### AWS CLI

Untuk mengatur konfigurasi penskalaan berbasis waktu untuk lapisan

Contoh berikut menetapkan konfigurasi berbasis waktu untuk instance tertentu. Anda harus terlebih dahulu menggunakan `create-instance` untuk menambahkan instance ke layer.

```
aws opsworks --region us-east-1 set-time-based-auto-scaling --instance-id
69b6237c-08c0-4edb-a6af-78f3d01cedf2 --auto-scaling-schedule file://schedule.json
```

Contoh menempatkan jadwal dalam file terpisah di direktori kerja bernama `schedule.json`. Untuk contoh ini, instance menyala selama beberapa jam sekitar tengah hari UTC (Coordinated Universal Time) pada hari Senin dan Selasa.

```
{
  "Monday": {
    "10": "on",
    "11": "on",
    "12": "on",
    "13": "on"
  },
  "Tuesday": {
    "10": "on",
    "11": "on",
    "12": "on",
    "13": "on"
  }
}
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Menggunakan Penskalaan Berbasis Waktu Otomatis di AWS OpsWorks Panduan Pengguna](#).

- Untuk detail API, lihat [SetTimeBasedAutoScaling](#) di Referensi AWS CLI Perintah.

## start-instance

Contoh kode berikut menunjukkan cara menggunakan `start-instance`.

### AWS CLI

Untuk memulai sebuah instance

**start-instance** Perintah berikut memulai instance 24/7 yang ditentukan.

```
aws opsworks start-instance --instance-id f705ee48-9000-4890-8bd3-20eb05825aaf
```

Keluaran: Tidak ada. Gunakan `describe-instance` untuk memeriksa status instans.

Tip Anda dapat memulai setiap instance offline dalam tumpukan dengan satu perintah dengan memanggil `start-stack`.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Memulai, Menghentikan, dan Mem-boot Ulang Instans 24/7 Secara Manual di Panduan Pengguna.AWS OpsWorks

- Untuk detail API, lihat [StartInstance](#) di Referensi AWS CLI Perintah.

## start-stack

Contoh kode berikut menunjukkan cara menggunakan `start-stack`.

### AWS CLI

Untuk memulai instance tumpukan

Contoh berikut memulai semua instance 24/7 stack. Untuk memulai contoh tertentu, gunakan `start-instance`.

```
aws opsworks --region us-east-1 start-stack --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Memulai Instance di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [StartStack](#) di Referensi AWS CLI Perintah.

## stop-instance

Contoh kode berikut menunjukkan cara menggunakan `stop-instance`.

## AWS CLI

Untuk menghentikan sebuah instance

Contoh berikut menghentikan instance tertentu, yang diidentifikasi oleh ID instance-nya. Anda dapat memperoleh ID instance dengan membuka halaman detail instance di AWS OpsWorks konsol atau dengan menjalankan `describe-instances` perintah.

```
aws opsworks stop-instance --region us-east-1 --instance-id 3a21cfac-4a1f-4ce2-a921-
b2cfba6f7771
```

Anda dapat memulai ulang instance yang dihentikan dengan memanggil `start-instance` atau dengan menghapus instance dengan menelepon `delete-instance`.

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menghentikan Instance di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [StopInstance](#) di Referensi AWS CLI Perintah.

## stop-stack

Contoh kode berikut menunjukkan cara menggunakan `stop-stack`.

## AWS CLI

Untuk menghentikan instance tumpukan

Contoh berikut menghentikan semua instance 24/7 stack. Untuk menghentikan contoh tertentu, gunakan `stop-instance`.

```
aws opsworks --region us-east-1 stop-stack --stack-id 8c428b08-a1a1-46ce-a5f8-
feddc43771b8
```

Keluaran: Tidak ada output.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Menghentikan Instance](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [StopStack](#) di Referensi AWS CLI Perintah.

## **unassign-instance**

Contoh kode berikut menunjukkan cara menggunakan `unassign-instance`.

### AWS CLI

Untuk membatalkan penetapan instance terdaftar dari lapisannya

`unassign-instance` Perintah berikut membatalkan penugasan instance dari lapisan terlampirnya.

```
aws opsworks --region us-east-1 unassign-instance --instance-id 4d6d1710-ded9-42a1-b08e-b043ad7af1e2
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Membatalkan Penetapan Instans Terdaftar di AWS OpsWorks](#) Panduan Pengguna.

- Untuk detail API, lihat [UnassignInstance](#) di Referensi AWS CLI Perintah.

## **unassign-volume**

Contoh kode berikut menunjukkan cara menggunakan `unassign-volume`.

### AWS CLI

Untuk membatalkan penugasan volume dari instance-nya

Contoh berikut membatalkan penetapan volume Amazon Elastic Block Store (Amazon EBS) terdaftar dari instancenya. Volume diidentifikasi oleh ID volumenya, yang merupakan GUID yang AWS OpsWorks ditetapkan saat Anda mendaftarkan volume dengan tumpukan, bukan ID volume Amazon Elastic Compute Cloud (Amazon EC2).



```
aws opsworks --region us-east-1 unassign-volume --volume-id 8430177d-52b7-4948-9c62-
e195af4703df
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Membatalkan Penetapan Volume Amazon EBS di Panduan Pengguna.AWS OpsWorks](#)

- Untuk detail API, lihat [UnassignVolume](#) di Referensi AWS CLI Perintah.

## update-app

Contoh kode berikut menunjukkan cara menggunakan `update-app`.

### AWS CLI

Untuk memperbarui aplikasi

Contoh berikut memperbarui aplikasi tertentu untuk mengubah namanya.

```
aws opsworks --region us-east-1 update-app --app-id 26a61ead-d201-47e3-
b55c-2a7c666942f8 --name NewAppName
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mengedit Aplikasi di Panduan AWS OpsWorks Pengguna](#).

- Untuk detail API, lihat [UpdateApp](#) di Referensi AWS CLI Perintah.

## update-elastic-ip

Contoh kode berikut menunjukkan cara menggunakan `update-elastic-ip`.

### AWS CLI

Untuk memperbarui nama alamat IP Elastis

Contoh berikut memperbarui nama alamat IP Elastis tertentu.

```
aws opsworks --region us-east-1 update-elastic-ip --elastic-ip 54.148.130.96 --name  
NewIPName
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Manajemen Sumber Daya di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [UpdateElasticIp](#) di Referensi AWS CLI Perintah.

## update-instance

Contoh kode berikut menunjukkan cara menggunakan `update-instance`.

### AWS CLI

Untuk memperbarui sebuah instance

Contoh berikut memperbarui jenis instance tertentu.

```
aws opsworks --region us-east-1 update-instance --instance-id  
dfe18b02-5327-493d-91a4-c5c0c448927f --instance-type c3.xlarge
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mengedit Konfigurasi Instans di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [UpdateInstance](#) di Referensi AWS CLI Perintah.

## update-layer

Contoh kode berikut menunjukkan cara menggunakan `update-layer`.

### AWS CLI

Untuk memperbarui lapisan

Contoh berikut memperbarui lapisan tertentu untuk menggunakan instans Amazon EBS yang dioptimalkan.

```
aws opsworks --region us-east-1 update-layer --layer-id
888c5645-09a5-4d0e-95a8-812ef1db76a4 --use-ebs-optimized-instances
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mengedit Konfigurasi OpsWorks Layer di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [UpdateLayer](#) di Referensi AWS CLI Perintah.

## update-my-user-profile

Contoh kode berikut menunjukkan cara menggunakan `update-my-user-profile`.

### AWS CLI

Untuk memperbarui profil pengguna

Contoh berikut memperbarui profil `development` pengguna untuk menggunakan kunci publik SSH tertentu. AWS Kredensial pengguna diwakili oleh `development` profil di `credentials` file (`~\.aws\credentials`), dan kuncinya ada di `.pem` file di direktori kerja.

```
aws opsworks --region us-east-1 --profile development update-my-user-profile --ssh-
public-key file:///development_key.pem
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mengedit Pengaturan AWS OpsWorks Pengguna di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [UpdateMyUserProfile](#) di Referensi AWS CLI Perintah.

## update-rds-db-instance

Contoh kode berikut menunjukkan cara menggunakan `update-rds-db-instance`.

## AWS CLI

Untuk memperbarui instans Amazon RDS DB terdaftar

Contoh berikut memperbarui nilai kata sandi utama instans Amazon RDS. Perhatikan bahwa perintah ini tidak mengubah kata sandi utama instans RDS, hanya kata sandi yang Anda berikan. AWS OpsWorks Jika kata sandi ini tidak cocok dengan kata sandi instans RDS, aplikasi Anda tidak akan dapat terhubung ke database.

```
aws opsworks --region us-east-1 update-rds-db-instance --db-password 123456789
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Mendaftarkan Instans Amazon RDS dengan Tumpukan di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [UpdateRdsDbInstance](#) di Referensi AWS CLI Perintah.

## update-volume

Contoh kode berikut menunjukkan cara menggunakan `update-volume`.

## AWS CLI

Untuk memperbarui volume terdaftar

Contoh berikut memperbarui titik pemasangan volume Amazon Elastic Block Store (Amazon EBS) terdaftar. Volume diidentifikasi oleh ID volumenya, yang merupakan GUID yang AWS OpsWorks menetapkan volume saat Anda mendaftarkannya dengan tumpukan, bukan ID volume Amazon Elastic Compute Cloud (Amazon EC2). :

```
aws opsworks --region us-east-1 update-volume --volume-id 8430177d-52b7-4948-9c62-e195af4703df --mount-point /mnt/myvol
```

Keluaran: Tidak ada.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menetapkan Volume Amazon EBS ke Instans di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [UpdateVolume](#) di Referensi AWS CLI Perintah.

## AWS OpsWorks CM contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS OpsWorks CM.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-node**

Contoh kode berikut menunjukkan cara menggunakan `associate-node`.

AWS CLI

Untuk mengaitkan node

`associate-node` Perintah berikut mengaitkan node bernama `i-44de882p` dengan server Chef Automate bernama `automate-06`, yang berarti bahwa `automate-06` server mengelola node, dan mengkomunikasikan perintah resep ke node melalui perangkat lunak `chef-client` agen yang diinstal pada node oleh perintah `associate-node`. Nama node yang valid adalah ID instans EC2. :

```
aws opsworks-cm associate-node --server-name "automate-06" --node-name
  "i-43de882p" --engine-attributes "Name=CHEF_ORGANIZATION,Value='MyOrganization'
  Name=CHEF_NODE_PUBLIC_KEY,Value='Public_key_contents'"
```

Output yang dikembalikan oleh perintah menyerupai berikut ini. Output:

```
{
  "NodeAssociationStatusToken": "AHUY8wFe4pdXtZC5DiJa5S0Lp5o14DH//
rHRqHDWXxwVoNBxcEy4V7R0NOFymh7E/1Hum0BPsemPQFE6dcGaiFk"
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menambahkan Node Secara Otomatis AWS OpsWorks untuk Chef Automate di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [AssociateNode](#) di Referensi AWS CLI Perintah.

## create-backup

Contoh kode berikut menunjukkan cara menggunakan `create-backup`.

### AWS CLI

Untuk membuat cadangan

`create-backup` Perintah berikut memulai backup manual dari server Chef Automate bernama `automate-06` di `us-east-1` wilayah tersebut. Perintah menambahkan pesan deskriptif ke cadangan dalam `--description` parameter.

```
aws opsworks-cm create-backup \
  --server-name 'automate-06' \
  --description "state of my infrastructure at launch"
```

Output menunjukkan informasi yang mirip dengan berikut ini tentang cadangan baru.

Output:

```
{
  "Backups": [
    {
      "BackupArn": "string",
      "BackupId": "automate-06-20160729133847520",
      "BackupType": "MANUAL",
      "CreatedAt": 2016-07-29T13:38:47.520Z,
      "Description": "state of my infrastructure at launch",
    }
  ]
}
```

```

        "Engine": "Chef",
        "EngineModel": "Single",
        "EngineVersion": "12",
        "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
        "InstanceType": "m4.large",
        "KeyPair": "",
        "PreferredBackupWindow": "",
        "PreferredMaintenanceWindow": "",
        "S3LogUrl": "https://s3.amazonaws.com/<bucket-name>/
automate-06-20160729133847520",
        "SecurityGroupIds": [ "sg-1a24c270" ],
        "ServerName": "automate-06",
        "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-
service-role.1114810729735",
        "Status": "OK",
        "StatusDescription": "",
        "SubnetIds": [ "subnet-49436a18" ],
        "ToolsVersion": "string",
        "UserArn": "arn:aws:iam::1019881987024:user/opsworks-user"
    }
],
}

```

Untuk informasi selengkapnya, lihat [Mencadangkan dan Memulihkan Server Otomatis AWS OpsWorks untuk Chef](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [CreateBackup](#) di Referensi AWS CLI Perintah.

## create-server

Contoh kode berikut menunjukkan cara menggunakan `create-server`.

### AWS CLI

Untuk membuat server

`create-server` Contoh berikut membuat server Chef Automate baru bernama `automate-06` di wilayah default Anda. Perhatikan bahwa default digunakan untuk sebagian besar pengaturan lain, seperti jumlah cadangan yang harus disimpan, dan waktu mulai pemeliharaan dan pencadangan. Sebelum Anda menjalankan `create-server` perintah, lengkapi prasyarat dalam [Memulai dengan AWS OpsWorks untuk Chef Automate di AWS Opsworks for Chef Automate User Guide](#).

```
aws opsworks-cm create-server \  
  --engine "Chef" \  
  --engine-model "Single" \  
  --engine-version "12" \  
  --server-name "automate-06" \  
  --instance-profile-arn "arn:aws:iam::1019881987024:instance-profile/aws-opsworks-cm-ec2-role" \  
  --instance-type "t2.medium" \  
  --key-pair "amazon-test" \  
  --service-role-arn "arn:aws:iam::044726508045:role/aws-opsworks-cm-service-role"
```

Output menunjukkan informasi yang mirip dengan berikut tentang server baru:

```
{  
  "Server": {  
    "BackupRetentionCount": 10,  
    "CreatedAt": 2016-07-29T13:38:47.520Z,  
    "DisableAutomatedBackup": FALSE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "Chef",  
    "EngineAttributes": [  
      {  
        "Name": "CHEF_DELIVERY_ADMIN_PASSWORD",  
        "Value": "1Password1"  
      }  
    ],  
    "EngineModel": "Single",  
    "EngineVersion": "12",  
    "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/aws-opsworks-cm-ec2-role",  
    "InstanceType": "t2.medium",  
    "KeyPair": "amazon-test",  
    "MaintenanceStatus": "",  
    "PreferredBackupWindow": "Sun:02:00",  
    "PreferredMaintenanceWindow": "00:00",  
    "SecurityGroupIds": [ "sg-1a24c270" ],  
    "ServerArn": "arn:aws:iam::1019881987024:instance/automate-06-1010V4UU2WRM2",  
    "ServerName": "automate-06",  
    "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-role",  
    "Status": "CREATING",  
    "StatusReason": "",
```



```
    "SubnetIds": [ "subnet-49436a18" ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [UpdateServer](#) di Referensi API AWS OpsWorks untuk Chef Automate.

- Untuk detail API, lihat [CreateServer](#) di Referensi AWS CLI Perintah.

## delete-backup

Contoh kode berikut menunjukkan cara menggunakan `delete-backup`.

### AWS CLI

Untuk menghapus cadangan

`delete-backup` Perintah berikut menghapus cadangan manual atau otomatis dari server Chef Automate, yang diidentifikasi oleh ID cadangan. Perintah ini berguna ketika Anda mendekati jumlah maksimum cadangan yang dapat Anda simpan, atau Anda ingin meminimalkan biaya penyimpanan Amazon S3 Anda. :

```
aws opsworks-cm delete-backup --backup-id "automate-06-2016-11-19T23:42:40.240Z"
```

Output menunjukkan apakah penghapusan cadangan berhasil.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mencadangkan dan Memulihkan Server Otomatis AWS OpsWorks untuk Chef](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DeleteBackup](#) di Referensi AWS CLI Perintah.

## delete-server

Contoh kode berikut menunjukkan cara menggunakan `delete-server`.

### AWS CLI

Untuk menghapus server

`delete-server` Perintah berikut menghapus server Chef Automate, diidentifikasi dengan nama server. Setelah server dihapus, itu tidak lagi dikembalikan oleh `DescribeServer` permintaan. :

```
aws opsworks-cm delete-server --server-name "automate-06"
```

Output menunjukkan apakah penghapusan server berhasil.

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat Menghapus Server Otomatis AWS OpsWorks untuk Chef di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DeleteServer](#) di Referensi AWS CLI Perintah.

## describe-account-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-account-attributes`.

### AWS CLI

Untuk menggambarkan atribut akun

`describe-account-attributes` Perintah berikut menampilkan informasi tentang penggunaan akun Anda AWS OpsWorks untuk sumber daya Chef Automate. :

```
aws opsworks-cm describe-account-attributes
```

Output untuk setiap entri atribut akun yang dikembalikan oleh perintah menyerupai berikut ini.

Output:

```
{
  "Attributes": [
    {
      "Maximum": 5,
      "Name": "ServerLimit",
      "Used": 2
    }
  ]
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat `DescribeAccountAttributes` di Referensi API AWS OpsWorks untuk Chef Automate.

- Untuk detail API, lihat [DescribeAccountAttributes](#) di Referensi AWS CLI Perintah.

## describe-backups

Contoh kode berikut menunjukkan cara menggunakan `describe-backups`.

### AWS CLI

Untuk menggambarkan cadangan

`describe-backups` Perintah berikut mengembalikan informasi tentang semua backup yang terkait dengan akun Anda di wilayah default Anda.

```
aws opsworks-cm describe-backups
```

Output untuk setiap entri cadangan yang dikembalikan oleh perintah menyerupai berikut ini.

Output:

```
{
  "Backups": [
    {
      "BackupArn": "string",
      "BackupId": "automate-06-20160729133847520",
      "BackupType": "MANUAL",
      "CreatedAt": 2016-07-29T13:38:47.520Z,
      "Description": "state of my infrastructure at launch",
      "Engine": "Chef",
      "EngineModel": "Single",
      "EngineVersion": "12",
      "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
      "InstanceType": "m4.large",
      "KeyPair": "",
      "PreferredBackupWindow": "",
      "PreferredMaintenanceWindow": "",
      "S3LogUrl": "https://s3.amazonaws.com/<bucket-name>/
automate-06-20160729133847520",
      "SecurityGroupIds": [ "sg-1a24c270" ],
      "ServerName": "automate-06",
      "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-
service-role.1114810729735",
```

```

        "Status": "Successful",
        "StatusDescription": "",
        "SubnetIds": [ "subnet-49436a18" ],
        "ToolsVersion": "string",
        "UserArn": "arn:aws:iam::1019881987024:user/opsworks-user"
    },
]
}

```

Untuk informasi selengkapnya, lihat [Mencadangkan dan Memulihkan Server Otomatis AWS OpsWorks untuk Chef](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DescribeBackups](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Untuk menggambarkan peristiwa

`describe-events` Contoh berikut mengembalikan informasi tentang semua peristiwa yang terkait dengan server Chef Automate yang ditentukan.

```

aws opsworks-cm describe-events \
  --server-name 'automate-06'

```

Output untuk setiap entri acara yang dikembalikan oleh perintah menyerupai contoh berikut:

```

{
  "ServerEvents": [
    {
      "CreatedAt": 2016-07-29T13:38:47.520Z,
      "LogUrl": "https://s3.amazonaws.com/<bucket-name>/
automate-06-20160729133847520",
      "Message": "Updates successfully installed.",
      "ServerName": "automate-06"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Tips Pemecahan Masalah Umum](#) di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## **describe-node-association-status**

Contoh kode berikut menunjukkan cara menggunakan `describe-node-association-status`.

### AWS CLI

Untuk menggambarkan status asosiasi simpul

`describe-node-association-status` Perintah berikut mengembalikan status permintaan untuk mengaitkan node dengan server Chef Automate bernama `automate-06` :

```
aws opsworks-cm describe-node-association-status --server-  
name "automate-06" --node-association-status-token "Af1JK1+/  
GoKLZJBdDQEx0065CDi57b1Qe9nKM8joSok0pQ9xr8DqApBN9/106sLdSv1fDEKkEx+eoCHVjoWHa0s="
```

Output untuk setiap entri atribut akun yang dikembalikan oleh perintah menyerupai berikut ini.

Output:

```
{  
  "NodeAssociationStatus": "IN_PROGRESS"  
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat `DescribeNodeAssociationStatus` di Referensi API AWS OpsWorks untuk Chef Automate.

- Untuk detail API, lihat [DescribeNodeAssociationStatus](#) di Referensi AWS CLI Perintah.

## **describe-servers**

Contoh kode berikut menunjukkan cara menggunakan `describe-servers`.

### AWS CLI

Untuk menggambarkan server

`describe-servers` Perintah berikut mengembalikan informasi tentang semua server yang terkait dengan akun Anda, dan di wilayah default Anda. :

```
aws opsworks-cm describe-servers
```

Output untuk setiap entri server yang dikembalikan oleh perintah menyerupai berikut ini. Output:

```
{
  "Servers": [
    {
      "BackupRetentionCount": 8,
      "CreatedAt": 2016-07-29T13:38:47.520Z,
      "DisableAutomatedBackup": FALSE,
      "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",
      "Engine": "Chef",
      "EngineAttributes": [
        {
          "Name": "CHEF_DELIVERY_ADMIN_PASSWORD",
          "Value": "1Password1"
        }
      ],
      "EngineModel": "Single",
      "EngineVersion": "12",
      "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
      "InstanceType": "m4.large",
      "KeyPair": "",
      "MaintenanceStatus": "SUCCESS",
      "PreferredBackupWindow": "03:00",
      "PreferredMaintenanceWindow": "Mon:09:00",
      "SecurityGroupIds": [ "sg-1a24c270" ],
      "ServerArn": "arn:aws:iam::1019881987024:instance/automate-06-1010V4UU2WRM2",
      "ServerName": "automate-06",
      "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-
role.1114810729735",
      "Status": "HEALTHY",
      "StatusReason": "",
      "SubnetIds": [ "subnet-49436a18" ]
    }
  ]
}
```

Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat `DescribeServers` di Panduan AWS OpsWorks untuk Chef Automate API.

- Untuk detail API, lihat [DescribeServers](#) di Referensi AWS CLI Perintah.

## **disassociate-node**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-node`.

### AWS CLI

Untuk memisahkan node

`disassociate-node` Perintah berikut memisahkan node bernama `i-44de882p`, menghapus node dari manajemen oleh server Chef Automate bernama `automate-06`. Nama node yang valid adalah ID instans EC2. :

```
aws opsworks-cm disassociate-node --server-name "automate-06" --node-name
  "i-43de882p" --engine-attributes "Name=CHEF_ORGANIZATION,Value='MyOrganization'
  Name=CHEF_NODE_PUBLIC_KEY,Value='Public_key_contents'"
```

Output yang dikembalikan oleh perintah menyerupai berikut ini. Output:

```
{
  "NodeAssociationStatusToken": "AHUY8wFe4pdXtZC5DiJa5S0Lp5o14DH//
  rHRqHDWxwVoNBxcEy4V7R0N0Fymh7E/1Hum0BPsemPQFE6dcGaiFk"
}
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat `Menghapus Server Otomatis AWS OpsWorks untuk Chef` di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [DisassociateNode](#) di Referensi AWS CLI Perintah.

## **restore-server**

Contoh kode berikut menunjukkan cara menggunakan `restore-server`.

### AWS CLI

Untuk memulihkan server

`restore-server` Perintah berikut melakukan restorasi di tempat server Chef Automate bernama `automate-06` di wilayah default Anda dari cadangan dengan ID dari `automate-06-2016-11-22T16:13:27.998Z`. Memulihkan server mengembalikan koneksi ke node yang dikelola server Chef Automate pada saat pencadangan yang ditentukan dilakukan.

```
aws opsworks-cm restore-server --backup-id "automate-06-2016-11-22T 16:13:27.998 Z" --server-name "automate-06"
```

Outputnya hanya ID perintah. Output:

```
(None)
```

### Informasi Lebih Lanjut

Untuk informasi selengkapnya, lihat [Mengembalikan Server Otomatis yang Gagal AWS OpsWorks untuk Chef](#) di Panduan AWS OpsWorks Pengguna.

- Untuk detail API, lihat [RestoreServer](#) di Referensi AWS CLI Perintah.

## start-maintenance

Contoh kode berikut menunjukkan cara menggunakan `start-maintenance`.

### AWS CLI

Untuk memulai pemeliharaan

`start-maintenance` Contoh berikut secara manual memulai pemeliharaan pada server Chef Automate atau Puppet Enterprise yang ditentukan di wilayah default Anda. Perintah ini berguna jika upaya pemeliharaan otomatis sebelumnya gagal, dan penyebab kegagalan pemeliharaan telah diselesaikan.

```
aws opsworks-cm start-maintenance \  
  --server-name 'automate-06'
```

Output:

```
{  
  "Server": {  
    "AssociatePublicIpAddress": true,  
  }  
}
```



```

    "BackupRetentionCount": 10,
    "ServerName": "automate-06",
    "CreatedAt": 1569229584.842,
    "CloudFormationStackArn": "arn:aws:cloudformation:us-
west-2:123456789012:stack/aws-opsworks-cm-instance-automate-06-1606611794746/
EXAMPLE0-31de-11eb-bdb0-0a5b0a1353b8",
    "DisableAutomatedBackup": false,
    "Endpoint": "automate-06-EXAMPLEvr8gjfk5f.us-west-2.opsworks-cm.io",
    "Engine": "ChefAutomate",
    "EngineModel": "Single",
    "EngineAttributes": [],
    "EngineVersion": "2020-07",
    "InstanceProfileArn": "arn:aws:iam::123456789012:instance-profile/aws-
opsworks-cm-ec2-role",
    "InstanceType": "m5.large",
    "PreferredMaintenanceWindow": "Sun:01:00",
    "PreferredBackupWindow": "Sun:15:00",
    "SecurityGroupIds": [
        "sg-EXAMPLE"
    ],
    "ServiceRoleArn": "arn:aws:iam::123456789012:role/service-role/aws-opsworks-
cm-service-role",
    "Status": "UNDER_MAINTENANCE",
    "SubnetIds": [
        "subnet-EXAMPLE"
    ],
    "ServerArn": "arn:aws:opsworks-cm:us-west-2:123456789012:server/
automate-06/0148382d-66b0-4196-8274-d1a2b6dff8d1"
}
}

```

Untuk informasi selengkapnya, lihat [Pemeliharaan Sistem \(server Perusahaan Boneka\)](#) atau [Pemeliharaan Sistem \(server Chef Automate\)](#) di AWS OpsWorks Panduan Pengguna.

- Untuk detail API, lihat [StartMaintenance](#) di Referensi AWS CLI Perintah.

## update-server-engine-attributes

Contoh kode berikut menunjukkan cara menggunakan `update-server-engine-attributes`.

### AWS CLI

Untuk memperbarui atribut mesin server

`update-server-engine-attributes` Perintah berikut memperbarui nilai atribut `CHEF_PIVOTAL_KEY` engine untuk server Chef Automate bernama `automate-06`. Saat ini tidak mungkin untuk mengubah nilai atribut mesin lainnya.

```
aws opsworks-cm update-server-engine-attributes \  
  --attribute-name CHEF_PIVOTAL_KEY \  
  --attribute-value "new key value" \  
  --server-name "automate-06"
```

Output menunjukkan informasi yang mirip dengan berikut ini tentang server yang diperbarui.

```
{  
  "Server": {  
    "BackupRetentionCount": 2,  
    "CreatedAt": 2016-07-29T13:38:47.520Z,  
    "DisableAutomatedBackup": FALSE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "Chef",  
    "EngineAttributes": [  
      {  
        "Name": "CHEF_PIVOTAL_KEY",  
        "Value": "new key value"  
      }  
    ],  
    "EngineModel": "Single",  
    "EngineVersion": "12",  
    "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/  
automate-06-1010V4UU2WRM2",  
    "InstanceType": "m4.large",  
    "KeyPair": "",  
    "MaintenanceStatus": "SUCCESS",  
    "PreferredBackupWindow": "Mon:09:15",  
    "PreferredMaintenanceWindow": "03:00",  
    "SecurityGroupIds": [ "sg-1a24c270" ],  
    "ServerArn": "arn:aws:iam::1019881987024:instance/  
automate-06-1010V4UU2WRM2",  
    "ServerName": "automate-06",  
    "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-  
role.1114810729735",  
    "Status": "HEALTHY",  
    "StatusReason": "",  
    "SubnetIds": [ "subnet-49436a18" ]  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [UpdateServerEngineAttributes](#) di Referensi API AWS OpsWorks untuk Chef Automate.

- Untuk detail API, lihat [UpdateServerEngineAttributes](#) di Referensi AWS CLI Perintah.

## update-server

Contoh kode berikut menunjukkan cara menggunakan `update-server`.

### AWS CLI

Untuk memperbarui server

`update-server` Perintah berikut memperbarui waktu mulai pemeliharaan server Chef Automate yang ditentukan di wilayah default Anda. `--preferred-maintenance-window` Parameter ditambahkan untuk mengubah hari mulai dan waktu pemeliharaan server ke hari Senin pukul 9:15 pagi. UTC. :

```
aws opsworks-cm update-server \  
  --server-name "automate-06" \  
  --preferred-maintenance-window "Mon:09:15"
```

Output menunjukkan informasi yang mirip dengan berikut ini tentang server yang diperbarui.

```
{  
  "Server": {  
    "BackupRetentionCount": 8,  
    "CreatedAt": "2016-07-29T13:38:47.520Z",  
    "DisableAutomatedBackup": TRUE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "Chef",  
    "EngineAttributes": [  
      {  
        "Name": "CHEF_DELIVERY_ADMIN_PASSWORD",  
        "Value": "1Password1"  
      }  
    ],  
    "EngineModel": "Single",  
    "EngineVersion": "12",
```

```

    "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
    "InstanceType": "m4.large",
    "KeyPair": "",
    "MaintenanceStatus": "OK",
    "PreferredBackupWindow": "Mon:09:15",
    "PreferredMaintenanceWindow": "03:00",
    "SecurityGroupIds": [ "sg-1a24c270" ],
    "ServerArn": "arn:aws:iam::1019881987024:instance/
automate-06-1010V4UU2WRM2",
    "ServerName": "automate-06",
    "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-
role.1114810729735",
    "Status": "HEALTHY",
    "StatusReason": "",
    "SubnetIds": [ "subnet-49436a18" ]
  }
}

```

Untuk informasi selengkapnya, lihat [UpdateServer](#) di Referensi API AWS OpsWorks untuk Chef Automate.

- Untuk detail API, lihat [UpdateServer](#) di Referensi AWS CLI Perintah.

## Organizations contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Organizations.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **accept-handshake**

Contoh kode berikut menunjukkan cara menggunakan `accept-handshake`.

#### AWS CLI

Untuk menerima jabatan tangan dari akun lain

Bill, pemilik sebuah organisasi, sebelumnya telah mengundang akun Juan untuk bergabung dengan organisasinya. Contoh berikut menunjukkan akun Juan menerima jabatan tangan dan dengan demikian menyetujui undangan tersebut.

```
aws organizations accept-handshake --handshake-id h-examplehandshakeid111
```

Outputnya menunjukkan hal berikut:

```
{
  "Handshake": {
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
    "RequestedTimestamp": 1481656459.257,
    "ExpirationTimestamp": 1482952459.257,
    "Id": "h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "juan@example.com",
        "Type": "EMAIL"
      }
    ],
    "Resources": [
      {
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@amazon.com"
          }
        ]
      }
    ]
  }
}
```

```

        {
            "Type": "MASTER_NAME",
            "Value": "Org Master Account"
        },
        {
            "Type": "ORGANIZATION_FEATURE_SET",
            "Value": "ALL"
        }
    ],
    "Type": "ORGANIZATION",
    "Value": "o-exampleorgid"
},
{
    "Type": "EMAIL",
    "Value": "juan@example.com"
}
],
"State": "ACCEPTED"
}
}

```

- Untuk detail API, lihat [AcceptHandshake](#) di Referensi AWS CLI Perintah.

## attach-policy

Contoh kode berikut menunjukkan cara menggunakan `attach-policy`.

### AWS CLI

Untuk melampirkan kebijakan ke root, OU, atau akun

#### Contoh 1

Contoh berikut menunjukkan cara melampirkan kebijakan kontrol layanan (SCP) ke OU:

```

aws organizations attach-policy
    --policy-id p-examplepolicyid111
    --target-id ou-examplerootid111-exampleouid111

```

#### Contoh 2

Contoh berikut menunjukkan cara melampirkan kebijakan kontrol layanan langsung ke akun:

```
aws organizations attach-policy
    --policy-id p-examplepolicyid111
    --target-id 333333333333
```

- Untuk detail API, lihat [AttachPolicy](#) di Referensi AWS CLI Perintah.

## cancel-handshake

Contoh kode berikut menunjukkan cara menggunakan `cancel-handshake`.

### AWS CLI

Untuk membatalkan jabat tangan yang dikirim dari akun lain

Bill sebelumnya mengirim undangan ke akun Susan untuk bergabung dengan organisasinya. Dia berubah pikiran dan memutuskan untuk membatalkan undangan sebelum Susan menerimanya. Contoh berikut menunjukkan pembatalan Bill:

```
aws organizations cancel-handshake --handshake-id h-examplehandshakeid111
```

Outputnya mencakup objek jabat tangan yang menunjukkan bahwa statusnya sekarang `CANCELED`:

```
{
  "Handshake": {
    "Id": "h-examplehandshakeid111",
    "State": "CANCELED",
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "susan@example.com",
        "Type": "EMAIL"
      }
    ]
  },
}
```

```

    "Resources": [
      {
        "Type": "ORGANIZATION",
        "Value": "o-exampleorgid",
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@example.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Master Account"
          },
          {
            "Type": "ORGANIZATION_FEATURE_SET",
            "Value": "CONSOLIDATED_BILLING"
          }
        ]
      },
      {
        "Type": "EMAIL",
        "Value": "anika@example.com"
      },
      {
        "Type": "NOTES",
        "Value": "This is a request for Susan's account to
join Bob's organization."
      }
    ],
    "RequestedTimestamp": 1.47008383521E9,
    "ExpirationTimestamp": 1.47137983521E9
  }
}

```

- Untuk detail API, lihat [CancelHandshake](#) di Referensi AWS CLI Perintah.

## create-account

Contoh kode berikut menunjukkan cara menggunakan `create-account`.

### AWS CLI

Untuk membuat akun anggota yang secara otomatis menjadi bagian dari organisasi



Contoh berikut menunjukkan cara membuat akun anggota dalam suatu organisasi. Akun anggota dikonfigurasi dengan nama Akun Produksi dan alamat email `susan@example.com`. Organizations secara otomatis membuat peran IAM menggunakan nama default `OrganizationAccountAccessRole` karena parameter `RoleName` tidak ditentukan. Selain itu, pengaturan yang memungkinkan pengguna IAM atau peran dengan izin yang cukup untuk mengakses data penagihan akun diatur ke nilai default `ALLOW` karena `IamUserAccessToBilling` parameter tidak ditentukan. Organizations secara otomatis mengirimkan email “Selamat Datang di AWS” kepada Susan:

```
aws organizations create-account --email susan@example.com --account-name
"Production Account"
```

Outputnya mencakup objek permintaan yang menunjukkan bahwa statusnya sekarang `IN_PROGRESS`:

```
{
  "CreateAccountStatus": {
    "State": "IN_PROGRESS",
    "Id": "car-examplecreateaccountrequestid111"
  }
}
```

Anda nantinya dapat menanyakan status permintaan saat ini dengan memberikan nilai respons `Id` ke `describe-create-account-status` perintah sebagai nilai untuk `create-account-request-id` parameter.

Untuk informasi selengkapnya, lihat [Membuat AWS Akun di Organisasi Anda](#) di Panduan Pengguna AWS Organizations.

- Untuk detail API, lihat [CreateAccount](#) di Referensi AWS CLI Perintah.

## **create-organization**

Contoh kode berikut menunjukkan cara menggunakan `create-organization`.

### AWS CLI

Contoh 1: Untuk membuat organisasi baru

Bill ingin membuat organisasi menggunakan kredensi dari akun `111111111111`. Contoh berikut menunjukkan bahwa akun menjadi akun utama di organisasi baru. Karena dia tidak menentukan

set fitur, organisasi baru default ke semua fitur yang diaktifkan dan kebijakan kontrol layanan diaktifkan di root.

```
aws organizations create-organization
```

Outputnya mencakup objek organisasi dengan rincian tentang organisasi baru:

```
{
  "Organization": {
    "AvailablePolicyTypes": [
      {
        "Status": "ENABLED",
        "Type": "SERVICE_CONTROL_POLICY"
      }
    ],
    "MasterAccountId": "111111111111",
    "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
    "MasterAccountEmail": "bill@example.com",
    "FeatureSet": "ALL",
    "Id": "o-exampleorgid",
    "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid"
  }
}
```

Contoh 2: Untuk membuat organisasi baru dengan hanya fitur penagihan terkonsolidasi yang diaktifkan

Contoh berikut membuat organisasi yang hanya mendukung fitur penagihan konsolidasi:

```
aws organizations create-organization --feature-set CONSOLIDATED_BILLING
```

Outputnya mencakup objek organisasi dengan rincian tentang organisasi baru:

```
{
  "Organization": {
    "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
    "AvailablePolicyTypes": [],
    "Id": "o-exampleorgid",
  }
}
```

```
    "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
    "MasterAccountEmail": "bill@example.com",
    "MasterAccountId": "111111111111",
    "FeatureSet": "CONSOLIDATED_BILLING"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Organisasi](#) di Panduan Pengguna AWS Organizations.

- Untuk detail API, lihat [CreateOrganization](#) di Referensi AWS CLI Perintah.

## **create-organizational-unit**

Contoh kode berikut menunjukkan cara menggunakan `create-organizational-unit`.

### AWS CLI

Untuk membuat OU di root atau induk OU

Contoh berikut menunjukkan cara membuat OU yang bernama `AccountingOu`:

```
aws organizations create-organizational-unit --parent-id r-examplerootid111 --name
AccountingOU
```

Outputnya mencakup objek `OrganizationalUnit` dengan detail tentang OU baru:

```
{
  "OrganizationalUnit": {
    "Id": "ou-examplerootid111-exampleoid111",
    "Arn": "arn:aws:organizations::111111111111:ou/o-exampleorgid/ou-
examplerootid111-exampleoid111",
    "Name": "AccountingOU"
  }
}
```

- Untuk detail API, lihat [CreateOrganizationalUnit](#) di Referensi AWS CLI Perintah.

## **create-policy**

Contoh kode berikut menunjukkan cara menggunakan `create-policy`.

## AWS CLI

Contoh 1: Untuk membuat kebijakan dengan file sumber teks untuk kebijakan JSON

Contoh berikut menunjukkan cara membuat kebijakan kontrol layanan (SCP) bernama `AllowAllS3Actions`. Isi kebijakan diambil dari file di komputer lokal yang disebut `policy.json`.

```
aws organizations create-policy --content file://policy.json --name
AllowAllS3Actions, --type SERVICE_CONTROL_POLICY --description "Allows delegation
of all S3 actions"
```

Outputnya mencakup objek kebijakan dengan detail tentang kebijakan baru:

```
{
  "Policy": {
    "Content": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\\\"Allow\\\",\\\"Action\":[\\\"s3:*\\\"],\\\"Resource\":[\\\"*\\\"]}]}",
    "PolicySummary": {
      "Arn": "arn:aws:organizations::o-exampleorgid:policy/
service_control_policy/p-examplepolicyid111",
      "Description": "Allows delegation of all S3 actions",
      "Name": "AllowAllS3Actions",
      "Type": "SERVICE_CONTROL_POLICY"
    }
  }
}
```

Contoh 2: Untuk membuat kebijakan dengan kebijakan JSON sebagai parameter

Contoh berikut menunjukkan cara membuat SCP yang sama, kali ini dengan menyematkan konten kebijakan sebagai string JSON dalam parameter. String harus lolos dengan garis miring terbalik sebelum tanda kutip ganda untuk memastikan bahwa mereka diperlakukan sebagai literal dalam parameter, yang dengan sendirinya dikelilingi oleh tanda kutip ganda:

```
aws organizations create-policy --content "{\"Version\":\"2012-10-17\",\"Statement
\":[{\"Effect\":\"Allow\",\"Action\":[\"s3:*\"],\"Resource\":[\"*\"]}]" --name
AllowAllS3Actions --type SERVICE_CONTROL_POLICY --description "Allows delegation of
all S3 actions"
```

Untuk informasi selengkapnya tentang membuat dan menggunakan kebijakan di organisasi Anda, lihat [Mengelola Kebijakan AWS Organisasi](#) di [Panduan Pengguna Organizations](#).

- Untuk detail API, lihat [CreatePolicy](#) di Referensi AWS CLI Perintah.

## decline-handshake

Contoh kode berikut menunjukkan cara menggunakan `decline-handshake`.

### AWS CLI

Untuk menolak jabat tangan yang dikirim dari akun lain

Contoh berikut menunjukkan bahwa Susan, admin yang merupakan pemilik akun 2222222222222222, menolak undangan untuk bergabung dengan organisasi Bill.

DeclineHandshake Operasi mengembalikan objek jabat tangan, menunjukkan bahwa keadaan sekarang DITOLAK:

```
aws organizations decline-handshake --handshake-id h-examplehandshakeid111
```

Outputnya mencakup objek jabat tangan yang menunjukkan status baru: DECLINED

```
{
  "Handshake": {
    "Id": "h-examplehandshakeid111",
    "State": "DECLINED",
    "Resources": [
      {
        "Type": "ORGANIZATION",
        "Value": "o-exampleorgid",
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@example.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Master Account"
          }
        ]
      }
    ],
    {
      "Type": "EMAIL",
      "Value": "susan@example.com"
    }
  }
}
```

```

        },
        {
            "Type": "NOTES",
            "Value": "This is an invitation to Susan's account
to join the Bill's organization."
        }
    ],
    "Parties": [
        {
            "Type": "EMAIL",
            "Id": "susan@example.com"
        },
        {
            "Type": "ORGANIZATION",
            "Id": "o-exampleorgid"
        }
    ],
    "Action": "INVITE",
    "RequestedTimestamp": 1470684478.687,
    "ExpirationTimestamp": 1471980478.687,
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111"
    }
}

```

- Untuk detail API, lihat [DeclineHandshake](#) di Referensi AWS CLI Perintah.

## delete-organization

Contoh kode berikut menunjukkan cara menggunakan `delete-organization`.

### AWS CLI

Untuk menghapus organisasi

Contoh berikut menunjukkan cara menghapus organisasi. Untuk melakukan operasi ini, Anda harus menjadi admin akun master di organisasi. Contoh ini mengasumsikan bahwa Anda sebelumnya menghapus semua akun anggota, OU, dan kebijakan dari organisasi:

```
aws organizations delete-organization
```

- Untuk detail API, lihat [DeleteOrganization](#) di Referensi AWS CLI Perintah.

## **delete-organizational-unit**

Contoh kode berikut menunjukkan cara menggunakan `delete-organizational-unit`.

### AWS CLI

Untuk menghapus OU

Contoh berikut menunjukkan cara menghapus OU. Contoh mengasumsikan bahwa Anda sebelumnya menghapus semua akun dan OU lainnya dari OU:

```
aws organizations delete-organizational-unit --organizational-unit-id ou-examplerootid111-exampleouid111
```

- Untuk detail API, lihat [DeleteOrganizationalUnit](#) di Referensi AWS CLI Perintah.

## **delete-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-policy`.

### AWS CLI

Untuk menghapus kebijakan

Contoh berikut menunjukkan cara menghapus kebijakan dari organisasi. Contoh ini mengasumsikan bahwa Anda sebelumnya melepaskan kebijakan dari semua entitas:

```
aws organizations delete-policy --policy-id p-examplepolicyid111
```

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## **describe-account**

Contoh kode berikut menunjukkan cara menggunakan `describe-account`.

### AWS CLI

Untuk mendapatkan detail tentang akun

Contoh berikut menunjukkan kepada Anda cara meminta detail tentang akun:

```
aws organizations describe-account --account-id 555555555555
```

Output menunjukkan objek akun dengan rincian tentang akun:

```
{
  "Account": {
    "Id": "555555555555",
    "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/555555555555",
    "Name": "Beta account",
    "Email": "anika@example.com",
    "JoinedMethod": "INVITED",
    "JoinedTimeStamp": 1481756563.134,
    "Status": "ACTIVE"
  }
}
```

- Untuk detail API, lihat [DescribeAccount](#) di Referensi AWS CLI Perintah.

## describe-create-account-status

Contoh kode berikut menunjukkan cara menggunakan `describe-create-account-status`.

### AWS CLI

Untuk mendapatkan status terbaru tentang permintaan untuk membuat akun

Contoh berikut menunjukkan cara meminta status terbaru untuk permintaan sebelumnya untuk membuat akun di organisasi. `--request-id` yang ditentukan berasal dari respons panggilan asli ke `create-account`. Permintaan pembuatan akun menunjukkan dengan bidang status bahwa Organizations berhasil menyelesaikan pembuatan akun.

Perintah:

```
aws organizations describe-create-account-status --create-account-request-id car-
examplecreateaccountrequestid111
```

Output:

```
{
  "CreateAccountStatus": {
    "State": "SUCCEEDED",
    "AccountId": "555555555555",
    "AccountName": "Beta account",
```



```

    "RequestedTimestamp": 1470684478.687,
    "CompletedTimestamp": 1470684532.472,
    "Id": "car-examplecreateaccountrequestid111"
  }
}

```

- Untuk detail API, lihat [DescribeCreateAccountStatus](#) di Referensi AWS CLI Perintah.

## describe-handshake

Contoh kode berikut menunjukkan cara menggunakan `describe-handshake`.

### AWS CLI

Untuk mendapatkan informasi tentang jabatan tangan

Contoh berikut menunjukkan kepada Anda cara meminta detail tentang jabatan tangan. ID jabatan tangan berasal dari panggilan asli ke `InviteAccountToOrganization`, atau dari panggilan ke `ListHandshakesForAccount` atau `ListHandshakesForOrganization`:

```
aws organizations describe-handshake --handshake-id h-examplehandshakeid111
```

Outputnya mencakup objek jabatan tangan yang memiliki semua detail tentang jabatan tangan yang diminta:

```

{
  "Handshake": {
    "Id": "h-examplehandshakeid111",
    "State": "OPEN",
    "Resources": [
      {
        "Type": "ORGANIZATION",
        "Value": "o-exampleorgid",
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@example.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Master Account"
          }
        ]
      }
    ]
  }
}

```

```

    ],
    "Parties": [
      {
        "Type": "ORGANIZATION",
        "Id": "o-exampleorgid"
      },
      {
        "Type": "EMAIL",
        "Id": "anika@example.com"
      }
    ],
    "Action": "INVITE",
    "RequestedTimestamp": 1470158698.046,
    "ExpirationTimestamp": 1471454698.046,
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111"
  }
}

```

- Untuk detail API, lihat [DescribeHandshake](#) di Referensi AWS CLI Perintah.

## describe-organization

Contoh kode berikut menunjukkan cara menggunakan `describe-organization`.

### AWS CLI

Untuk mendapatkan informasi tentang organisasi saat ini

Contoh berikut menunjukkan cara meminta detail tentang organisasi:

```
aws organizations describe-organization
```

Outputnya mencakup objek organisasi yang memiliki rincian tentang organisasi:

```
{
  "Organization": {
```

```

    "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
    "MasterAccountEmail": "bill@example.com",
    "MasterAccountId": "111111111111",
    "Id": "o-exampleorgid",
    "FeatureSet": "ALL",
    "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
    "AvailablePolicyTypes": [
        {
            "Status": "ENABLED",
            "Type": "SERVICE_CONTROL_POLICY"
        }
    ]
}

```

- Untuk detail API, lihat [DescribeOrganization](#) di Referensi AWS CLI Perintah.

## describe-organizational-unit

Contoh kode berikut menunjukkan cara menggunakan `describe-organizational-unit`.

### AWS CLI

Untuk mendapatkan informasi tentang OU

`describe-organizational-unit` Contoh berikut meminta rincian tentang OU.

```
aws organizations describe-organizational-unit \
  --organizational-unit-id ou-examplerootid111-exampleoid111
```

Output:

```

{
  "OrganizationalUnit": {
    "Name": "Accounting Group",
    "Arn": "arn:aws:organizations::123456789012:ou/o-exampleorgid/ou-
examplerootid111-exampleoid111",
    "Id": "ou-examplerootid111-exampleoid111"
  }
}

```

- Untuk detail API, lihat [DescribeOrganizationalUnit](#) di Referensi AWS CLI Perintah.

## describe-policy

Contoh kode berikut menunjukkan cara menggunakan `describe-policy`.

### AWS CLI

Untuk mendapatkan informasi tentang kebijakan

Contoh berikut menunjukkan cara meminta informasi tentang kebijakan:

```
aws organizations describe-policy --policy-id p-examplepolicyid111
```

Outputnya mencakup objek kebijakan yang berisi detail tentang kebijakan:

```
{
  "Policy": {
    "Content": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": [\n    {\n      \"Effect\": \"Allow\",\n      \"Action\": \"*\",\n      \"Resource\": \"*\"\n    }\n  ]\n}",
    "PolicySummary": {
      "Arn": "arn:aws:organizations::111111111111:policy/o-exampleorgid/service_control_policy/p-examplepolicyid111",
      "Type": "SERVICE_CONTROL_POLICY",
      "Id": "p-examplepolicyid111",
      "AwsManaged": false,
      "Name": "AllowAllS3Actions",
      "Description": "Enables admins to delegate S3 permissions"
    }
  }
}
```

- Untuk detail API, lihat [DescribePolicy](#) di Referensi AWS CLI Perintah.

## detach-policy

Contoh kode berikut menunjukkan cara menggunakan `detach-policy`.

### AWS CLI

Untuk melepaskan kebijakan dari root, OU, atau akun

Contoh berikut menunjukkan cara melepaskan kebijakan dari OU:

```
aws organizations detach-policy --target-id ou-examplerootid111-exampleouid111 --policy-id p-examplepolicyid111
```

- Untuk detail API, lihat [DetachPolicy](#) di Referensi AWS CLI Perintah.

## **disable-policy-type**

Contoh kode berikut menunjukkan cara menggunakan `disable-policy-type`.

### AWS CLI

Untuk menonaktifkan jenis kebijakan di root

Contoh berikut menunjukkan cara menonaktifkan jenis kebijakan kebijakan kontrol layanan (SCP) di root:

```
aws organizations disable-policy-type --root-id r-examplerootid111 --policy-type SERVICE_CONTROL_POLICY
```

Output menunjukkan bahwa elemen `PolicyTypes` respon tidak lagi menyertakan `SERVICE_CONTROL_POLICY`:

```
{
  "Root": {
    "PolicyTypes": [],
    "Name": "Root",
    "Id": "r-examplerootid111",
    "Arn": "arn:aws:organizations::111111111111:root/o-exampleorgid/r-examplerootid111"
  }
}
```

- Untuk detail API, lihat [DisablePolicyType](#) di Referensi AWS CLI Perintah.

## **enable-all-features**

Contoh kode berikut menunjukkan cara menggunakan `enable-all-features`.

## AWS CLI

Untuk mengaktifkan semua fitur dalam suatu organisasi

Contoh ini menunjukkan administrator yang meminta semua akun yang diundang di organisasi untuk menyetujui semua fitur yang diaktifkan di organisasi. AWS Organizations mengirimkan email ke alamat yang terdaftar di setiap akun anggota yang diundang meminta pemilik untuk menyetujui perubahan ke semua fitur dengan menerima jabat tangan yang dikirim. Setelah semua akun anggota yang diundang menerima jabat tangan, administrator organisasi dapat menyelesaikan perubahan ke semua fitur, dan mereka yang memiliki izin yang sesuai dapat membuat kebijakan dan menerapkannya ke root, OU, dan akun:

```
aws organizations enable-all-features
```

Outputnya adalah objek jabat tangan yang dikirim ke semua akun anggota yang diundang untuk persetujuan:

```
{
  "Handshake": {
    "Action": "ENABLE_ALL_FEATURES",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-exampleorgid/enable_all_features/h-examplehandshakeid111",
    "ExpirationTimestamp": 1.483127868609E9,
    "Id": "h-examplehandshakeid111",
    "Parties": [
      {
        "id": "o-exampleorgid",
        "type": "ORGANIZATION"
      }
    ],
    "requestedTimestamp": 1.481831868609E9,
    "resources": [
      {
        "type": "ORGANIZATION",
        "value": "o-exampleorgid"
      }
    ],
    "state": "REQUESTED"
  }
}
```

- Untuk detail API, lihat [EnableAllFeatures](#) di Referensi AWS CLI Perintah.

## enable-policy-type

Contoh kode berikut menunjukkan cara menggunakan `enable-policy-type`.

### AWS CLI

Untuk mengaktifkan penggunaan tipe kebijakan di root

Contoh berikut menunjukkan cara mengaktifkan jenis kebijakan kebijakan kontrol layanan (SCP) di root:

```
aws organizations enable-policy-type --root-id r-examplerootid111 --policy-type
SERVICE_CONTROL_POLICY
```

Output menunjukkan objek root dengan elemen respons `PolicyTypes` yang menunjukkan bahwa SCP sekarang diaktifkan:

```
{
  "Root": {
    "PolicyTypes": [
      {
        "Status": "ENABLED",
        "Type": "SERVICE_CONTROL_POLICY"
      }
    ],
    "Id": "r-examplerootid111",
    "Name": "Root",
    "Arn": "arn:aws:organizations::111111111111:root/o-exampleorgid/r-
examplerootid111"
  }
}
```

- Untuk detail API, lihat [EnablePolicyType](#) di Referensi AWS CLI Perintah.

## invite-account-to-organization

Contoh kode berikut menunjukkan cara menggunakan `invite-account-to-organization`.

### AWS CLI

Mengundang akun untuk bergabung dengan organisasi

Contoh berikut menunjukkan akun master yang dimiliki oleh bill@example.com mengundang akun milik juan@example.com untuk bergabung dengan organisasi:

```
aws organizations invite-account-to-organization --target '{"Type": "EMAIL", "Id": "juan@example.com"}' --notes "This is a request for Juan's account to join Bill's organization."
```

Outputnya mencakup struktur jabat tangan yang menunjukkan apa yang dikirim ke akun yang diundang:

```
{
  "Handshake": {
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-exampleorgid/invite/h-examplehandshakeid111",
    "ExpirationTimestamp": 1482952459.257,
    "Id": "h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "juan@example.com",
        "Type": "EMAIL"
      }
    ],
    "RequestedTimestamp": 1481656459.257,
    "Resources": [
      {
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@amazon.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Org Master Account"
          },
          {
            "Type": "ORGANIZATION_FEATURE_SET",
            "Value": "FULL"
          }
        ]
      }
    ]
  }
}
```



```
    ],  
    "Type": "ORGANIZATION",  
    "Value": "o-exampleorgid"  
  },  
  {  
    "Type": "EMAIL",  
    "Value": "juan@example.com"  
  }  
],  
"State": "OPEN"  
}  
}
```

- Untuk detail API, lihat [InviteAccountToOrganization](#) di Referensi AWS CLI Perintah.

## leave-organization

Contoh kode berikut menunjukkan cara menggunakan `leave-organization`.

### AWS CLI

Untuk meninggalkan organisasi sebagai akun anggota

Contoh berikut menunjukkan administrator akun anggota yang meminta untuk meninggalkan organisasi yang saat ini menjadi anggota:

```
aws organizations leave-organization
```

- Untuk detail API, lihat [LeaveOrganization](#) di Referensi AWS CLI Perintah.

## list-accounts-for-parent

Contoh kode berikut menunjukkan cara menggunakan `list-accounts-for-parent`.

### AWS CLI

Untuk mengambil daftar semua akun di root induk tertentu atau OU

Contoh berikut menunjukkan cara meminta daftar akun di OU:

```
aws organizations list-accounts-for-parent --parent-id ou-examplerootid111-  
exampleouid111
```

Outputnya mencakup daftar objek ringkasan akun.

```
{
  "Accounts": [
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481835795.536,
      "Id": "333333333333",
      "Name": "Development Account",
      "Email": "juan@example.com",
      "Status": "ACTIVE"
    },
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/444444444444",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481835812.143,
      "Id": "444444444444",
      "Name": "Test Account",
      "Email": "anika@example.com",
      "Status": "ACTIVE"
    }
  ]
}
```

- Untuk detail API, lihat [ListAccountsForParent](#) di Referensi AWS CLI Perintah.

## list-accounts

Contoh kode berikut menunjukkan cara menggunakan `list-accounts`.

### AWS CLI

Untuk mengambil daftar semua akun dalam suatu organisasi

Contoh berikut menunjukkan cara meminta daftar akun di organisasi:

```
aws organizations list-accounts
```

Outputnya mencakup daftar objek ringkasan akun.

```
{
  "Accounts": [
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481830215.45,
      "Id": "111111111111",
      "Name": "Master Account",
      "Email": "bill@example.com",
      "Status": "ACTIVE"
    },
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/222222222222",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481835741.044,
      "Id": "222222222222",
      "Name": "Production Account",
      "Email": "alice@example.com",
      "Status": "ACTIVE"
    },
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481835795.536,
      "Id": "333333333333",
      "Name": "Development Account",
      "Email": "juan@example.com",
      "Status": "ACTIVE"
    },
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/444444444444",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481835812.143,
      "Id": "444444444444",
      "Name": "Test Account",
      "Email": "anika@example.com",
      "Status": "ACTIVE"
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [ListAccounts](#) di Referensi AWS CLI Perintah.

## list-children

Contoh kode berikut menunjukkan cara menggunakan `list-children`.

### AWS CLI

Untuk mengambil akun anak dan OU dari OU orang tua atau root

Berikut contoh Anda bagaimana untuk daftar root atau OU yang berisi account 444444444444:

```
aws organizations list-children --child-type ORGANIZATIONAL_UNIT --parent-id ou-examplerootid111-exampleouid111
```

Output menunjukkan dua anak OU yang terkandung oleh orang tua:

```
{
  "Children": [
    {
      "Id": "ou-examplerootid111-exampleouid111",
      "Type": "ORGANIZATIONAL_UNIT"
    },
    {
      "Id": "ou-examplerootid111-exampleouid222",
      "Type": "ORGANIZATIONAL_UNIT"
    }
  ]
}
```

- Untuk detail API, lihat [ListChildren](#) di Referensi AWS CLI Perintah.

## list-create-account-status

Contoh kode berikut menunjukkan cara menggunakan `list-create-account-status`.

### AWS CLI

Contoh 1: Untuk mengambil daftar permintaan pembuatan akun yang dibuat di organisasi saat ini

Contoh berikut menunjukkan cara meminta daftar permintaan pembuatan akun untuk organisasi yang telah berhasil diselesaikan:

```
aws organizations list-create-account-status --states SUCCEEDED
```

Outputnya mencakup array objek dengan informasi tentang setiap permintaan.

```
{
  "CreateAccountStatuses": [
    {
      "AccountId": "4444444444444444",
      "AccountName": "Developer Test Account",
      "CompletedTimeStamp": 1481835812.143,
      "Id": "car-examplecreateaccountrequestid111",
      "RequestedTimeStamp": 1481829432.531,
      "State": "SUCCEEDED"
    }
  ]
}
```

Contoh 2: Untuk mengambil daftar permintaan pembuatan akun yang sedang berlangsung yang dibuat di organisasi saat ini

Contoh berikut mendapatkan daftar permintaan pembuatan akun yang sedang berlangsung untuk organisasi:

```
aws organizations list-create-account-status --states IN_PROGRESS
```

Outputnya mencakup array objek dengan informasi tentang setiap permintaan.

```
{
  "CreateAccountStatuses": [
    {
      "State": "IN_PROGRESS",
      "Id": "car-examplecreateaccountrequestid111",
      "RequestedTimeStamp": 1481829432.531,
      "AccountName": "Production Account"
    }
  ]
}
```

- Untuk detail API, lihat [ListCreateAccountStatus](#) di Referensi AWS CLI Perintah.

## list-handshakes-for-account

Contoh kode berikut menunjukkan cara menggunakan `list-handshakes-for-account`.

### AWS CLI

Untuk mengambil daftar jabat tangan yang dikirim ke akun

Contoh berikut menunjukkan cara mendapatkan daftar semua jabat tangan yang terkait dengan akun kredensial yang digunakan untuk memanggil operasi:

```
aws organizations list-handshakes-for-account
```

Outputnya mencakup daftar struktur jabat tangan dengan informasi tentang setiap jabat tangan termasuk statusnya saat ini:

```
{
  "Handshake": {
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
    "ExpirationTimestamp": 1482952459.257,
    "Id": "h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "juan@example.com",
        "Type": "EMAIL"
      }
    ],
    "RequestedTimestamp": 1481656459.257,
    "Resources": [
      {
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@amazon.com"
          }
        ]
      }
    ]
  }
}
```

```

        },
        {
            "Type": "MASTER_NAME",
            "Value": "Org Master Account"
        },
        {
            "Type": "ORGANIZATION_FEATURE_SET",
            "Value": "FULL"
        }
    ],
    "Type": "ORGANIZATION",
    "Value": "o-exampleorgid"
},
{
    "Type": "EMAIL",
    "Value": "juan@example.com"
}
],
"State": "OPEN"
}
}

```

- Untuk detail API, lihat [ListHandshakesForAccount](#) di Referensi AWS CLI Perintah.

## list-handshakes-for-organization

Contoh kode berikut menunjukkan cara menggunakan `list-handshakes-for-organization`.

### AWS CLI

Untuk mengambil daftar jabat tangan yang terkait dengan organisasi

Contoh berikut menunjukkan cara mendapatkan daftar jabat tangan yang terkait dengan organisasi saat ini:

```
aws organizations list-handshakes-for-organization
```

Outputnya menunjukkan dua jabat tangan. Yang pertama adalah undangan ke akun Juan dan menunjukkan keadaan TERBUKA. Yang kedua adalah undangan ke akun Anika dan menunjukkan status DITERIMA:

```
{
```

```

    "Handshakes": [
      {
        "Action": "INVITE",
        "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
        "ExpirationTimestamp": 1482952459.257,
        "Id": "h-examplehandshakeid111",
        "Parties": [
          {
            "Id": "o-exampleorgid",
            "Type": "ORGANIZATION"
          },
          {
            "Id": "juan@example.com",
            "Type": "EMAIL"
          }
        ],
        "RequestedTimestamp": 1481656459.257,
        "Resources": [
          {
            "Resources": [
              {
                "Type": "MASTER_EMAIL",
                "Value": "bill@amazon.com"
              },
              {
                "Type": "MASTER_NAME",
                "Value": "Org Master
Account"
              },
              {
                "Type":
"ORGANIZATION_FEATURE_SET",
                "Value": "FULL"
              }
            ],
            "Type": "ORGANIZATION",
            "Value": "o-exampleorgid"
          },
          {
            "Type": "EMAIL",
            "Value": "juan@example.com"
          }
        ]
      }
    ]
  }
}

```



```

        "Type": "NOTES",
        "Value": "This is an invitation to Juan's
account to join Bill's organization."
    }
  ],
  "State": "OPEN"
},
{
  "Action": "INVITE",
  "State": "ACCEPTED",
  "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
  "ExpirationTimestamp": 1.471797437427E9,
  "Id": "h-examplehandshakeid222",
  "Parties": [
    {
      "Id": "o-exampleorgid",
      "Type": "ORGANIZATION"
    },
    {
      "Id": "anika@example.com",
      "Type": "EMAIL"
    }
  ],
  "RequestedTimestamp": 1.469205437427E9,
  "Resources": [
    {
      "Resources": [
        {
          "Type": "MASTER_EMAIL",
          "Value": "bill@example.com"
        },
        {
          "Type": "MASTER_NAME",
          "Value": "Master Account"
        }
      ],
      "Type": "ORGANIZATION",
      "Value": "o-exampleorgid"
    },
    {
      "Type": "EMAIL",
      "Value": "anika@example.com"
    }
  ],

```

```

        {
            "Type": "NOTES",
            "Value": "This is an invitation to Anika's
account to join Bill's organization."
        }
    ]
}

```

- Untuk detail API, lihat [ListHandshakesForOrganization](#) di Referensi AWS CLI Perintah.

## list-organizational-units-for-parent

Contoh kode berikut menunjukkan cara menggunakan `list-organizational-units-for-parent`.

### AWS CLI

Untuk mengambil daftar OU di OU induk atau root

Contoh berikut menunjukkan kepada Anda cara mendapatkan daftar OU di root tertentu:

```
aws organizations list-organizational-units-for-parent --parent-id r-
examplerootid111
```

Output menunjukkan bahwa root yang ditentukan berisi dua OU dan menunjukkan detail masing-masing:

```

{
  "OrganizationalUnits": [
    {
      "Name": "AccountingDepartment",
      "Arn": "arn:aws:organizations::o-exampleorgid:ou/r-
examplerootid111/ou-examplerootid111-exampleoid111"
    },
    {
      "Name": "ProductionDepartment",
      "Arn": "arn:aws:organizations::o-exampleorgid:ou/r-
examplerootid111/ou-examplerootid111-exampleoid222"
    }
  ]
}

```

```
}
```

- Untuk detail API, lihat [ListOrganizationalUnitsForParent](#) di Referensi AWS CLI Perintah.

## list-parents

Contoh kode berikut menunjukkan cara menggunakan `list-parents`.

### AWS CLI

Untuk membuat daftar OU induk atau root untuk akun atau anak OU

Berikut contoh Anda bagaimana untuk daftar root atau induk OU yang berisi account 44444444444444:

```
aws organizations list-parents --child-id 44444444444444
```

Output menunjukkan bahwa akun yang ditentukan ada di OU dengan ID tertentu:

```
{
  "Parents": [
    {
      "Id": "ou-examplerootid111-exampleouid111",
      "Type": "ORGANIZATIONAL_UNIT"
    }
  ]
}
```

- Untuk detail API, lihat [ListParents](#) di Referensi AWS CLI Perintah.

## list-policies-for-target

Contoh kode berikut menunjukkan cara menggunakan `list-policies-for-target`.

### AWS CLI

Untuk mengambil daftar SCP yang dilampirkan langsung ke akun

Contoh berikut menunjukkan cara mendapatkan daftar semua kebijakan kontrol layanan (SCP), seperti yang ditentukan oleh parameter `Filter`, yang langsung dilampirkan ke akun:

```
aws organizations list-policies-for-target --filter SERVICE_CONTROL_POLICY --target-id 44444444444444
```

Output mencakup daftar struktur kebijakan dengan informasi ringkasan tentang kebijakan. Daftar ini tidak menyertakan kebijakan yang berlaku untuk akun karena warisan dari lokasinya dalam hierarki OU:

```
{
  "Policies": [
    {
      "Type": "SERVICE_CONTROL_POLICY",
      "Name": "AllowAllEC2Actions",
      "AwsManaged": false,
      "Id": "p-examplepolicyid222",
      "Arn": "arn:aws:organizations::o-exampleorgid:policy/service_control_policy/p-examplepolicyid222",
      "Description": "Enables account admins to delegate permissions for any EC2 actions to users and roles in their accounts."
    }
  ]
}
```

- Untuk detail API, lihat [ListPoliciesForTarget](#) di Referensi AWS CLI Perintah.

## list-policies

Contoh kode berikut menunjukkan cara menggunakan `list-policies`.

### AWS CLI

Untuk mengambil daftar semua kebijakan dalam organisasi dari jenis tertentu

Contoh berikut menunjukkan cara mendapatkan daftar SCP, seperti yang ditentukan oleh parameter filter:

```
aws organizations list-policies --filter SERVICE_CONTROL_POLICY
```

Outputnya mencakup daftar kebijakan dengan informasi ringkasan:

```
{
  "Policies": [
```

```

    {
        "Type": "SERVICE_CONTROL_POLICY",
        "Name": "AllowAllS3Actions",
        "AwsManaged": false,
        "Id": "p-examplepolicyid111",
        "Arn": "arn:aws:organizations::111111111111:policy/
service_control_policy/p-examplepolicyid111",
        "Description": "Enables account admins to delegate
permissions for any S3 actions to users and roles in their accounts."
    },
    {
        "Type": "SERVICE_CONTROL_POLICY",
        "Name": "AllowAllEC2Actions",
        "AwsManaged": false,
        "Id": "p-examplepolicyid222",
        "Arn": "arn:aws:organizations::111111111111:policy/
service_control_policy/p-examplepolicyid222",
        "Description": "Enables account admins to delegate
permissions for any EC2 actions to users and roles in their accounts."
    },
    {
        "AwsManaged": true,
        "Description": "Allows access to every operation",
        "Type": "SERVICE_CONTROL_POLICY",
        "Id": "p-FullAWSAccess",
        "Arn": "arn:aws:organizations::aws:policy/
service_control_policy/p-FullAWSAccess",
        "Name": "FullAWSAccess"
    }
]
}

```

- Untuk detail API, lihat [ListPolicies](#) di Referensi AWS CLI Perintah.

## list-roots

Contoh kode berikut menunjukkan cara menggunakan `list-roots`.

### AWS CLI

Untuk mengambil daftar akar dalam suatu organisasi

Contoh ini menunjukkan kepada Anda cara mendapatkan daftar root untuk suatu organisasi:

```
aws organizations list-roots
```

Outputnya mencakup daftar struktur root dengan informasi ringkasan:

```
{
  "Roots": [
    {
      "Name": "Root",
      "Arn": "arn:aws:organizations::111111111111:root/o-
exampleorgid/r-examplerootid111",
      "Id": "r-examplerootid111",
      "PolicyTypes": [
        {
          "Status": "ENABLED",
          "Type": "SERVICE_CONTROL_POLICY"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [ListRoots](#) di Referensi AWS CLI Perintah.

## list-targets-for-policy

Contoh kode berikut menunjukkan cara menggunakan `list-targets-for-policy`.

### AWS CLI

Untuk mengambil daftar akar, OU, dan akun yang dilampirkan kebijakan

Contoh berikut menunjukkan cara mendapatkan daftar root, OU, dan akun yang dilampirkan kebijakan yang ditentukan:

```
aws organizations list-targets-for-policy --policy-id p-FullAWSAccess
```

Outputnya mencakup daftar objek lampiran dengan informasi ringkasan tentang root, OU, dan akun yang dilampirkan kebijakan:

```
{
```

```

    "Targets": [
      {
        "Arn": "arn:aws:organizations::111111111111:root/o-
exampleorgid/r-examplerootid111",
        "Name": "Root",
        "TargetId": "r-examplerootid111",
        "Type": "ROOT"
      },
      {
        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333;",
        "Name": "Developer Test Account",
        "TargetId": "333333333333",
        "Type": "ACCOUNT"
      },
      {
        "Arn": "arn:aws:organizations::111111111111:ou/o-
exampleorgid/ou-examplerootid111-exampleouid111",
        "Name": "Accounting",
        "TargetId": "ou-examplerootid111-exampleouid111",
        "Type": "ORGANIZATIONAL_UNIT"
      }
    ]
  }
}

```

- Untuk detail API, lihat [ListTargetsForPolicy](#) di Referensi AWS CLI Perintah.

## move-account

Contoh kode berikut menunjukkan cara menggunakan `move-account`.

### AWS CLI

Untuk memindahkan akun antara root atau OU

Contoh berikut menunjukkan cara memindahkan akun master di organisasi dari root ke OU:

```
aws organizations move-account --account-id 333333333333 --source-parent-id r-
examplerootid111 --destination-parent-id ou-examplerootid111-exampleouid111
```

- Untuk detail API, lihat [MoveAccount](#) di Referensi AWS CLI Perintah.

## remove-account-from-organization

Contoh kode berikut menunjukkan cara menggunakan `remove-account-from-organization`.

### AWS CLI

Untuk menghapus akun dari organisasi sebagai akun utama

Contoh berikut menunjukkan cara menghapus akun dari organisasi:

```
aws organizations remove-account-from-organization --account-id 333333333333
```

- Untuk detail API, lihat [RemoveAccountFromOrganization](#) di Referensi AWS CLI Perintah.

## update-organizational-unit

Contoh kode berikut menunjukkan cara menggunakan `update-organizational-unit`.

### AWS CLI

Untuk mengganti nama OU

Contoh ini menunjukkan cara mengganti nama OU: Dalam contoh ini, OU diganti namanya menjadi "AccountingOu":

```
aws organizations update-organizational-unit --organizational-unit-id ou-examplerootid111-exampleouid111 --name AccountingOU
```

Output menunjukkan nama baru:

```
{
  "OrganizationalUnit": {
    "Id": "ou-examplerootid111-exampleouid111"
    "Name": "AccountingOU",
    "Arn": "arn:aws:organizations::111111111111:ou/o-exampleorgid/ou-examplerootid111-exampleouid111"
  }
}
```

- Untuk detail API, lihat [UpdateOrganizationalUnit](#) di Referensi AWS CLI Perintah.



## update-policy

Contoh kode berikut menunjukkan cara menggunakan `update-policy`.

### AWS CLI

Contoh 1: Untuk mengganti nama kebijakan

`update-policy` Contoh berikut mengganti nama kebijakan dan memberikan deskripsi baru.

```
aws organizations update-policy \  
  --policy-id p-examplepolicyid111 \  
  --name Renamed-Policy \  
  --description "This description replaces the original."
```

Output menunjukkan nama dan deskripsi baru.

```
{  
  "Policy": {  
    "Content": "{\n  \"Version\": \"2012-10-17\",  
  \"Statement\": {\n    \"Effect\": \"Allow\",  
    \"Action\": \"ec2:*\",  
    \"Resource\": \"*\"\n  }\n}",  
    "PolicySummary": {  
      "Id": "p-examplepolicyid111",  
      "AwsManaged": false,  
      "Arn": "arn:aws:organizations::111111111111:policy/o-exampleorgid/  
service_control_policy/p-examplepolicyid111",  
      "Description": "This description replaces the original.",  
      "Name": "Renamed-Policy",  
      "Type": "SERVICE_CONTROL_POLICY"  
    }  
  }  
}
```

Contoh 2: Untuk mengganti konten teks JSON kebijakan

Contoh berikut menunjukkan cara mengganti teks JSON SCP pada contoh sebelumnya dengan string teks kebijakan JSON baru yang memungkinkan S3 alih-alih EC2:

```
aws organizations update-policy \  
  --policy-id p-examplepolicyid111 \  
  --content "{\n  \"Version\": \"2012-10-17\",  
  \"Statement\": {\n    \"Effect\": \"Allow\",  
    \"Action\": \"s3:*\",  
    \"Resource\": \"*\"\n  }\n}"
```

```
--content "{\"Version\":\"2012-10-17\",\"Statement\":{\"Effect\":\"Allow\",
\"Action\":\"s3:*\",\"Resource\":\"*\"}}"
```

Output menunjukkan konten baru:

```
{
  "Policy": {
    "Content": "{ \"Version\": \"2012-10-17\", \"Statement\": { \"Effect\":
\"Allow\", \"Action\": \"s3:*\", \"Resource\": \"*\" } }",
    "PolicySummary": {
      "Arn": "arn:aws:organizations::111111111111:policy/o-exampleorgid/
service_control_policy/p-examplepolicyid111",
      "AwsManaged": false;
      "Description": "This description replaces the original.",
      "Id": "p-examplepolicyid111",
      "Name": "Renamed-Policy",
      "Type": "SERVICE_CONTROL_POLICY"
    }
  }
}
```

- Untuk detail API, lihat [UpdatePolicy](#) di Referensi AWS CLI Perintah.

## AWS Outposts contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Outposts.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

## Tindakan

### get-outpost-instance-types

Contoh kode berikut menunjukkan cara menggunakan `get-outpost-instance-types`.

#### AWS CLI

Untuk mendapatkan jenis instance di Outpost Anda

`get-outpost-instance-types` Contoh berikut mendapatkan jenis instance untuk Outpost yang ditentukan.

```
aws outposts get-outpost-instance-types \
  --outpost-id op-0ab23c4567EXAMPLE
```

Output:

```
{
  "InstanceTypes": [
    {
      "InstanceType": "c5d.large"
    },
    {
      "InstanceType": "i3en.24xlarge"
    },
    {
      "InstanceType": "m5d.large"
    },
    {
      "InstanceType": "r5d.large"
    }
  ],
  "OutpostId": "op-0ab23c4567EXAMPLE",
  "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Meluncurkan instance di Outpost Anda](#) di Panduan AWS Pengguna Outposts.

- Untuk detail API, lihat [GetOutpostInstanceTypes](#) di Referensi AWS CLI Perintah.

## get-outpost

Contoh kode berikut menunjukkan cara menggunakan `get-outpost`.

### AWS CLI

Untuk mendapatkan detail Outpost

`get-outpost` Contoh berikut menampilkan rincian untuk Outpost yang ditentukan.

```
aws outposts get-outpost \  
  --outpost-id op-0ab23c4567EXAMPLE
```

Output:

```
{  
  "Outpost": {  
    "OutpostId": "op-0ab23c4567EXAMPLE",  
    "OwnerId": "123456789012",  
    "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/  
op-0ab23c4567EXAMPLE",  
    "SiteId": "os-0ab12c3456EXAMPLE",  
    "Name": "EXAMPLE",  
    "LifecycleStatus": "ACTIVE",  
    "AvailabilityZone": "us-west-2a",  
    "AvailabilityZoneId": "usw2-az1",  
    "Tags": {}  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Outposts](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [GetOutpost](#) di Referensi AWS CLI Perintah.

## list-outposts

Contoh kode berikut menunjukkan cara menggunakan `list-outposts`.

### AWS CLI

Untuk daftar Outposts

`list-outposts` Contoh berikut mencantumkan Outposts di akun Anda AWS .

```
aws outposts list-outposts
```

Output:

```
{
  "Outposts": [
    {
      "OutpostId": "op-0ab23c4567EXAMPLE",
      "OwnerId": "123456789012",
      "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE",
      "SiteId": "os-0ab12c3456EXAMPLE",
      "Name": "EXAMPLE",
      "Description": "example",
      "LifecycleStatus": "ACTIVE",
      "AvailabilityZone": "us-west-2a",
      "AvailabilityZoneId": "usw2-az1",
      "Tags": {
        "Name": "EXAMPLE"
      }
    },
    {
      "OutpostId": "op-4fe3dc21baEXAMPLE",
      "OwnerId": "123456789012",
      "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-4fe3dc21baEXAMPLE",
      "SiteId": "os-0ab12c3456EXAMPLE",
      "Name": "EXAMPLE2",
      "LifecycleStatus": "ACTIVE",
      "AvailabilityZone": "us-west-2a",
      "AvailabilityZoneId": "usw2-az1",
      "Tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Outposts](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [ListOutposts](#) di Referensi AWS CLI Perintah.

## list-sites

Contoh kode berikut menunjukkan cara menggunakan `list-sites`.

### AWS CLI

Untuk daftar situs

`list-sites` Contoh berikut mencantumkan situs Outpost yang tersedia di AWS akun Anda.

```
aws outposts list-sites
```

Output:

```
{
  "Sites": [
    {
      "SiteId": "os-0ab12c3456EXAMPLE",
      "AccountId": "123456789012",
      "Name": "EXAMPLE",
      "Description": "example",
      "Tags": {}
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Outposts](#) di Panduan Pengguna AWS Outposts.

- Untuk detail API, lihat [ListSites](#) di Referensi AWS CLI Perintah.

## AWS Payment Cryptography contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Payment Cryptography.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **create-alias**

Contoh kode berikut menunjukkan cara menggunakan `create-alias`.

AWS CLI

Untuk membuat alias untuk kunci

`create-alias` Contoh berikut membuat alias untuk kunci.

```
aws payment-cryptography create-alias \  
  --alias-name alias/sampleAlias1 \  
  --key-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaif1lw2h
```

Output:

```
{  
  "Alias": {  
    "AliasName": "alias/sampleAlias1",  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
    kwapwa6qaif1lw2h"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Tentang alias](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [CreateAlias](#) di Referensi AWS CLI Perintah.

## **create-key**

Contoh kode berikut menunjukkan cara menggunakan `create-key`.

## AWS CLI

Untuk membuat kunci

`create-key` Contoh berikut menghasilkan kunci TDES 2KEY yang dapat Anda gunakan untuk menghasilkan dan memverifikasi nilai CVV/CVV2.

```
aws payment-cryptography create-key \  
  --exportable \  
  --key-attributes KeyAlgorithm=TDES_2KEY,  
  KeyUsage=TR31_C0_CARD_VERIFICATION_KEY,KeyClass=SYMMETRIC_KEY,  
  KeyModesOfUse={Generate=true,Verify=true}
```

Output:

```
{  
  "Key": {  
    "CreateTimestamp": "1686800690",  
    "Enabled": true,  
    "Exportable": true,  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
kwapwa6qaifllw2h",  
    "KeyAttributes": {  
      "KeyAlgorithm": "TDES_2KEY",  
      "KeyClass": "SYMMETRIC_KEY",  
      "KeyModesOfUse": {  
        "Decrypt": false,  
        "DeriveKey": false,  
        "Encrypt": false,  
        "Generate": true,  
        "NoRestrictions": false,  
        "Sign": false,  
        "Unwrap": false,  
        "Verify": true,  
        "Wrap": false  
      },  
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"  
    },  
    "KeyCheckValue": "F2E50F",  
    "KeyCheckValueAlgorithm": "ANSI_X9_24",  
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",  
    "KeyState": "CREATE_COMPLETE",  
    "UsageStartTimestamp": "1686800690"  
  },  
  "KeyCheckValue": "F2E50F",  
  "KeyCheckValueAlgorithm": "ANSI_X9_24",  
  "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",  
  "KeyState": "CREATE_COMPLETE",  
  "UsageStartTimestamp": "1686800690"
```



```
}  
}
```

Untuk informasi selengkapnya, lihat [Menghasilkan kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [CreateKey](#) di Referensi AWS CLI Perintah.

## delete-alias

Contoh kode berikut menunjukkan cara menggunakan `delete-alias`.

### AWS CLI

Untuk menghapus alias

`delete-alias` Contoh berikut menghapus alias. Itu tidak mempengaruhi kuncinya.

```
aws payment-cryptography delete-alias \  
  --alias-name alias/sampleAlias1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Tentang alias](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [DeleteAlias](#) di Referensi AWS CLI Perintah.

## delete-key

Contoh kode berikut menunjukkan cara menggunakan `delete-key`.

### AWS CLI

Untuk menghapus kunci

`delete-key` Contoh berikut menjadwalkan kunci untuk dihapus setelah 7 hari, yang merupakan periode tunggu default.

```
aws payment-cryptography delete-key \  
  --key-identifier arn:aws:payment-cryptography:us-west-2:123456789012:key/  
  kwapwa6qaiif1lw2h
```

**Output:**

```
{
  "Key": {
    "CreateTimestamp": "1686801198",
    "DeletePendingTimestamp": "1687405998",
    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_2KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
        "Generate": true,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
      },
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
    },
    "KeyCheckValue": "F2E50F",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "DELETE_PENDING",
    "UsageStartTimestamp": "1686801190"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [DeleteKey](#) di Referensi AWS CLI Perintah.

**export-key**

Contoh kode berikut menunjukkan cara menggunakan `export-key`.

## AWS CLI

Untuk mengekspor kunci

export-key Contoh berikut mengekspor kunci.

```
aws payment-cryptography export-key \
  --export-key-identifier arn:aws:payment-cryptography:us-west-2:123456789012:key/
lco3w6agsk7zgu2l \
  --key-material '{"Tr34KeyBlock": { \
    "CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-cryptography:us-
west-2:123456789012:key/ftobshq7pvioc5fx", \
    "ExportToken": "export-token-cu4lg26ofcziixny", \
    "KeyBlockFormat": "X9_TR34_2012", \
    "WrappingKeyCertificate": file://wrapping-key-certificate.pem } }'
```

Isi dari wrapping-key-certificate.pem:

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2VENDQXFXZ0F3SUJBZ01SQ1ZZS8xMXFUK2svVz1RUDJQOE1V
```

Output:

```
{
  "WrappedKey": {
    "KeyMaterial":
"308205A106092A864886F70D010702A08205923082058E020101310D300B06096086480165030402013082031F
    "WrappedKeyMaterialFormat": "TR34_KEY_BLOCK"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengekspor kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [ExportKey](#) di Referensi AWS CLI Perintah.

## get-alias

Contoh kode berikut menunjukkan cara menggunakan get-alias.

## AWS CLI

Untuk mendapatkan alias

`get-alias` Contoh berikut mengembalikan ARN dari kunci yang terkait dengan alias.

```
aws payment-cryptography get-alias \  
  --alias-name alias/sampleAlias1
```

Output:

```
{  
  "Alias": {  
    "AliasName": "alias/sampleAlias1",  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
kwapwa6qaiif1lw2h"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Tentang alias](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GetAlias](#) di Referensi AWS CLI Perintah.

## get-key

Contoh kode berikut menunjukkan cara menggunakan `get-key`.

### AWS CLI

Untuk mendapatkan metadata kunci

`get-key` Contoh berikut mengembalikan metadata kunci yang terkait dengan alias. Operasi ini tidak mengembalikan materi kriptografi.

```
aws payment-cryptography get-key \  
  --key-identifier alias/sampleAlias1
```

Output:

```
{  
  "Key": {  
    "CreateTimestamp": "1686800690",  
    "DeletePendingTimestamp": "1687405998",
```

```

    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_2KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
        "Generate": true,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
      },
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
    },
    "KeyCheckValue": "F2E50F",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "DELETE_PENDING",
    "UsageStartTimestamp": "1686801190"
  }
}

```

Untuk informasi selengkapnya, lihat [Mendapatkan kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GetKey](#) di Referensi AWS CLI Perintah.

## get-parameters-for-export

Contoh kode berikut menunjukkan cara menggunakan `get-parameters-for-export`.

### AWS CLI

Untuk menginisialisasi proses ekspor

`get-parameters-for-export` Contoh berikut menghasilkan key pair, menandatangani kunci, dan kemudian mengembalikan sertifikat dan root sertifikat.

```
aws payment-cryptography get-parameters-for-export \
  --signing-key-algorithm RSA_2048 \
  --key-material-type TR34_KEY_BLOCK
```

### Output:

```
{
  "ExportToken": "export-token-ep5cwyzone7oya53",
  "ParametersValidUntilTimestamp": "1687415640",
  "SigningKeyAlgorithm": "RSA_2048",
  "SigningKeyCertificate":

  "MIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
  b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAd
  BgkqhkiG9w0BCQEWEG5vb25lQGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
  MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
  VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
  b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAdBgkqhkiG9w0BCQEWEG5vb25lQGft
  YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
  21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
  rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
  Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
  nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
  FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
  NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
  "SigningKeyCertificateChain":
  "MIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
  b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAd
  BgkqhkiG9w0BCQEWEG5vb25lQGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
  MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
  VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
  b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAdBgkqhkiG9w0BCQEWEG5vb25lQGft
  YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
  21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
  rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
  Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
  nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
  FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
  NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}
```

Untuk informasi selengkapnya, lihat [Mengeksport kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GetParametersForExport](#) di Referensi AWS CLI Perintah.

## get-parameters-for-import

Contoh kode berikut menunjukkan cara menggunakan `get-parameters-for-import`.

### AWS CLI

Untuk menginisialisasi proses impor

`get-parameters-for-import` Contoh berikut menghasilkan key pair, menandatangani kunci, dan kemudian mengembalikan sertifikat dan root sertifikat.

```
aws payment-cryptography get-parameters-for-import \
  --key-material-type TR34_KEY_BLOCK \
  --wrapping-key-algorithm RSA_2048
```

Output:

```
{
  "ImportToken": "import-token-qgmafpa7nt2kfbb",
  "ParametersValidUntilTimestamp": "1687415640",
  "WrappingKeyAlgorithm": "RSA_2048",
  "WrappingKeyCertificate":
  "MIICiTCCAfICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
  b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAd
  BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
  MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
  VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
  b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGft
  YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
  21uUSfwfEvySwTC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
  rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
  Ibb30hjZncvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
  nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
  FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvjx79LjStB
  NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
  "WrappingKeyCertificateChain":
  "NIICiTCCAfICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
```

```
VVMxCzAJBgNVBAGTAldBMRAwDgYDVoQHEwdTZWF0dGx1MQ8wDQYDVoQKEwZBbWF6
b24xFDASBgNVBAwTC01BTSBDb25zb2x1MRIwEAYDVoQDEwLUZXN0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VoQHEwdTZWF0dGx1MQ8wDQYDVoQKEwZBbWF6b24xFDASBgNVBAwTC01BTSBDb25z
b2x1MRIwEAYDVoQDEwLUZXN0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIIJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}
```

Untuk informasi selengkapnya, lihat [Mengimpor kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GetParametersForImport](#) di Referensi AWS CLI Perintah.

## get-public-key-certificate

Contoh kode berikut menunjukkan cara menggunakan `get-public-key-certificate`.

### AWS CLI

Untuk mengembalikan kunci publik

`get-public-key-certificate` Contoh berikut mengembalikan bagian kunci publik dari sebuah key pair.

```
aws payment-cryptography get-public-key-certificate \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaifllw2h
```

Output:

```
{
  "KeyCertificate":
  "MIICiCCAfICCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVoQHEwdTZWF0dGx1MQ8wDQYDVoQKEwZBbWF6
  b24xFDASBgNVBAwTC01BTSBDb25zb2x1MRIwEAYDVoQDEwLUZXN0Q21sYWMxHzAd
  BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
```



```

MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAwTC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1ZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
"KeyCertificateChain":
"NIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAwTC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1ZAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAwTC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1ZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}

```

Untuk informasi selengkapnya, lihat [Mendapatkan kunci publik/sertifikat yang terkait dengan key pair di Panduan Pengguna Kriptografi AWS](#) Pembayaran.

- Untuk detail API, lihat [GetPublicKeyCertificate](#) di Referensi AWS CLI Perintah.

## import-key

Contoh kode berikut menunjukkan cara menggunakan `import-key`.

### AWS CLI

Untuk mengimpor kunci TR-34

`import-key` Contoh berikut mengimpor kunci TR-34.

```
aws payment-cryptography import-key \
```

```
--key-material='{ "Tr34KeyBlock": {" \
  CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-
cryptography:us-west-2:123456789012:key/rmm5wn2q564nijnjm", \
  "ImportToken": "import-token-5ott6ho5nts7bbc", \
  "KeyBlockFormat": "X9_TR34_2012", \
  "SigningKeyCertificate": file://signing-key-certificate.pem, \
  "WrappedKeyBlock": file://wrapped-key-block.pem }}}
```

Isi dari signing-key-certificate.pem:

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2RENDQXFTZ0F3SUJBZ01RYWVCK25IbE1WZU1PR1ZiNjU1Q2Jz
```

Isi dari wrapped-key-block.pem:

```
3082059806092A864886F70D010702A082058930820585020101310D300B06096086480165030402013082031606
```

Output:

```
{
  "Key": {
    "CreateTimestamp": "2023-06-09T16:56:27.621000-07:00",
    "Enabled": true,
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
bzmvgxgdg3sktwxd",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_2KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
        "Generate": true,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
      },
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
    },
    "KeyCheckValue": "D9B20E",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
```

```
    "KeyOrigin": "EXTERNAL",
    "KeyState": "CREATE_COMPLETE",
    "UsageStartTimestamp": "2023-06-09T16:56:27.621000-07:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengimpor kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [ImportKey](#) di Referensi AWS CLI Perintah.

## list-aliases

Contoh kode berikut menunjukkan cara menggunakan `list-aliases`.

### AWS CLI

Untuk mendapatkan daftar alias

`list-aliases` Contoh berikut menunjukkan semua alias di akun Anda di Wilayah ini.

```
aws payment-cryptography list-aliases
```

Output:

```
{
  "Aliases": [
    {
      "AliasName": "alias/sampleAlias1",
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/kwapwa6qaif1lw2h"
    },
    {
      "AliasName": "alias/sampleAlias2",
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/kwapwa6qaif1lw2h"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Tentang alias](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [ListAliases](#) di Referensi AWS CLI Perintah.

## list-keys

Contoh kode berikut menunjukkan cara menggunakan `list-keys`.

### AWS CLI

Untuk mendapatkan daftar kunci

`list-keys` Contoh berikut menunjukkan semua kunci di akun Anda di Wilayah ini.

```
aws payment-cryptography list-keys
```

Output:

```
{
  "Keys": [
    {
      "CreateTimestamp": "1666506840",
      "Enabled": false,
      "Exportable": true,
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/kwapwa6qaifllw2h",
      "KeyAttributes": {
        "KeyAlgorithm": "TDES_3KEY",
        "KeyClass": "SYMMETRIC_KEY",
        "KeyModesOfUse": {
          "Decrypt": true,
          "DeriveKey": false,
          "Encrypt": true,
          "Generate": false,
          "NoRestrictions": false,
          "Sign": false,
          "Unwrap": true,
          "Verify": false,
          "Wrap": true
        },
        "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
      },
      "KeyCheckValue": "369D",
      "KeyCheckValueAlgorithm": "ANSI_X9_24",
      "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    }
  ]
}
```

```
    "KeyState": "CREATE_COMPLETE",
    "UsageStopTimestamp": "1666938840"
  }
]
```

Untuk informasi selengkapnya, lihat [Daftar kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [ListKeys](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mendapatkan daftar tag untuk kunci

`list-tags-for-resource` Contoh berikut mendapatkan tag untuk kunci.

```
aws payment-cryptography list-tags-for-resource \
  --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaif1lw2h
```

Output:

```
{
  "Tags": [
    {
      "Key": "BIN",
      "Value": "20151120"
    },
    {
      "Key": "Project",
      "Value": "Production"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola tag kunci dengan operasi API](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## restore-key

Contoh kode berikut menunjukkan cara menggunakan `restore-key`.

### AWS CLI

Untuk mengembalikan kunci yang dijadwalkan untuk dihapus

`restore-key` Contoh berikut membatalkan penghapusan kunci.

```
aws payment-cryptography restore-key \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaifllw2h
```

### Output:

```
{  
  "Key": {  
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaifllw2h",  
    "KeyAttributes": {  
      "KeyUsage": "TR31_V2_VISA_PIN_VERIFICATION_KEY",  
      "KeyClass": "SYMMETRIC_KEY",  
      "KeyAlgorithm": "TDES_3KEY",  
      "KeyModesOfUse": {  
        "Encrypt": false,  
        "Decrypt": false,  
        "Wrap": false,  
        "Unwrap": false,  
        "Generate": true,  
        "Sign": false,  
        "Verify": true,  
        "DeriveKey": false,  
        "NoRestrictions": false  
      }  
    },  
    "KeyCheckValue": "",  
    "KeyCheckValueAlgorithm": "ANSI_X9_24",  
    "Enabled": false,  
    "Exportable": true,  
    "KeyState": "CREATE_COMPLETE",  
  }  
}
```

```
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "CreateTimestamp": "1686800690",
    "UsageStopTimestamp": "1687405998"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [RestoreKey](#) di Referensi AWS CLI Perintah.

## start-key-usage

Contoh kode berikut menunjukkan cara menggunakan `start-key-usage`.

### AWS CLI

Untuk mengaktifkan kunci

`start-key-usage` Contoh berikut memungkinkan kunci yang akan digunakan.

```
aws payment-cryptography start-key-usage \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaifllw2h
```

### Output:

```
{
  "Key": {
    "CreateTimestamp": "1686800690",
    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_3KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": true,
        "DeriveKey": false,
        "Encrypt": true,
        "Generate": false,
        "NoRestrictions": false,
```

```

        "Sign": false,
        "Unwrap": true,
        "Verify": false,
        "Wrap": true
    },
    "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
},
"KeyCheckValue": "369D",
"KeyCheckValueAlgorithm": "ANSI_X9_24",
"KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
"KeyState": "CREATE_COMPLETE",
"UsageStartTimestamp": "1686800690"
}
}

```

Untuk informasi selengkapnya, lihat [Mengaktifkan dan menonaktifkan kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [StartKeyUsage](#) di Referensi AWS CLI Perintah.

## stop-key-usage

Contoh kode berikut menunjukkan cara menggunakan `stop-key-usage`.

### AWS CLI

Untuk menonaktifkan kunci

`stop-key-usage` Contoh berikut menonaktifkan kunci.

```

aws payment-cryptography stop-key-usage \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaif1lw2h

```

Output:

```

{
  "Key": {
    "CreateTimestamp": "1686800690",
    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
    alsuwxug3pgy6xh",

```



```

    "KeyAttributes": {
      "KeyAlgorithm": "TDES_3KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": true,
        "DeriveKey": false,
        "Encrypt": true,
        "Generate": false,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": true,
        "Verify": false,
        "Wrap": true
      },
      "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
    },
    "KeyCheckValue": "369D",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "CREATE_COMPLETE",
    "UsageStartTimestamp": "1686800690"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengaktifkan dan menonaktifkan kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [StopKeyUsage](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai kunci

`tag-resource` Contoh berikut menandai kunci.

```

aws payment-cryptography tag-resource \
  --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaifllw2h \
  --tags Key=sampleTag,Value=sampleValue

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola tag kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari kunci

`untag-resource` Contoh berikut menghapus tag dari kunci.

```
aws payment-cryptography untag-resource \  
  --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaiif1lw2h \  
  --tag-keys sampleTag
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola tag kunci](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-alias**

Contoh kode berikut menunjukkan cara menggunakan `update-alias`.

### AWS CLI

Untuk memperbarui alias

`update-alias` Contoh berikut mengaitkan alias dengan kunci yang berbeda.

```
aws payment-cryptography update-alias \  
  --alias-name alias/sampleAlias1 \  
  --key-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  tqv5yij6wtxx64pi
```

## Output:

```
{
  "Alias": {
    "AliasName": "alias/sampleAlias1",
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
tqv5yij6wtxx64pi "
  }
}
```

Untuk informasi selengkapnya, lihat [Tentang alias](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [UpdateAlias](#) di Referensi AWS CLI Perintah.

## AWS Payment Cryptography Contoh Data Plane menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With AWS Payment Cryptography Data Plane.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **decrypt-data**

Contoh kode berikut menunjukkan cara menggunakan `decrypt-data`.

## AWS CLI

Untuk mendekripsi ciphertext

`decrypt-data` Contoh berikut mendekripsi data ciphertext menggunakan kunci simetris. Untuk operasi ini, kunci harus `KeyModesOfUse` disetel ke `Decrypt` dan `KeyUsage` disetel ke `TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`.

```
aws payment-cryptography-data decrypt-data \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaiif1lw2h \  
  --cipher-text 33612AB9D6929C3A828EB6030082B2BD \  
  --decryption-attributes 'Symmetric={Mode=CBC}'
```

Output:

```
{  
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaiif1lw2h",  
  "KeyCheckValue": "71D7AE",  
  "PlainText": "31323334313233343132333431323334"  
}
```

Untuk informasi selengkapnya, lihat [Mendekripsi data di Panduan Pengguna Kriptografi AWS Pembayaran](#).

- Untuk detail API, lihat [DecryptData](#) di Referensi AWS CLI Perintah.

## **encrypt-data**

Contoh kode berikut menunjukkan cara menggunakan `encrypt-data`.

## AWS CLI

Untuk mengenkripsi data

`encrypt-data` Contoh berikut mengenkripsi data plaintext menggunakan kunci simetris. Untuk operasi ini, kunci harus `KeyModesOfUse` disetel ke `Encrypt` dan `KeyUsage` disetel ke `TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`.

```
aws payment-cryptography-data encrypt-data \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaiif1lw2h \  
  --plaintext 31323334313233343132333431323334
```

```
--key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
--plain-text 31323334313233343132333431323334 \
--encryption-attributes 'Symmetric={Mode=CBC}'
```

Output:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
  "KeyCheckValue": "71D7AE",
  "CipherText": "33612AB9D6929C3A828EB6030082B2BD"
}
```

Untuk informasi selengkapnya, lihat [Mengenikmati data](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [EncryptData](#) di Referensi AWS CLI Perintah.

## generate-card-validation-data

Contoh kode berikut menunjukkan cara menggunakan `generate-card-validation-data`.

AWS CLI

Untuk menghasilkan CVV

`generate-card-validation-data` Contoh berikut menghasilkan CVV/CVV2.

```
aws payment-cryptography-data generate-card-validation-data \
--key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
--primary-account-number=171234567890123 \
--generation-attributes CardVerificationValue2={CardExpiryDate=0123}
```

Output:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
  "KeyCheckValue": "CADD1",
```

```
"ValidationData": "801"
}
```

Untuk informasi selengkapnya, lihat [Menghasilkan data kartu](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GenerateCardValidationData](#) di Referensi AWS CLI Perintah.

## generate-mac

Contoh kode berikut menunjukkan cara menggunakan `generate-mac`.

### AWS CLI

Untuk menghasilkan MAC

`generate-card-validation-data` Contoh berikut menghasilkan Hash Based Message Authentication Code (HMAC) untuk otentikasi data kartu menggunakan algoritma HMAC\_SHA256 dan kunci enkripsi HMAC. Kuncinya harus `KeyUsage` disetel ke `TR31_M7_HMAC_KEY` dan `KeyModesOfUse` ke `Generate`.

```
aws payment-cryptography-data generate-mac \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaif1lw2h \
  --message-data
"3b313038383439303031303733393431353d32343038323236303030373030303f33" \
  --generation-attributes Algorithm=HMAC_SHA256
```

Output:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaif1lw2h,
  "KeyCheckValue": "2976E7",
  "Mac": "ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C"
}
```

Untuk informasi selengkapnya, lihat [Menghasilkan MAC](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GenerateMac](#) di Referensi AWS CLI Perintah.

## generate-pin-data

Contoh kode berikut menunjukkan cara menggunakan `generate-pin-data`.

### AWS CLI

Untuk menghasilkan PIN

`generate-card-validation-data` Contoh berikut menghasilkan PIN acak baru menggunakan skema PIN Visa.

```
aws payment-cryptography-data generate-pin-data \
  --generation-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2 \
  --encryption-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
  --primary-account-number 171234567890123 \
  --pin-block-format ISO_FORMAT_0 \
  --generation-attributes VisaPin={PinVerificationKeyIndex=1}
```

Output:

```
{
  "GenerationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
  "GenerationKeyCheckValue": "7F2363",
  "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
  "EncryptionKeyCheckValue": "7CC9E2",
  "EncryptedPinBlock": "AC17DC148BDA645E",
  "PinData": {
    "VerificationValue": "5507"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghasilkan data PIN](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [GeneratePinData](#) di Referensi AWS CLI Perintah.

## re-encrypt-data

Contoh kode berikut menunjukkan cara menggunakan `re-encrypt-data`.

## AWS CLI

Untuk mengenkripsi ulang data dengan kunci yang berbeda

re-encrypt-data Contoh berikut mendekripsi teks sandi yang dienkripsi menggunakan kunci simetris AES dan mengenkripsi ulang menggunakan kunci Derived Unique Key Per Transaction (DUKPT).

```
aws payment-cryptography-data re-encrypt-data \
  --incoming-key-identifier arn:aws:payment-cryptography:us-
west-2:111122223333:key/hyv7ymboitd4vfy \
  --outgoing-key-identifier arn:aws:payment-cryptography:us-
west-2:111122223333:key/jl6ythkcvzesbxen \
  --cipher-text
4D2B0BDBA192D5AEFEAA5B3EC28E4A65383C313FFA25140101560F75FE1B99F27192A90980AB9334 \
  --incoming-encryption-attributes
"Dukt={Mode=ECB,KeySerialNumber=012345678911111}" \
  --outgoing-encryption-attributes '{"Symmetric": {"Mode": "ECB"}}'
```

Output:

```
{
  "CipherText":
  "F94959DA30EEFF0C035483C6067667CF6796E3C1AD28C2B61F9CFEB772A8DD41C0D6822931E0D3B1",
  "KeyArn": "arn:aws:payment-cryptography:us-west-2:111122223333:key/
jl6ythkcvzesbxen",
  "KeyCheckValue": "2E8CD9"
}
```

Untuk informasi selengkapnya, lihat [Mengekripsi dan mendekripsi data](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [ReEncryptData](#) di Referensi AWS CLI Perintah.

## translate-pin-data

Contoh kode berikut menunjukkan cara menggunakan translate-pin-data.

### AWS CLI

Untuk menerjemahkan data PIN



`translate-pin-data` Contoh berikut menerjemahkan PIN dari enkripsi PEK TDES menggunakan blok PIN ISO 0 ke Blok PIN AES ISO 4 menggunakan algoritma DUKPT.

```
aws payment-cryptography-data translate-pin-data \
  --encrypted-pin-block "AC17DC148BDA645E" \
  --incoming-translation-
attributes=IsoFormat0='{PrimaryAccountNumber=171234567890123}' \
  --incoming-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
  --outgoing-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/4pmyquwjs3yj4vwe \
  --outgoing-translation-attributes
IsoFormat4='{PrimaryAccountNumber=171234567890123}' \
  --outgoing-dukpt-attributes KeySerialNumber="FFFF9876543210E00008"
```

Output:

```
{
  "PinBlock": "1F4209C670E49F83E75CC72E81B787D9",
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt
  "KeyCheckValue": "7CC9E2"
}
```

Untuk informasi selengkapnya, lihat [Terjemahkan data PIN](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [TranslatePinData](#) di Referensi AWS CLI Perintah.

## **verify-auth-request-cryptogram**

Contoh kode berikut menunjukkan cara menggunakan `verify-auth-request-cryptogram`.

AWS CLI

Untuk memverifikasi permintaan autentikasi

`verify-auth-request-cryptogram` Contoh berikut memverifikasi Cryptogram Permintaan Otorisasi (ARQC).

```
aws payment-cryptography-data verify-auth-request-cryptogram \
```

```
--auth-request-cryptogram F6E1BD1E6037FB3E \
--auth-response-attributes '{"ArpcMethod1": {"AuthResponseCode": "1111"}}' \
--key-identifier arn:aws:payment-cryptography:us-west-2:111122223333:key/
pboipdfzd4mdklya \
--major-key-derivation-mode "EMV_OPTION_A" \
--session-key-derivation-attributes '{"EmvCommon":
{"ApplicationTransactionCounter": "1234", "PanSequenceNumber":
"01", "PrimaryAccountNumber": "471234567890123"}}' \
--transaction-data "123456789ABCDEF"
```

Output:

```
{
  "AuthResponseValue": "D899B8C6FBF971AA",
  "KeyArn": "arn:aws:payment-cryptography:us-west-2:111122223333:key/
pboipdfzd4mdklya",
  "KeyCheckValue": "985792"
}
```

Untuk informasi selengkapnya, lihat [kriptogram Verifikasi permintaan auth \(ARQC\) di Panduan Pengguna Kriptografi AWS](#) Pembayaran.

- Untuk detail API, lihat [VerifyAuthRequestCryptogram](#) di Referensi AWS CLI Perintah.

## verify-card-validation-data

Contoh kode berikut menunjukkan cara menggunakan `verify-card-validation-data`.

AWS CLI

Untuk memvalidasi CVV

`verify-card-validation-data` Contoh berikut memvalidasi CVV/CVV2 untuk PAN.

```
aws payment-cryptography-data verify-card-validation-data \
--key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi \
--primary-account-number=171234567890123 \
--verification-attributes CardVerificationValue2={CardExpiryDate=0123} \
--validation-data 801
```

Output:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
  tqv5yij6wtxx64pi",
  "KeyCheckValue": "CADD1"
}
```

Untuk informasi selengkapnya, lihat [Verifikasi data kartu](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [VerifyCardValidationData](#) di Referensi AWS CLI Perintah.

## verify-mac

Contoh kode berikut menunjukkan cara menggunakan `verify-mac`.

### AWS CLI

Untuk memverifikasi MAC

`verify-mac` Contoh berikut memverifikasi Hash Based Message Authentication Code (HMAC) untuk otentikasi data kartu menggunakan algoritma HMAC\_SHA256 dan kunci enkripsi HMAC.

```
aws payment-cryptography-data verify-mac \
  --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
  qnobl5lghrzunce6 \
  --message-data
  "3b343038383439303031303733393431353d32343038323236303030373030303f33" \
  --verification-attributes='Algorithm=HMAC_SHA256' \
  --mac ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C
```

Output:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
  qnobl5lghrzunce6,
  "KeyCheckValue": "2976E7",
}
```

Untuk informasi selengkapnya, lihat [Verifikasi MAC](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [VerifyMac](#) di Referensi AWS CLI Perintah.

## verify-pin-data

Contoh kode berikut menunjukkan cara menggunakan `verify-pin-data`.

### AWS CLI

Untuk memverifikasi PIN

`verify-pin-data` Contoh berikut memvalidasi PIN untuk PAN.

```
aws payment-cryptography-data verify-pin-data \
  --verification-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2 \
  --encryption-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
  --primary-account-number 171234567890123 \
  --pin-block-format ISO_FORMAT_0 \
  --verification-attributes
  VisaPin="{PinVerificationKeyIndex=1,VerificationValue=5507}" \
  --encrypted-pin-block AC17DC148BDA645E
```

Output:

```
{
  "VerificationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
  "VerificationKeyCheckValue": "7F2363",
  "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
  "EncryptionKeyCheckValue": "7CC9E2",
}
```

Untuk informasi selengkapnya, lihat [Verifikasi data PIN](#) di Panduan Pengguna Kriptografi AWS Pembayaran.

- Untuk detail API, lihat [VerifyPinData](#) di Referensi AWS CLI Perintah.

## Amazon Pinpoint contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface dengan Amazon Pinpoint.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-app**

Contoh kode berikut menunjukkan cara menggunakan `create-app`.

AWS CLI

Contoh 1: Untuk membuat aplikasi

`create-app` Contoh berikut membuat aplikasi baru (proyek).

```
aws pinpoint create-app \  
  --create-application-request Name=ExampleCorp
```

Output:

```
{  
  "ApplicationResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",  
    "Id": "810c7aab86d42fb2b56c8c966example",  
    "Name": "ExampleCorp",  
    "tags": {}  
  }  
}
```

Contoh 2: Untuk membuat aplikasi yang diberi tag

`create-app`Contoh berikut membuat aplikasi baru (proyek) dan mengaitkan tag (kunci dan nilai) dengan aplikasi.

```
aws pinpoint create-app \  
  --create-application-request Name=ExampleCorp,tags={"Stack"="Test"}
```

Output:

```
{  
  "ApplicationResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",  
    "Id": "810c7aab86d42fb2b56c8c966example",  
    "Name": "ExampleCorp",  
    "tags": {  
      "Stack": "Test"  
    }  
  }  
}
```

- Untuk detail API, lihat [CreateApp](#)di Referensi AWS CLI Perintah.

## **create-sms-template**

Contoh kode berikut menunjukkan cara menggunakan `create-sms-template`.

AWS CLI

Membuat template pesan untuk pesan yang dikirim melalui saluran SMS

`create-sms-template`Contoh berikut membuat template pesan SMS.

```
aws pinpoint create-sms-template \  
  --template-name TestTemplate \  
  --sms-template-request file://myfile.json \  
  --region us-east-1
```

Isi dari `myfile.json`:

```
{  
  "Body": "hello\n how are you?\n food is good",  
  "TemplateDescription": "Test SMS Template"
```

```
}
```

Output:

```
{
  "CreateTemplateMessageBody": {
    "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:templates/
TestTemplate/SMS",
    "Message": "Created",
    "RequestID": "8c36b17f-a0b0-400f-ac21-29e9b62a975d"
  }
}
```

Untuk informasi selengkapnya, lihat [templat pesan Amazon Pinpoint](#) di Panduan Pengguna Amazon Pinpoint.

- Untuk detail API, lihat [CreateSmsTemplate](#) di Referensi AWS CLI Perintah.

## delete-app

Contoh kode berikut menunjukkan cara menggunakan `delete-app`.

### AWS CLI

Untuk menghapus aplikasi

`delete-app` Contoh berikut menghapus aplikasi (proyek).

```
aws pinpoint delete-app \
  --application-id 810c7aab86d42fb2b56c8c966example
```

Output:

```
{
  "ApplicationResponse": {
    "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
    "Id": "810c7aab86d42fb2b56c8c966example",
    "Name": "ExampleCorp",
    "tags": {}
  }
}
```

- Untuk detail API, lihat [DeleteApp](#) di Referensi AWS CLI Perintah.

## get-apns-channel

Contoh kode berikut menunjukkan cara menggunakan `get-apns-channel`.

### AWS CLI

Untuk mengambil informasi tentang status dan pengaturan saluran APN untuk aplikasi

`get-apns-channel` Contoh berikut mengambil informasi tentang status dan pengaturan saluran APN untuk aplikasi.

```
aws pinpoint get-apns-channel \  
  --application-id 9ab1068eb0a6461c86cce7f27ce0efd7 \  
  --region us-east-1
```

Output:

```
{  
  "APNSChannelResponse": {  
    "ApplicationId": "9ab1068eb0a6461c86cce7f27ce0efd7",  
    "CreationDate": "2019-05-09T21:54:45.082Z",  
    "DefaultAuthenticationMethod": "CERTIFICATE",  
    "Enabled": true,  
    "HasCredential": true,  
    "HasTokenKey": false,  
    "Id": "apns",  
    "IsArchived": false,  
    "LastModifiedDate": "2019-05-09T22:04:01.067Z",  
    "Platform": "APNS",  
    "Version": 2  
  }  
}
```

- Untuk detail API, lihat [GetApnsChannel](#) di Referensi AWS CLI Perintah.

## get-app

Contoh kode berikut menunjukkan cara menggunakan `get-app`.



## AWS CLI

Untuk mengambil informasi tentang aplikasi (proyek)

`get-app` Contoh berikut mengambil informasi tentang aplikasi (proyek).

```
aws pinpoint get-app \  
  --application-id 810c7aab86d42fb2b56c8c966example \  
  --region us-east-1
```

Output:

```
{  
  "ApplicationResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-  
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",  
    "Id": "810c7aab86d42fb2b56c8c966example",  
    "Name": "ExampleCorp",  
    "tags": {  
      "Year": "2019",  
      "Stack": "Production"  
    }  
  }  
}
```

- Untuk detail API, lihat [GetApp](#) di Referensi AWS CLI Perintah.

## `get-apps`

Contoh kode berikut menunjukkan cara menggunakan `get-apps`.

## AWS CLI

Untuk mengambil informasi tentang semua aplikasi Anda

`get-apps` Contoh berikut mengambil informasi tentang semua aplikasi Anda (proyek).

```
aws pinpoint get-apps
```

Output:

```
{
```

```

"ApplicationsResponse": {
  "Item": [
    {
      "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
      "Id": "810c7aab86d42fb2b56c8c966example",
      "Name": "ExampleCorp",
      "tags": {
        "Year": "2019",
        "Stack": "Production"
      }
    },
    {
      "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/42d8c7eb0990a57ba1d5476a3example",
      "Id": "42d8c7eb0990a57ba1d5476a3example",
      "Name": "AnyCompany",
      "tags": {}
    },
    {
      "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/80f5c382b638ffe5ad12376bbexample",
      "Id": "80f5c382b638ffe5ad12376bbexample",
      "Name": "ExampleCorp_Test",
      "tags": {
        "Year": "2019",
        "Stack": "Test"
      }
    }
  ],
  "NextToken":
  "eyJJdcmVhdGlvbkRhdGUiOiIyMDE5LTA3LTE2VDE0jM40jUzljkwM1oiLCJBY2NvdW50SWQiOiI1MTIzOTcxODM4Nz"
}

```

Kehadiran nilai `NextToken` respons menunjukkan bahwa ada lebih banyak output yang tersedia. Panggil perintah lagi dan berikan nilai itu sebagai parameter `NextToken` input.

- Untuk detail API, lihat [GetApps](#) di Referensi AWS CLI Perintah.

## get-campaign

Contoh kode berikut menunjukkan cara menggunakan `get-campaign`.

## AWS CLI

Untuk mengambil informasi tentang status, konfigurasi, dan setelan lain untuk kampanye

`get-campaign` Contoh berikut mengambil informasi tentang status, konfigurasi, dan setelan lain untuk kampanye.

```
aws pinpoint get-campaign \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --campaign-id a1e63c6cc0eb43ed826ffcc3cc90b30d \  
  --region us-east-1
```

Output:

```
{  
  "CampaignResponse": {  
    "AdditionalTreatments": [],  
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
    "Arn": "arn:aws:mobiletargeting:us-  
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/campaigns/  
a1e63c6cc0eb43ed826ffcc3cc90b30d",  
    "CreationDate": "2019-10-08T18:40:16.581Z",  
    "Description": " ",  
    "HoldoutPercent": 0,  
    "Id": "a1e63c6cc0eb43ed826ffcc3cc90b30d",  
    "IsPaused": false,  
    "LastModifiedDate": "2019-10-08T18:40:16.581Z",  
    "Limits": {  
      "Daily": 0,  
      "MaximumDuration": 60,  
      "MessagesPerSecond": 50,  
      "Total": 0  
    },  
    "MessageConfiguration": {  
      "EmailMessage": {  
        "FromAddress": "sender@example.com",  
        "HtmlBody": "<!DOCTYPE html>\n <html lang=\"en\">\n <head>\n  
<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\" />\n</head>\n<body>Hello</body>\n</html>",  
        "Title": "PinpointDemo"  
      }  
    },  
    "Name": "MyCampaign",
```

```

    "Schedule": {
      "IsLocalTime": false,
      "StartTime": "IMMEDIATE",
      "Timezone": "utc"
    },
    "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
    "SegmentVersion": 1,
    "State": {
      "CampaignStatus": "COMPLETED"
    },
    "tags": {},
    "TemplateConfiguration": {},
    "Version": 1
  }
}

```

- Untuk detail API, lihat [GetCampaign](#) di Referensi AWS CLI Perintah.

## get-campaigns

Contoh kode berikut menunjukkan cara menggunakan `get-campaigns`.

### AWS CLI

Untuk mengambil informasi tentang status, konfigurasi, dan setelan lainnya untuk semua kampanye yang terkait dengan aplikasi

`get-campaigns` Contoh berikut mengambil informasi tentang status, konfigurasi, dan setelan lainnya untuk semua kampanye yang terkait dengan aplikasi.

```

aws pinpoint get-campaigns \
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
  --region us-east-1

```

### Output:

```

{
  "CampaignsResponse": {
    "Item": [
      {
        "AdditionalTreatments": [],
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",

```

```

    "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/
campaigns/7e1280344c8f4a9aa40a00b006fe44f1",
    "CreationDate": "2019-10-08T18:40:22.905Z",
    "Description": " ",
    "HoldoutPercent": 0,
    "Id": "7e1280344c8f4a9aa40a00b006fe44f1",
    "IsPaused": false,
    "LastModifiedDate": "2019-10-08T18:40:22.905Z",
    "Limits": {},
    "MessageConfiguration": {
      "EmailMessage": {
        "FromAddress": "sender@example.com",
        "HtmlBody": "<!DOCTYPE html>\n  <html lang=\"en
\n  <head>\n    <meta http-equiv=\"Content-Type\" content=\"text/html;
charset=utf-8\" />\n</head>\n<body>Hello</body>\n</html>",
        "Title": "PinpointDemo Test"
      }
    },
    "Name": "MyCampaign1",
    "Schedule": {
      "IsLocalTime": false,
      "QuietTime": {},
      "StartTime": "IMMEDIATE",
      "Timezone": "UTC"
    },
    "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
    "SegmentVersion": 1,
    "State": {
      "CampaignStatus": "COMPLETED"
    },
    "tags": {},
    "TemplateConfiguration": {},
    "Version": 1
  },
  {
    "AdditionalTreatments": [],
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
    "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/campaigns/
a1e63c6cc0eb43ed826ffcc3cc90b30d",
    "CreationDate": "2019-10-08T18:40:16.581Z",
    "Description": " ",
    "HoldoutPercent": 0,

```

```

    "Id": "a1e63c6cc0eb43ed826ffcc3cc90b30d",
    "IsPaused": false,
    "LastModifiedDate": "2019-10-08T18:40:16.581Z",
    "Limits": {
      "Daily": 0,
      "MaximumDuration": 60,
      "MessagesPerSecond": 50,
      "Total": 0
    },
    "MessageConfiguration": {
      "EmailMessage": {
        "FromAddress": "sender@example.com",
        "HtmlBody": "<!DOCTYPE html>\n  <html lang=\n
\n  <head>\n  <meta http-equiv=\n\"Content-Type\n
\n  </head>\n<body>Demo</body>\n</html>",
        "Title": "PinpointDemo"
      }
    },
    "Name": "MyCampaign2",
    "Schedule": {
      "IsLocalTime": false,
      "StartTime": "IMMEDIATE",
      "Timezone": "utc"
    },
    "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
    "SegmentVersion": 1,
    "State": {
      "CampaignStatus": "COMPLETED"
    },
    "tags": {},
    "TemplateConfiguration": {},
    "Version": 1
  }
]
}

```

- Untuk detail API, lihat [GetCampaigns](#) di Referensi AWS CLI Perintah.

## get-channels

Contoh kode berikut menunjukkan cara menggunakan `get-channels`.

## AWS CLI

Untuk mengambil informasi tentang riwayat dan status setiap saluran untuk aplikasi

`get-channels` Contoh berikut mengambil informasi tentang riwayat dan status setiap saluran untuk aplikasi.

```
aws pinpoint get-channels \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Output:

```
{  
  "ChannelsResponse": {  
    "Channels": {  
      "GCM": {  
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
        "CreationDate": "2019-10-08T18:28:23.182Z",  
        "Enabled": true,  
        "HasCredential": true,  
        "Id": "gcm",  
        "IsArchived": false,  
        "LastModifiedDate": "2019-10-08T18:28:23.182Z",  
        "Version": 1  
      },  
      "SMS": {  
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
        "CreationDate": "2019-10-08T18:39:18.511Z",  
        "Enabled": true,  
        "Id": "sms",  
        "IsArchived": false,  
        "LastModifiedDate": "2019-10-08T18:39:18.511Z",  
        "Version": 1  
      },  
      "EMAIL": {  
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
        "CreationDate": "2019-10-08T18:27:23.990Z",  
        "Enabled": true,  
        "Id": "email",  
        "IsArchived": false,  
        "LastModifiedDate": "2019-10-08T18:27:23.990Z",  
        "Version": 1  
      }  
    }  
  }  
}
```

```

    },
    "IN_APP": {
      "Enabled": true,
      "IsArchived": false,
      "Version": 0
    }
  }
}

```

- Untuk detail API, lihat [GetChannels](#) di Referensi AWS CLI Perintah.

## get-email-channel

Contoh kode berikut menunjukkan cara menggunakan `get-email-channel`.

### AWS CLI

Untuk mengambil informasi tentang status dan pengaturan saluran Email untuk aplikasi

`get-email-channel` Contoh berikut mengambil status dan pengaturan saluran Email untuk aplikasi.

```

aws pinpoint get-email-channel \
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
  --region us-east-1

```

### Output:

```

{
  "EmailChannelResponse": {
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
    "CreationDate": "2019-10-08T18:27:23.990Z",
    "Enabled": true,
    "FromAddress": "sender@example.com",
    "Id": "email",
    "Identity": "arn:aws:ses:us-east-1:AIDACKCEVSQ6C2EXAMPLE:identity/
sender@example.com",
    "IsArchived": false,
    "LastModifiedDate": "2019-10-08T18:27:23.990Z",
    "MessagesPerSecond": 1,
    "Platform": "EMAIL",

```



```
    "RoleArn": "arn:aws:iam::AIDACKCEVSQ6C2EXAMPLE:role/pinpoint-events",
    "Version": 1
  }
}
```

- Untuk detail API, lihat [GetEmailChannel](#) di Referensi AWS CLI Perintah.

## get-endpoint

Contoh kode berikut menunjukkan cara menggunakan `get-endpoint`.

### AWS CLI

Untuk mengambil informasi tentang pengaturan dan atribut titik akhir tertentu untuk aplikasi

`get-endpoint` Contoh berikut mengambil informasi tentang pengaturan dan atribut titik akhir tertentu untuk aplikasi.

```
aws pinpoint get-endpoint \
  --application-id 611e3e3cdd47474c9c1399a505665b91 \
  --endpoint-id testendpoint \
  --region us-east-1
```

### Output:

```
{
  "EndpointResponse": {
    "Address": "+11234567890",
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
    "Attributes": {},
    "ChannelType": "SMS",
    "CohortId": "63",
    "CreationDate": "2019-01-28T23:55:11.534Z",
    "EffectiveDate": "2021-08-06T00:04:51.763Z",
    "EndpointStatus": "ACTIVE",
    "Id": "testendpoint",
    "Location": {
      "Country": "USA"
    },
    "Metrics": {
      "SmsDelivered": 1.0
    },
  },
}
```

```

    "OptOut": "ALL",
    "RequestId": "a204b1f2-7e26-48a7-9c80-b49a2143489d",
    "User": {
      "UserAttributes": {
        "Age": [
          "24"
        ]
      },
      "UserId": "testuser"
    }
  }
}

```

- Untuk detail API, lihat [GetEndpoint](#) di Referensi AWS CLI Perintah.

## get-gcm-channel

Contoh kode berikut menunjukkan cara menggunakan `get-gcm-channel`.

### AWS CLI

Untuk mengambil informasi tentang status dan pengaturan saluran GCM untuk aplikasi

`get-gcm-channel` Contoh berikut mengambil informasi tentang status dan pengaturan saluran GCM untuk aplikasi.

```

aws pinpoint get-gcm-channel \
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
  --region us-east-1

```

### Output:

```

{
  "GCMChannelResponse": {
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
    "CreationDate": "2019-10-08T18:28:23.182Z",
    "Enabled": true,
    "HasCredential": true,
    "Id": "gcm",
    "IsArchived": false,
    "LastModifiedDate": "2019-10-08T18:28:23.182Z",
    "Platform": "GCM",
  }
}

```

```
    "Version": 1
  }
}
```

- Untuk detail API, lihat [GetGcmChannel](#) di Referensi AWS CLI Perintah.

## get-sms-channel

Contoh kode berikut menunjukkan cara menggunakan `get-sms-channel`.

### AWS CLI

Untuk mengambil informasi tentang status dan pengaturan saluran SMS untuk suatu aplikasi

`get-sms-channel` Contoh berikut mengambil status dan pengaturan saluran sms untuk aplikasi.

```
aws pinpoint get-sms-channel \
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
  --region us-east-1
```

### Output:

```
{
  "SMSChannelResponse": {
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
    "CreationDate": "2019-10-08T18:39:18.511Z",
    "Enabled": true,
    "Id": "sms",
    "IsArchived": false,
    "LastModifiedDate": "2019-10-08T18:39:18.511Z",
    "Platform": "SMS",
    "PromotionalMessagesPerSecond": 20,
    "TransactionalMessagesPerSecond": 20,
    "Version": 1
  }
}
```

- Untuk detail API, lihat [GetSmsChannel](#) di Referensi AWS CLI Perintah.

## get-sms-template

Contoh kode berikut menunjukkan cara menggunakan `get-sms-template`.

## AWS CLI

Mengambil konten dan pengaturan template pesan untuk pesan yang dikirim melalui saluran SMS

`get-sms-template` Contoh berikut mengambil konten dan pengaturan template pesan SMS.

```
aws pinpoint get-sms-template \  
  --template-name TestTemplate \  
  --region us-east-1
```

Output:

```
{  
  "SMSTemplateResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:templates/  
TestTemplate/SMS",  
    "Body": "hello\n how are you?\n food is good",  
    "CreationDate": "2023-06-20T21:37:30.124Z",  
    "LastModifiedDate": "2023-06-20T21:37:30.124Z",  
    "tags": {},  
    "TemplateDescription": "Test SMS Template",  
    "TemplateName": "TestTemplate",  
    "TemplateType": "SMS",  
    "Version": "1"  
  }  
}
```

Untuk informasi selengkapnya, lihat [templat pesan Amazon Pinpoint](#) di Panduan Pengguna Amazon Pinpoint.

- Untuk detail API, lihat [GetSmsTemplatedi](#) Referensi AWS CLI Perintah.

## `get-voice-channel`

Contoh kode berikut menunjukkan cara menggunakan `get-voice-channel`.

## AWS CLI

Untuk mengambil informasi tentang status dan pengaturan saluran suara untuk aplikasi

`get-voice-channel` Contoh berikut mengambil status dan pengaturan saluran suara untuk aplikasi.

```
aws pinpoint get-voice-channel \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Output:

```
{  
  "VoiceChannelResponse": {  
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
    "CreationDate": "2022-04-28T00:17:03.836Z",  
    "Enabled": true,  
    "Id": "voice",  
    "IsArchived": false,  
    "LastModifiedDate": "2022-04-28T00:17:03.836Z",  
    "Platform": "VOICE",  
    "Version": 1  
  }  
}
```

- Untuk detail API, lihat [GetVoiceChannel](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mengambil daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mengambil semua tag (nama kunci dan nilai) yang terkait dengan sumber daya yang ditentukan.

```
aws pinpoint list-tags-for-resource \  
  --resource-arn arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example
```

Output:

```
{  
  "TagsModel": {  
    "tags": {
```

```
        "Year": "2019",
        "Stack": "Production"
    }
}
```

Untuk informasi selengkapnya, lihat 'Menandai Sumber Daya Amazon Pinpoint' \_\_ di Panduan Pengembang Amazon Pinpoint. < <https://docs.aws.amazon.com/pinpoint/latest/developerguide/tagging-resources.html>>

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## phone-number-validate

Contoh kode berikut menunjukkan cara menggunakan `phone-number-validate`.

### AWS CLI

Mengambil informasi tentang nomor telepon

Berikut ini `phone-number-validate` mengambil informasi tentang nomor telepon.

```
aws pinpoint phone-number-validate \
  --number-validate-request PhoneNumber="+12065550142" \
  --region us-east-1
```

Output:

```
{
  "NumberValidateResponse": {
    "Carrier": "ExampleCorp Mobile",
    "City": "Seattle",
    "CleansedPhoneNumberE164": "+12065550142",
    "CleansedPhoneNumberNational": "2065550142",
    "Country": "United States",
    "CountryCodeIso2": "US",
    "CountryCodeNumeric": "1",
    "OriginalPhoneNumber": "+12065550142",
    "PhoneType": "MOBILE",
    "PhoneTypeCode": 0,
    "Timezone": "America/Los_Angeles",
    "ZipCode": "98101"
  }
}
```

```
}  
}
```

Untuk informasi selengkapnya, lihat [saluran SMS Amazon Pinpoint](#) di Panduan Pengguna Amazon Pinpoint.

- Untuk detail API, lihat [PhoneNumberValidated](#) di Referensi AWS CLI Perintah.

## send-messages

Contoh kode berikut menunjukkan cara menggunakan `send-messages`.

### AWS CLI

Untuk mengirim pesan SMS menggunakan titik akhir aplikasi

`send-messages` Contoh berikut mengirimkan pesan langsung untuk aplikasi dengan titik akhir.

```
aws pinpoint send-messages \  
  --application-id 611e3e3cdd47474c9c1399a505665b91 \  
  --message-request file://myfile.json \  
  --region us-west-2
```

Isi dari `myfile.json`:

```
{  
  "MessageConfiguration": {  
    "SMSMessage": {  
      "Body": "hello, how are you?"  
    }  
  },  
  "Endpoints": {  
    "testendpoint": {}  
  }  
}
```

Output:

```
{  
  "MessageResponse": {  
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
```

```

    "EndpointResult": {
      "testendpoint": {
        "Address": "+12345678900",
        "DeliveryStatus": "SUCCESSFUL",
        "MessageId": "itnuqhai5alf1n6ahv3udc05n7hhddr6gb3lq6g0",
        "StatusCode": 200,
        "StatusMessage": "MessageId:
itnuqhai5alf1n6ahv3udc05n7hhddr6gb3lq6g0"
      }
    },
    "RequestId": "c7e23264-04b2-4a46-b800-d24923f74753"
  }
}

```

Untuk informasi selengkapnya, lihat [saluran SMS Amazon Pinpoint](#) di Panduan Pengguna Amazon Pinpoint.

- Untuk detail API, lihat [SendMessages](#) di Referensi AWS CLI Perintah.

## send-users-messages

Contoh kode berikut menunjukkan cara menggunakan `send-users-messages`.

### AWS CLI

Untuk mengirim pesan SMS untuk pengguna aplikasi

`send-users-messages` Contoh berikut mengirimkan pesan langsung untuk pengguna aplikasi.

```

aws pinpoint send-users-messages \
  --application-id 611e3e3cdd47474c9c1399a505665b91 \
  --send-users-message-request file://myfile.json \
  --region us-west-2

```

Isi dari `myfile.json`:

```

{
  "MessageConfiguration": {
    "SMSMessage": {
      "Body": "hello, how are you?"
    }
  },
}

```



```

    "Users": {
      "testuser": {}
    }
  }
}

```

Output:

```

{
  "SendUsersMessageResponse": {
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
    "RequestId": "e0b12cf5-2359-11e9-bb0b-d5fb91876b25",
    "Result": {
      "testuser": {
        "testuserendpoint": {
          "DeliveryStatus": "SUCCESSFUL",
          "MessageId": "7qu4hk5bqhda3i7i2n4pjf98qcu8b7p45ifsmo0",
          "StatusCode": 200,
          "StatusMessage": "MessageId:
7qu4hk5bqhda3i7i2n4pjf98qcu8b7p45ifsmo0",
          "Address": "+12345678900"
        }
      }
    }
  }
}

```

Untuk informasi selengkapnya, lihat [saluran SMS Amazon Pinpoint](#) di Panduan Pengguna Amazon Pinpoint.

- Untuk detail API, lihat [SendUsersMessages](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber daya

Contoh berikut menambahkan dua tag (nama kunci dan nilai) ke sumber daya.

```
aws pinpoint list-tags-for-resource \
```

```
--resource-arn arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \  
--tags-model tags={Stack=Production,Year=2019}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat 'Menandai Sumber Daya Amazon Pinpoint '\_\_\_ di Panduan Pengembang Amazon Pinpoint. < <https://docs.aws.amazon.com/pinpoint/latest/developerguide/tagging-resources.html>>

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Contoh 1: Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag tertentu (nama kunci dan nilai) dari sumber daya.

```
aws pinpoint untag-resource \  
  --resource-arn arn:aws:mobiletargeting:us-west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \  
  --tag-keys Year
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menghapus beberapa tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag tertentu (nama kunci dan nilai-nilai) dari sumber daya.

```
aws pinpoint untag-resource \  
  --resource-arn arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \  
  --tag-keys Year Stack
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat 'Menandai Sumber Daya Amazon Pinpoint' \_\_ di Panduan Pengembang Amazon Pinpoint. < <https://docs.aws.amazon.com/pinpoint/latest/developerguide/tagging-resources.html>>

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-sms-channel

Contoh kode berikut menunjukkan cara menggunakan `update-sms-channel`.

### AWS CLI

Untuk mengaktifkan saluran SMS atau memperbarui status dan pengaturan saluran SMS untuk suatu aplikasi.

`update-sms-channel` Contoh berikut memungkinkan saluran SMS untuk saluran SMS untuk suatu aplikasi.

```
aws pinpoint update-sms-channel \  
  --application-id 611e3e3cdd47474c9c1399a505665b91 \  
  --sms-channel-request Enabled=true \  
  --region us-west-2
```

### Output:

```
{  
  "SMSChannelResponse": {  
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",  
    "CreationDate": "2019-01-28T23:25:25.224Z",  
    "Enabled": true,  
    "Id": "sms",  
    "IsArchived": false,  
    "LastModifiedDate": "2023-05-18T23:22:50.977Z",  
    "Platform": "SMS",  
    "PromotionalMessagesPerSecond": 20,  
    "TransactionalMessagesPerSecond": 20,  
    "Version": 3  
  }  
}
```

Untuk informasi selengkapnya, lihat [saluran SMS Amazon Pinpoint](#) di Panduan Pengguna Amazon Pinpoint.

- Untuk detail API, lihat [UpdateSmsChannel](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Polly menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon Polly. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-lexicon**

Contoh kode berikut menunjukkan cara menggunakan `delete-lexicon`.

AWS CLI

Untuk menghapus leksikon

`delete-lexicon` Contoh berikut menghapus leksikon yang ditentukan.

```
aws polly delete-lexicon \  
  --name w3c
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan DeleteLexicon operasi](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [DeleteLexicon](#) di Referensi AWS CLI Perintah.

## get-lexicon

Contoh kode berikut menunjukkan cara menggunakan `get-lexicon`.

### AWS CLI

Untuk mengambil konten leksikon

`get-lexicon` Contoh berikut mengambil isi dari leksikon pengucapan yang ditentukan.

```
aws polly get-lexicon \  
  --name w3c
```

Output:

```
{  
  "Lexicon": {  
    "Content": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<lexicon version=  
\"1.0\" \n      xmlns=      \"http://www.w3.org/2005/01/pronunciation-lexicon  
\" \n      xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" \n      xsi:schemaLocation=\"http://www.w3.org/2005/01/pronunciation-lexicon \n      http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd\" \n      alphabet=\"ipa\" \n      xml:lang=\"en-US\">\n  <lexeme>\n    <grapheme>W3C</  
grapheme>\n    <alias>World Wide Web Consortium</alias>\n  </lexeme>\n</lexicon>  
\n",  
    "Name": "w3c"  
  },  
  "LexiconAttributes": {  
    "Alphabet": "ipa",  
    "LanguageCode": "en-US",  
    "LastModified": 1603908910.99,  
    "LexiconArn": "arn:aws:polly:us-west-2:880185128111:lexicon/w3c",  
    "LexemesCount": 1,  
    "Size": 492  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan GetLexicon operasi](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [GetLexicon](#) di Referensi AWS CLI Perintah.

## get-speech-synthesis-task

Contoh kode berikut menunjukkan cara menggunakan `get-speech-synthesis-task`.

### AWS CLI

Untuk mendapatkan informasi tentang tugas sintesis ucapan

`get-speech-synthesis-task` Contoh berikut mengambil informasi tentang tugas sintesis ucapan yang ditentukan.

```
aws polly get-speech-synthesis-task \
  --task-id 70b61c0f-57ce-4715-a247-cae8729dcce9
```

### Output:

```
{
  "SynthesisTask": {
    "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
    "TaskStatus": "completed",
    "OutputUri": "https://s3.us-west-2.amazonaws.com/my-s3-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
    "CreationTime": 1603911042.689,
    "RequestCharacters": 1311,
    "OutputFormat": "mp3",
    "TextType": "text",
    "VoiceId": "Joanna"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat file audio panjang](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [GetSpeechSynthesisTask](#) di Referensi AWS CLI Perintah.

## list-lexicons

Contoh kode berikut menunjukkan cara menggunakan `list-lexicons`.

## AWS CLI

Untuk membuat daftar leksikon Anda

`list-lexicons` Contoh berikut mencantumkan leksikon pengucapan Anda.

```
aws polly list-lexicons
```

Output:

```
{
  "Lexicons": [
    {
      "Name": "w3c",
      "Attributes": {
        "Alphabet": "ipa",
        "LanguageCode": "en-US",
        "LastModified": 1603908910.99,
        "LexiconArn": "arn:aws:polly:us-east-2:123456789012:lexicon/w3c",
        "LexemesCount": 1,
        "Size": 492
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menggunakan ListLexicons operasi](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [ListLexicons](#) di Referensi AWS CLI Perintah.

## **list-speech-synthesis-tasks**

Contoh kode berikut menunjukkan cara menggunakan `list-speech-synthesis-tasks`.

### AWS CLI

Untuk membuat daftar tugas sintesis ucapan Anda

`list-speech-synthesis-tasks` Contoh berikut mencantumkan tugas sintesis ucapan Anda.

```
aws polly list-speech-synthesis-tasks
```

**Output:**

```
{
  "SynthesisTasks": [
    {
      "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
      "TaskStatus": "completed",
      "OutputUri": "https://s3.us-west-2.amazonaws.com/my-s3-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
      "CreationTime": 1603911042.689,
      "RequestCharacters": 1311,
      "OutputFormat": "mp3",
      "TextType": "text",
      "VoiceId": "Joanna"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat file audio panjang](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [ListSpeechSynthesisTasks](#) di Referensi AWS CLI Perintah.

**put-lexicon**

Contoh kode berikut menunjukkan cara menggunakan `put-lexicon`.

**AWS CLI**

Untuk menyimpan leksikon

`put-lexicon` Contoh berikut menyimpan leksikon pengucapan yang ditentukan. `example.pls` file tersebut menentukan leksikon yang sesuai dengan PLS W3C.

```
aws polly put-lexicon \
  --name w3c \
  --content file://example.pls
```

Isi dari `example.pls`

```
{
```



```
<?xml version="1.0" encoding="UTF-8"?>
<lexicon version="1.0"
  xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon
    http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"
  alphabet="ipa"
  xml:lang="en-US">
  <lexeme>
    <grapheme>W3C</grapheme>
    <alias>World Wide Web Consortium</alias>
  </lexeme>
</lexicon>
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan PutLexicon operasi](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [PutLexicon](#) di Referensi AWS CLI Perintah.

## start-speech-synthesis-task

Contoh kode berikut menunjukkan cara menggunakan `start-speech-synthesis-task`.

### AWS CLI

Untuk mensintesis teks

`start-speech-synthesis-task` Contoh berikut mensintesis teks `text_file.txt` dan menyimpan file MP3 yang dihasilkan dalam bucket yang ditentukan.

```
aws polly start-speech-synthesis-task \
  --output-format mp3 \
  --output-s3-bucket-name my-s3-bucket \
  --text file://text_file.txt \
  --voice-id Joanna
```

Output:

```
{
```

```
"SynthesisTask": {
  "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
  "TaskStatus": "scheduled",
  "OutputUri": "https://s3.us-east-2.amazonaws.com/my-s3-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
  "CreationTime": 1603911042.689,
  "RequestCharacters": 1311,
  "OutputFormat": "mp3",
  "TextType": "text",
  "VoiceId": "Joanna"
}
```

Untuk informasi selengkapnya, lihat [Membuat file audio panjang](#) di Panduan Pengembang Amazon Polly.

- Untuk detail API, lihat [StartSpeechSynthesisTask](#) di Referensi AWS CLI Perintah.

## Daftar Harga AWS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Daftar Harga AWS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-services**

Contoh kode berikut menunjukkan cara menggunakan `describe-services`.

## AWS CLI

Untuk mengambil metadata layanan

Contoh ini mengambil metadata untuk kode layanan Amazon EC2.

Perintah:

```
aws pricing describe-services --service-code AmazonEC2 --format-version aws_v1 --max-items 1
```

Output:

```
{
  "Services": [
    {
      "ServiceCode": "AmazonEC2",
      "AttributeNames": [
        "volumeType",
        "maxIopsvolume",
        "instance",
        "instanceCapacity10xlarge",
        "locationType",
        "instanceFamily",
        "operatingSystem",
        "clockSpeed",
        "LeaseContractLength",
        "ecu",
        "networkPerformance",
        "instanceCapacity8xlarge",
        "group",
        "maxThroughputvolume",
        "gpuMemory",
        "ebsOptimized",
        "elasticGpuType",
        "maxVolumeSize",
        "gpu",
        "processorFeatures",
        "intelAvxAvailable",
        "instanceCapacity4xlarge",
        "servicecode",
        "groupDescription",
        "processorArchitecture",
```

```
    "physicalCores",
    "productFamily",
    "enhancedNetworkingSupported",
    "intelTurboAvailable",
    "memory",
    "dedicatedEbsThroughput",
    "vcpu",
    "OfferingClass",
    "instanceCapacityLarge",
    "capacitystatus",
    "termType",
    "storage",
    "intelAvx2Available",
    "storageMedia",
    "physicalProcessor",
    "provisioned",
    "servicename",
    "PurchaseOption",
    "instanceCapacity18xlarge",
    "instanceType",
    "tenancy",
    "usagetype",
    "normalizationSizeFactor",
    "instanceCapacity2xlarge",
    "instanceCapacity16xlarge",
    "maxIopsBurstPerformance",
    "instanceCapacity12xlarge",
    "instanceCapacity32xlarge",
    "instanceCapacityXlarge",
    "licenseModel",
    "currentGeneration",
    "preInstalledSw",
    "location",
    "instanceCapacity24xlarge",
    "instanceCapacity9xlarge",
    "instanceCapacityMedium",
    "operation"
  ]
}
],
"FormatVersion": "aws_v1"
}
```

- Untuk detail API, lihat [DescribeServices](#) di Referensi AWS CLI Perintah.

## get-attribute-values

Contoh kode berikut menunjukkan cara menggunakan `get-attribute-values`.

### AWS CLI

Untuk mengambil daftar nilai atribut

`get-attribute-values` Contoh berikut mengambil daftar nilai yang tersedia untuk atribut yang diberikan.

```
aws pricing get-attribute-values \  
  --service-code AmazonEC2 \  
  --attribute-name volumeType \  
  --max-items 2
```

Output:

```
{  
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ==",  
  "AttributeValues": [  
    {  
      "Value": "Cold HDD"  
    },  
    {  
      "Value": "General Purpose"  
    }  
  ]  
}
```

- Untuk detail API, lihat [GetAttributeValues](#) di Referensi AWS CLI Perintah.

## get-products

Contoh kode berikut menunjukkan cara menggunakan `get-products`.

### AWS CLI

Untuk mengambil daftar produk

Contoh ini mengambil daftar produk yang cocok dengan kriteria yang diberikan.

Perintah:

```
aws pricing get-products --filters file:///filters.json --format-version aws_v1 --
max-results 1 --service-code AmazonEC2
```

filters.json:

```
[
  {
    "Type": "TERM_MATCH",
    "Field": "ServiceCode",
    "Value": "AmazonEC2"
  },
  {
    "Type": "TERM_MATCH",
    "Field": "volumeType",
    "Value": "Provisioned IOPS"
  }
]
```

Output:

```
{
  "FormatVersion": "aws_v1",
  "NextToken": "WGDY7ko8fQXdlauZVdasFQ==:RVSagyIFn770XQ0zdUIc09BY6ucBG9itXAZGZF/
zioUz0sUKh6PCcPwa0yPZRiMePb986TeoKYB9155fw/
CyoMq5ymnGmT1Vj39T1jbbAlhcqnVfTmPIilx8Uy5bdDaBYy/e/20fw9Edzsykbs8LTBUmbiDQ
+BBds5yeI9AQkUepRuKk3aEahFPxJ55kx/zk",
  "PriceList": [
    {"productFamily": "Storage", "attributes": {"storageMedia": "SSD-backed", "maxThroughputVolume": "320 MB/sec", "volumeType": "Provisioned IOPS", "maxIopsVolume": "20000", "serviceCode": "AmazonEC2", "usageType": "APS1-EBS:VolumeUsage.piops", "locationType": "AWS Region", "location": "Asia Pacific (Singapore)", "serviceName": "Amazon Elastic Compute Cloud", "maxVolumeSize": "16 TiB", "operation": "", "sku": "3MKHN58N7RDDVGKJ"}, "serviceCode": "AmazonEC2", "terms": {"OnDemand": {"3MKHN58N7RDDVGKJ.JRTCKXETXF": {"priceDimensions": {"3MKHN58N7RDDVGKJ.JRTCKXETXF.6YS6EN2CT7": {"unit": "GB-Mo", "endRange": "Inf", "description": "$0.138 per GB-month of Provisioned IOPS SSD (io1) provisioned storage - Asia Pacific (Singapore)", "appliesTo": [], "rateCode": "3MKHN58N7RDDVGKJ.JRTCKXETXF.6YS6EN2CT7", "beginRange": "0", "pricePerUnit": {"USD": "0.1380000000"}}}}, "sku": "3MKHN58N7RDDVGKJ", "effectiveDate": "2018-08-01T00:00:00Z", "offerTermCode": "JRTCKXETXF", "termAttributes": {}}, "version": "20180808005701", "publicationDate": "2018-08-08T00:57:01Z"}
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [GetProducts](#) di Referensi AWS CLI Perintah.

## AWS Private CA contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Private CA.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-certificate-authority-audit-report**

Contoh kode berikut menunjukkan cara menggunakan `create-certificate-authority-audit-report`.

### AWS CLI

Untuk membuat laporan audit otoritas sertifikat

`create-certificate-authority-audit-report` Perintah berikut membuat laporan audit untuk CA pribadi yang diidentifikasi oleh ARN.

```
aws acm-pca create-certificate-authority-audit-report --certificate-
authority-arn arn:aws:acm-pca:us-east-1:accountid:certificate-
authority/12345678-1234-1234-1234-123456789012 --s3-bucket-name your-bucket-name --
audit-report-response-format JSON
```

- Untuk detail API, lihat [CreateCertificateAuthorityAuditReport](#) di Referensi AWS CLI Perintah.

## **create-certificate-authority**

Contoh kode berikut menunjukkan cara menggunakan `create-certificate-authority`.

### AWS CLI

Untuk membuat otoritas sertifikat pribadi

`create-certificate-authority` Perintah berikut membuat otoritas sertifikat pribadi di AWS akun Anda.

```
aws acm-pca create-certificate-authority --certificate-authority-configuration
file:///C:\ca_config.txt --revocation-configuration file:///C:\revoke_config.txt --
certificate-authority-type "SUBORDINATE" --idempotency-token 98256344
```

- Untuk detail API, lihat [CreateCertificateAuthority](#) di Referensi AWS CLI Perintah.

## **delete-certificate-authority**

Contoh kode berikut menunjukkan cara menggunakan `delete-certificate-authority`.

### AWS CLI

Untuk menghapus otoritas sertifikat pribadi

`delete-certificate-authority` Perintah berikut menghapus otoritas sertifikat yang diidentifikasi oleh ARN.

```
aws acm-pca delete-certificate-authority --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012
```

- Untuk detail API, lihat [DeleteCertificateAuthority](#) di Referensi AWS CLI Perintah.

## **describe-certificate-authority-audit-report**

Contoh kode berikut menunjukkan cara menggunakan `describe-certificate-authority-audit-report`.



## AWS CLI

Untuk menggambarkan laporan audit untuk otoritas sertifikat

`describe-certificate-authority-audit-report` Perintah berikut mencantumkan informasi tentang laporan audit yang ditentukan untuk CA yang diidentifikasi oleh ARN.

```
aws acm-pca describe-certificate-authority-audit-report --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/99999999-8888-7777-6666-555555555555 --audit-report-id
11111111-2222-3333-4444-555555555555
```

- Untuk detail API, lihat [DescribeCertificateAuthorityAuditReport](#) di Referensi AWS CLI Perintah.

## `describe-certificate-authority`

Contoh kode berikut menunjukkan cara menggunakan `describe-certificate-authority`.

## AWS CLI

Untuk menggambarkan otoritas sertifikat swasta

`describe-certificate-authority` Perintah berikut mencantumkan informasi tentang CA pribadi yang diidentifikasi oleh ARN.

```
aws acm-pca describe-certificate-authority --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012
```

- Untuk detail API, lihat [DescribeCertificateAuthority](#) di Referensi AWS CLI Perintah.

## `get-certificate-authority-certificate`

Contoh kode berikut menunjukkan cara menggunakan `get-certificate-authority-certificate`.

## AWS CLI

Untuk mengambil sertifikat otoritas sertifikat (CA)

`get-certificate-authority-certificate` Perintah berikut mengambil sertifikat dan rantai sertifikat untuk CA pribadi yang ditentukan oleh ARN.

```
aws acm-pca get-certificate-authority-certificate --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --output text
```

- Untuk detail API, lihat [GetCertificateAuthorityCertificate](#) di Referensi AWS CLI Perintah.

## get-certificate-authority-csr

Contoh kode berikut menunjukkan cara menggunakan `get-certificate-authority-csr`.

### AWS CLI

Untuk mengambil permintaan penandatanganan sertifikat untuk otoritas sertifikat

`get-certificate-authority-csr` Perintah berikut mengambil CSR untuk CA pribadi yang ditentukan oleh ARN.

```
aws acm-pca get-certificate-authority-csr --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --output text
```

- Untuk detail API, lihat [GetCertificateAuthorityCsr](#) di Referensi AWS CLI Perintah.

## get-certificate

Contoh kode berikut menunjukkan cara menggunakan `get-certificate`.

### AWS CLI

Untuk mengambil sertifikat yang dikeluarkan

`get-certificate` Contoh berikut mengambil sertifikat dari CA pribadi yang ditentukan.

```
aws acm-pca get-certificate \
  --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 \
  --certificate-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012/
certificate/6707447683a9b7f4055627ffd55cebcc \
  --output text
```

**Output:**

```

-----BEGIN CERTIFICATE-----
MIIEDzCCAvEgAwIBAgIRAJuJ8f6ZVYL7gG/rS3qvzrZMwDQYJKoZIhvcNAQELBQAw
cTElMAkGA1UEBhMCVVMxEzARBgNVBAGMC1dhc2hpbmd0b24xEDA0BgNVBACMB1Nl
...certificate body truncated for brevity...
tKCSglgZZrd4FdLw1EkGm+UVXnodwMtJEQyy3oTfZjURPIyyaqskTu/KSS7YDjk0
KQNY73D6Ltmd0EbAyq10XiDxqY41lvKHJ1eZrPaBmYNABxU=
-----END CERTIFICATE----- -----BEGIN CERTIFICATE-----
MIIDrzCCApegAwIBAgIRA0skdzLvcj1eShkoyEE693AwDQYJKoZIhvcNAQELBQAw
cTElMAkGA1UEBhMCVVMxEzARBgNVBAGMC1dhc2hpbmd0b24xEDA0BgNVBACMB1Nl
...certificate body truncated for brevity...
kdRGB6P2hpxstDOUIwAoCbhoaWwFA4ybJzfn+j0QhAziNlRdKQRR8nODWpKt7H9w
dJ5nxsTk/fniJz86Ddtp6n8s82wYdkN3cVffeK72A9aTCOU=
-----END CERTIFICATE-----

```

Bagian pertama dari output adalah sertifikat itu sendiri. Bagian kedua adalah rantai sertifikat yang berantai ke sertifikat CA root. Perhatikan bahwa ketika Anda menggunakan `--output text` opsi, TAB karakter disisipkan di antara dua bagian sertifikat (yang merupakan penyebab teks indentasi). Jika Anda bermaksud untuk mengambil output ini dan mengurai sertifikat dengan alat lain, Anda mungkin perlu menghapus TAB karakter sehingga diproses dengan benar.

- Untuk detail API, lihat [GetCertificatedi Referensi AWS CLI Perintah](#).

**import-certificate-authority-certificate**

Contoh kode berikut menunjukkan cara menggunakan `import-certificate-authority-certificate`.

**AWS CLI**

Untuk mengimpor sertifikat otoritas sertifikat Anda ke ACM PCA

`import-certificate-authority-certificate` Perintah berikut mengimpor sertifikat CA pribadi yang ditandatangani untuk CA yang ditentukan oleh ARN ke ACM PCA.

```

aws acm-pca import-certificate-authority-certificate --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --certificate file://C:\ca_cert.pem
--certificate-chain file://C:\ca_cert_chain.pem

```

- Untuk detail API, lihat [ImportCertificateAuthorityCertificatedi Referensi AWS CLI Perintah](#).

## issue-certificate

Contoh kode berikut menunjukkan cara menggunakan `issue-certificate`.

### AWS CLI

Untuk mengeluarkan sertifikat pribadi

`issue-certificate` Perintah berikut menggunakan CA pribadi yang ditentukan oleh ARN untuk mengeluarkan sertifikat pribadi.

```
aws acm-pca issue-certificate --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --csr file://C:\cert_1.csr --signing-algorithm "SHA256WITHRSA" --validity Value=365,Type="DAYS" --idempotency-token 1234
```

- Untuk detail API, lihat [IssueCertificate](#) di Referensi AWS CLI Perintah.

## list-certificate-authorities

Contoh kode berikut menunjukkan cara menggunakan `list-certificate-authorities`.

### AWS CLI

Untuk mendaftarkan otoritas sertifikat pribadi Anda

`list-certificate-authorities` Perintah berikut mencantumkan informasi tentang semua CA pribadi di akun Anda.

```
aws acm-pca list-certificate-authorities --max-results 10
```

- Untuk detail API, lihat [ListCertificateAuthorities](#) di Referensi AWS CLI Perintah.

## list-tags

Contoh kode berikut menunjukkan cara menggunakan `list-tags`.

### AWS CLI

Untuk membuat daftar tag untuk otoritas sertifikat Anda

`list-tags` Perintah berikut mencantumkan tag yang terkait dengan CA pribadi yang ditentukan oleh ARN.

```
aws acm-pca list-tags --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/123455678-1234-1234-1234-123456789012 --max-results 10
```

- Untuk detail API, lihat [ListTags](#) di Referensi AWS CLI Perintah.

## **revoke-certificate**

Contoh kode berikut menunjukkan cara menggunakan `revoke-certificate`.

### AWS CLI

Untuk mencabut sertifikat pribadi

`revoke-certificate` Perintah berikut mencabut sertifikat pribadi dari CA yang diidentifikasi oleh ARN.

```
aws acm-pca revoke-certificate --certificate-authority-arn arn:aws:acm-pca:us-west-2:1234567890:certificate-authority/12345678-1234-1234-1234-123456789012 --certificate-serial 67:07:44:76:83:a9:b7:f4:05:56:27:ff:d5:5c:eb:cc --revocation-reason "KEY_COMPROMISE"
```

- Untuk detail API, lihat [RevokeCertificate](#) di Referensi AWS CLI Perintah.

## **tag-certificate-authority**

Contoh kode berikut menunjukkan cara menggunakan `tag-certificate-authority`.

### AWS CLI

Untuk melampirkan tag ke otoritas sertifikat pribadi

`tag-certificate-authority` Perintah berikut melampirkan satu atau beberapa tag ke CA pribadi Anda.

```
aws acm-pca tag-certificate-authority --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --tags Key=Admin,Value=Alice
```

- Untuk detail API, lihat [TagCertificateAuthority](#) di Referensi AWS CLI Perintah.

## untag-certificate-authority

Contoh kode berikut menunjukkan cara menggunakan `untag-certificate-authority`.

### AWS CLI

Untuk menghapus satu atau beberapa tag dari otoritas sertifikat pribadi Anda

`untag-certificate-authority` Perintah berikut menghapus tag dari CA pribadi yang diidentifikasi oleh ARN.

```
aws acm-pca untag-certificate-authority --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --tags Key=Purpose,Value=Website
```

- Untuk detail API, lihat [UntagCertificateAuthority](#) di Referensi AWS CLI Perintah.

## update-certificate-authority

Contoh kode berikut menunjukkan cara menggunakan `update-certificate-authority`.

### AWS CLI

Untuk memperbarui konfigurasi otoritas sertifikat pribadi Anda

`update-certificate-authority` Perintah berikut memperbarui status dan konfigurasi CA pribadi yang diidentifikasi oleh ARN.

```
aws acm-pca update-certificate-authority --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-1232456789012 --revocation-configuration file://C:\revoke_config.txt --status "DISABLED"
```

- Untuk detail API, lihat [UpdateCertificateAuthority](#) di Referensi AWS CLI Perintah.

## AWS Proton contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Proton.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **cancel-service-instance-deployment**

Contoh kode berikut menunjukkan cara menggunakan `cancel-service-instance-deployment`.

AWS CLI

Untuk membatalkan penerapan instance layanan

`cancel-service-instance-deployment` Contoh berikut membatalkan penerapan instance layanan.

```
aws proton cancel-service-instance-deployment \  
  --service-instance-name "instance-one" \  
  --service-name "simple-svc"
```

Output:

```
{  
  "serviceInstance": {  
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-  
instance/instance-one",  
    "createdAt": "2021-04-02T21:29:59.962000+00:00",  
    "deploymentStatus": "CANCELLING",  
    "environmentName": "simple-env",  
    "lastDeploymentAttemptedAt": "2021-04-02T21:45:15.406000+00:00",
```

```

    "lastDeploymentSucceededAt": "2021-04-02T21:38:00.823000+00:00",
    "name": "instance-one",
    "serviceName": "simple-svc",
    "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_optional_input: abc\n my_sample_pipeline_required_input:
'123'\ninstances:\n- name: my-instance\n environment: MySimpleEnv
\n spec:\n  my_sample_service_instance_optional_input: def\n
my_sample_service_instance_required_input: '456'\n- name: my-other-instance\n
environment: MySimpleEnv\n spec:\n  my_sample_service_instance_required_input:
'789'\n",
    "templateMajorVersion": "1",
    "templateMinorVersion": "1",
    "templateName": "svc-simple"
  }
}

```

Untuk informasi selengkapnya, lihat [Memperbarui instance layanan](#) di Panduan Administrator AWS Proton atau [Memperbarui instance layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [CancelServiceInstanceDeployment](#) di Referensi AWS CLI Perintah.

## cancel-service-pipeline-deployment

Contoh kode berikut menunjukkan cara menggunakan `cancel-service-pipeline-deployment`.

### AWS CLI

Untuk membatalkan penerapan saluran pipa layanan

`cancel-service-pipeline-deployment` Contoh berikut membatalkan penerapan pipeline layanan.

```
aws proton cancel-service-pipeline-deployment \
  --service-name "simple-svc"
```

Output:

```
{
  "pipeline": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/pipeline",
    "createdAt": "2021-04-02T21:29:59.962000+00:00",
    "deploymentStatus": "CANCELLING",
  }
}
```



```

    "lastDeploymentAttemptedAt": "2021-04-02T22:02:45.095000+00:00",
    "lastDeploymentSucceededAt": "2021-04-02T21:39:28.991000+00:00",
    "templateMajorVersion": "1",
    "templateMinorVersion": "1",
    "templateName": "svc-simple"
  }
}

```

Untuk informasi selengkapnya, lihat [Memperbarui pipeline layanan](#) di Panduan Administrator AWS Proton atau [Perbarui saluran layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [CancelServicePipelineDeployment](#) di Referensi AWS CLI Perintah.

## create-service

Contoh kode berikut menunjukkan cara menggunakan `create-service`.

### AWS CLI

Untuk membuat layanan

`create-service` Contoh berikut membuat layanan dengan pipa layanan.

```

aws proton create-service \
  --name "MySimpleService" \
  --template-name "fargate-service" \
  --template-major-version "1" \
  --branch-name "mainline" \
  --repository-connection-arn "arn:aws:codestar-connections:region-id:account-id:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
  --repository-id "myorg/myapp" \
  --spec file://spec.yaml

```

Isi dari `spec.yaml`:

```

proton: ServiceSpec

pipeline:
  my_sample_pipeline_required_input: "hello"
  my_sample_pipeline_optional_input: "bye"

instances:

```

```
- name: "acme-network-dev"
  environment: "ENV_NAME"
  spec:
    my_sample_service_instance_required_input: "hi"
    my_sample_service_instance_optional_input: "ho"
```

**Output:**

```
{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/MySimpleService",
    "createdAt": "2020-11-18T19:50:27.460000+00:00",
    "lastModifiedAt": "2020-11-18T19:50:27.460000+00:00",
    "name": "MySimpleService",
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryId": "myorg/myapp",
    "status": "CREATE_IN_PROGRESS",
    "templateName": "fargate-service"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat layanan](#) di Panduan Administrator AWS Proton dan [Membuat layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [CreateService](#) di Referensi AWS CLI Perintah.

**delete-service**

Contoh kode berikut menunjukkan cara menggunakan `delete-service`.

**AWS CLI**

Untuk menghapus layanan

`delete-service` Contoh berikut menghapus layanan.

```
aws proton delete-service \
  --name "simple-svc"
```

**Output:**

```
{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc",
    "branchName": "mainline",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",
    "description": "Edit by updating description",
    "lastModifiedAt": "2020-11-29T00:30:39.248000+00:00",
    "name": "simple-svc",
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryId": "myorg/myapp",
    "status": "DELETE_IN_PROGRESS",
    "templateName": "fargate-service"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus layanan](#) di Panduan Administrator AWS Proton.

- Untuk detail API, lihat [DeleteService](#) di Referensi AWS CLI Perintah.

## get-service-instance

Contoh kode berikut menunjukkan cara menggunakan `get-service-instance`.

### AWS CLI

Untuk mendapatkan detail instance layanan

`get-service-instance` Contoh berikut mendapatkan data detail untuk instance layanan.

```
aws proton get-service-instance \
  --name "instance-one" \
  --service-name "simple-svc"
```

Output:

```
{
  "serviceInstance": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",
```

```

    "deploymentStatus": "SUCCEEDED",
    "environmentName": "simple-env",
    "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
    "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
    "name": "instance-one",
    "serviceName": "simple-svc",
    "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_optional_input: hello world\n
my_sample_pipeline_required_input: pipeline up\ninstances:\n- name: instance-one\n
environment: my-simple-env\n spec:\n   my_sample_service_instance_optional_input:
Ola\n   my_sample_service_instance_required_input: Ciao\n",
    "templateMajorVersion": "1",
    "templateMinorVersion": "0",
    "templateName": "svc-simple"
  }
}

```

Untuk informasi selengkapnya, lihat [Melihat data layanan](#) di Panduan Administrator AWS Proton atau [Melihat data layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [GetServiceInstance](#) di Referensi AWS CLI Perintah.

## get-service

Contoh kode berikut menunjukkan cara menggunakan `get-service`.

### AWS CLI

Untuk mendapatkan detail layanan

`get-service` Contoh berikut mendapatkan data detail untuk suatu layanan.

```
aws proton get-service \
  --name "simple-svc"
```

Output:

```
{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc",
    "branchName": "mainline",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",

```

```

    "lastModifiedAt": "2020-11-28T22:44:51.207000+00:00",
    "name": "simple-svc",
    "pipeline": {
      "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/
pipeline/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "createdAt": "2020-11-28T22:40:50.512000+00:00",
      "deploymentStatus": "SUCCEEDED",
      "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
      "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
      "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_required_input: hello\n my_sample_pipeline_optional_input:
bye\ninstances:\n- name: instance-svc-simple\n environment: my-simple-
env\n spec:\n  my_sample_service_instance_required_input: hi\n
my_sample_service_instance_optional_input: ho\n",
      "templateMajorVersion": "1",
      "templateMinorVersion": "1",
      "templateName": "svc-simple"
    },
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "repositoryId": "myorg/myapp",
    "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_required_input: hello\n my_sample_pipeline_optional_input:
bye\ninstances:\n- name: instance-svc-simple\n environment: my-simple-
env\n spec:\n  my_sample_service_instance_required_input: hi\n
my_sample_service_instance_optional_input: ho\n",
    "status": "ACTIVE",
    "templateName": "svc-simple"
  }
}

```

Untuk informasi selengkapnya, lihat [Melihat data layanan](#) di Panduan Administrator AWS Proton atau [Melihat data layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [GetService](#) di Referensi AWS CLI Perintah.

## list-service-instances

Contoh kode berikut menunjukkan cara menggunakan `list-service-instances`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua instance layanan

`list-service-instances` Contoh berikut mencantumkan instance layanan.

```
aws proton list-service-instances
```

Output:

```
{
  "serviceInstances": [
    {
      "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/
service-instance/instance-one",
      "createdAt": "2020-11-28T22:40:50.512000+00:00",
      "deploymentStatus": "SUCCEEDED",
      "environmentArn": "arn:aws:proton:region-id:123456789012:environment/
simple-env",
      "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
      "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
      "name": "instance-one",
      "serviceName": "simple-svc",
      "templateMajorVersion": "1",
      "templateMinorVersion": "0",
      "templateName": "fargate-service"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat data instance layanan](#) di Panduan Administrator AWS Proton atau [Melihat data instance layanan](#) di Panduan Pengguna AWS Proton.

Contoh 2: Untuk daftar contoh layanan yang ditentukan

`get-service-instance` Contoh berikut mendapat instance layanan.

```
aws proton get-service-instance \
  --name "instance-one" \
  --service-name "simple-svc"
```

Output:

```
{
  "serviceInstance": {
```

```

    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",
    "deploymentStatus": "SUCCEEDED",
    "environmentName": "simple-env",
    "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
    "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
    "name": "instance-one",
    "serviceName": "simple-svc",
    "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_optional_input: hello world\n
my_sample_pipeline_required_input: pipeline up\ninstances:\n- name: instance-one\n
environment: my-simple-env\n spec:\n   my_sample_service_instance_optional_input:
Ola\n   my_sample_service_instance_required_input: Ciao\n",
    "templateMajorVersion": "1",
    "templateMinorVersion": "0",
    "templateName": "svc-simple"
  }
}

```

Untuk informasi selengkapnya, lihat [Melihat data instance layanan](#) di Panduan Administrator AWS Proton atau [Melihat data instance layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [ListServiceInstances](#) di Referensi AWS CLI Perintah.

## update-service-instance

Contoh kode berikut menunjukkan cara menggunakan `update-service-instance`.

### AWS CLI

Untuk memperbarui instance layanan ke versi minor baru

`update-service-instance` Contoh berikut memperbarui instance layanan ke versi minor baru dari template layanannya yang menambahkan instance baru bernama "my-other-instance" dengan input baru yang diperlukan.

```

aws proton update-service-instance \
  --service-name "simple-svc" \
  --spec "file://service-spec.yaml" \
  --template-major-version "1" \
  --template-minor-version "1" \

```

```
--deployment-type "MINOR_VERSION" \  
--name "instance-one"
```

Isi dari `service-spec.yaml`:

```
proton: ServiceSpec  
pipeline:  
  my_sample_pipeline_optional_input: "abc"  
  my_sample_pipeline_required_input: "123"  
instances:  
  - name: "instance-one"  
    environment: "simple-env"  
    spec:  
      my_sample_service_instance_optional_input: "def"  
      my_sample_service_instance_required_input: "456"  
  - name: "my-other-instance"  
    environment: "simple-env"  
    spec:  
      my_sample_service_instance_required_input: "789"
```

Output:

```
{  
  "serviceInstance": {  
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-  
instance/instance-one",  
    "createdAt": "2021-04-02T21:29:59.962000+00:00",  
    "deploymentStatus": "IN_PROGRESS",  
    "environmentName": "arn:aws:proton:region-id:123456789012:environment/  
simple-env",  
    "lastDeploymentAttemptedAt": "2021-04-02T21:38:00.823000+00:00",  
    "lastDeploymentSucceededAt": "2021-04-02T21:29:59.962000+00:00",  
    "name": "instance-one",  
    "serviceName": "simple-svc",  
    "templateMajorVersion": "1",  
    "templateMinorVersion": "0",  
    "templateName": "svc-simple"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui instance layanan](#) di Panduan Administrator AWS Proton atau [Memperbarui instance layanan](#) di Panduan Pengguna AWS Proton.



- Untuk detail API, lihat [UpdateServiceInstance](#) di Referensi AWS CLI Perintah.

## update-service-pipeline

Contoh kode berikut menunjukkan cara menggunakan `update-service-pipeline`.

### AWS CLI

Untuk memperbarui pipeline layanan

`update-service-pipeline` Contoh berikut memperbarui pipeline layanan ke versi minor baru dari template layanannya.

```
aws proton update-service-pipeline \
  --service-name "simple-svc" \
  --spec "file://service-spec.yaml" \
  --template-major-version "1" \
  --template-minor-version "1" \
  --deployment-type "MINOR_VERSION"
```

Output:

```
{
  "pipeline": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/pipeline/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "createdAt": "2021-04-02T21:29:59.962000+00:00",
    "deploymentStatus": "IN_PROGRESS",
    "lastDeploymentAttemptedAt": "2021-04-02T21:39:28.991000+00:00",
    "lastDeploymentSucceededAt": "2021-04-02T21:29:59.962000+00:00",
    "spec": "proton: ServiceSpec\n\npipeline:\n
my_sample_pipeline_optional_input: \"abc\"\n
my_sample_pipeline_required_input:
\"123\"\n\ninstances:\n
- name: \"my-instance\"\n
  environment: \"MySimpleEnv\"\n\n
spec:\n
  my_sample_service_instance_optional_input: \"def\"\n\n
  my_sample_service_instance_required_input: \"456\"\n
  - name: \"my-other-instance\"\n
  environment: \"MySimpleEnv\"\n\n
spec:\n
my_sample_service_instance_required_input: \"789\"\n\n",
    "templateMajorVersion": "1",
    "templateMinorVersion": "0",
    "templateName": "svc-simple"
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui pipeline layanan](#) di Panduan Administrator AWS Proton atau [Perbarui saluran layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [UpdateServicePipeline](#) di Referensi AWS CLI Perintah.

## update-service

Contoh kode berikut menunjukkan cara menggunakan `update-service`.

### AWS CLI

Untuk memperbarui layanan

`update-service` Contoh berikut mengedit deskripsi layanan.

```
aws proton update-service \  
  --name "MySimpleService" \  
  --description "Edit by updating description"
```

Output:

```
{  
  "service": {  
    "arn": "arn:aws:proton:region-id:123456789012:service/MySimpleService",  
    "branchName": "mainline",  
    "createdAt": "2021-03-12T22:39:42.318000+00:00",  
    "description": "Edit by updating description",  
    "lastModifiedAt": "2021-03-12T22:44:21.975000+00:00",  
    "name": "MySimpleService",  
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-  
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "repositoryId": "myorg/myapp",  
    "status": "ACTIVE",  
    "templateName": "fargate-service"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengedit layanan](#) di Panduan Administrator AWS Proton atau [Mengedit layanan](#) di Panduan Pengguna AWS Proton.

- Untuk detail API, lihat [UpdateService](#) di Referensi AWS CLI Perintah.

## Contoh QLDB menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan QLDB AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **cancel-journal-kinesis-stream**

Contoh kode berikut menunjukkan cara menggunakan `cancel-journal-kinesis-stream`.

AWS CLI

Untuk membatalkan aliran jurnal

`cancel-journal-kinesis-stream` Contoh berikut membatalkan aliran jurnal yang ditentukan dari buku besar.

```
aws qldb cancel-journal-kinesis-stream \  
  --ledger-name myExampleLedger \  
  --stream-id 7ISckqwe4y25YyHLzYUFAf
```

Output:

```
{  
  "StreamId": "7ISckqwe4y25YyHLzYUFAf"
```

```
}
```

Untuk informasi selengkapnya, lihat [Streaming data jurnal dari Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [CancelJournalKinesisStream](#) di Referensi AWS CLI Perintah.

## create-ledger

Contoh kode berikut menunjukkan cara menggunakan `create-ledger`.

### AWS CLI

Contoh 1: Untuk membuat buku besar dengan properti default

`create-ledger` Contoh berikut membuat buku besar dengan nama `myExampleLedger` dan modus izin. `STANDARD` Parameter opsional untuk perlindungan penghapusan dan kunci AWS KMS tidak ditentukan, sehingga mereka default ke `true` dan kunci KMS yang AWS dimiliki masing-masing.

```
aws qlldb create-ledger \  
  --name myExampleLedger \  
  --permissions-mode STANDARD
```

Output:

```
{  
  "State": "CREATING",  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "DeletionProtection": true,  
  "CreationDateTime": 1568839243.951,  
  "Name": "myExampleLedger",  
  "PermissionsMode": "STANDARD"  
}
```

Contoh 2: Untuk membuat buku besar dengan perlindungan penghapusan dinonaktifkan, kunci KMS yang dikelola pelanggan, dan tag yang ditentukan

`create-ledger` Contoh berikut membuat buku besar dengan nama `myExampleLedger2` dan modus izin. `STANDARD` Fitur perlindungan penghapusan dinonaktifkan, kunci KMS yang dikelola

pelanggan yang ditentukan digunakan untuk enkripsi saat istirahat, dan tag yang ditentukan dilampirkan ke sumber daya.

```
aws qlldb create-ledger \  
  --name myExampleLedger2 \  
  --permissions-mode STANDARD \  
  --no-deletion-protection \  
  --kms-key arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111 \  
  --tags IsTest=true,Domain=Test
```

Output:

```
{  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger2",  
  "DeletionProtection": false,  
  "CreationDateTime": 1568839543.557,  
  "State": "CREATING",  
  "Name": "myExampleLedger2",  
  "PermissionsMode": "STANDARD",  
  "KmsKeyArn": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111"  
}
```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Buku Besar QLDB Amazon di Panduan Pengembang QLDB Amazon](#).

- Untuk detail API, lihat [CreateLedger](#) di Referensi AWS CLI Perintah.

## delete-ledger

Contoh kode berikut menunjukkan cara menggunakan `delete-ledger`.

### AWS CLI

Untuk menghapus buku besar

`delete-ledger` Contoh berikut menghapus buku besar yang ditentukan.

```
aws qlldb delete-ledger \  
  --name myExampleLedger
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Buku Besar QLDB Amazon di Panduan Pengembang QLDB Amazon](#).

- Untuk detail API, lihat [DeleteLedger](#) di Referensi AWS CLI Perintah.

## describe-journal-kinesis-stream

Contoh kode berikut menunjukkan cara menggunakan `describe-journal-kinesis-stream`.

### AWS CLI

Untuk menggambarkan aliran jurnal

`describe-journal-kinesis-stream` Contoh berikut menampilkan rincian untuk aliran jurnal tertentu dari buku besar.

```
aws qlldb describe-journal-kinesis-stream \  
  --ledger-name myExampleLedger \  
  --stream-id 7ISCKqwe4y25YyHLzYUFAf
```

### Output:

```
{  
  "Stream": {  
    "LedgerName": "myExampleLedger",  
    "CreationTime": 1591221984.677,  
    "InclusiveStartTime": 1590710400.0,  
    "ExclusiveEndTime": 1590796799.0,  
    "RoleArn": "arn:aws:iam::123456789012:role/my-kinesis-stream-role",  
    "StreamId": "7ISCKqwe4y25YyHLzYUFAf",  
    "Arn": "arn:aws:qlldb:us-east-1:123456789012:stream/  
myExampleLedger/7ISCKqwe4y25YyHLzYUFAf",  
    "Status": "ACTIVE",  
    "KinesisConfiguration": {  
      "StreamArn": "arn:aws:kinesis:us-east-1:123456789012:stream/stream-for-  
qlldb",  
      "AggregationEnabled": true  
    },  
    "StreamName": "myExampleLedger-stream"  
  }  
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Streaming data jurnal dari Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [DescribeJournalKinesisStream](#) di Referensi AWS CLI Perintah.

## describe-journal-s3-export

Contoh kode berikut menunjukkan cara menggunakan `describe-journal-s3-export`.

### AWS CLI

Untuk menggambarkan pekerjaan ekspor jurnal

`describe-journal-s3-export` Contoh berikut menampilkan rincian untuk pekerjaan ekspor tertentu dari buku besar.

```
aws qlldb describe-journal-s3-export \  
  --name myExampleLedger \  
  --export-id ADR20NPKN5LINYGb4dp7yZ
```

### Output:

```
{  
  "ExportDescription": {  
    "S3ExportConfiguration": {  
      "Bucket": "awsExampleBucket",  
      "Prefix": "ledgerexport1/",  
      "EncryptionConfiguration": {  
        "ObjectEncryptionType": "SSE_S3"  
      }  
    },  
    "RoleArn": "arn:aws:iam::123456789012:role/my-s3-export-role",  
    "Status": "COMPLETED",  
    "ExportCreationTime": 1568847801.418,  
    "InclusiveStartTime": 1568764800.0,  
    "ExclusiveEndTime": 1568847599.0,  
    "LedgerName": "myExampleLedger",  
    "ExportId": "ADR20NPKN5LINYGb4dp7yZ"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengeksport Jurnal Anda di Amazon QLDB](#) di Panduan Pengembang QLDB Amazon.

- Untuk detail API, lihat [DescribeJournalS3Export](#) di Referensi AWS CLI Perintah.

## describe-ledger

Contoh kode berikut menunjukkan cara menggunakan `describe-ledger`.

### AWS CLI

Untuk menggambarkan buku besar

`describe-ledger` Contoh berikut menampilkan rincian untuk buku besar yang ditentukan.

```
aws qldb describe-ledger \  
  --name myExampleLedger
```

Output:

```
{  
  "CreationDateTime": 1568839243.951,  
  "Arn": "arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "State": "ACTIVE",  
  "Name": "myExampleLedger",  
  "DeletionProtection": true,  
  "PermissionsMode": "STANDARD",  
  "EncryptionDescription": {  
    "KmsKeyArn": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-  
cdef-EXAMPLE11111",  
    "EncryptionStatus": "ENABLED"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Buku Besar QLDB Amazon di Panduan Pengembang QLDB](#) Amazon.

- Untuk detail API, lihat [DescribeLedger](#) di Referensi AWS CLI Perintah.

## export-journal-to-s3

Contoh kode berikut menunjukkan cara menggunakan `export-journal-to-s3`.



## AWS CLI

Untuk mengekspor blok jurnal ke S3

`export-journal-to-s3` Contoh berikut membuat pekerjaan ekspor untuk blok jurnal dalam rentang tanggal dan waktu tertentu dari buku besar dengan nama `myExampleLedger`. Pekerjaan ekspor menulis blok ke dalam bucket Amazon S3 tertentu.

```
aws qlldb export-journal-to-s3 \  
  --name myExampleLedger \  
  --inclusive-start-time 2019-09-18T00:00:00Z \  
  --exclusive-end-time 2019-09-18T22:59:59Z \  
  --role-arn arn:aws:iam::123456789012:role/my-s3-export-role \  
  --s3-export-configuration file://my-s3-export-config.json
```

Isi dari `my-s3-export-config.json`:

```
{  
  "Bucket": "awsExampleBucket",  
  "Prefix": "ledgerexport1/",  
  "EncryptionConfiguration": {  
    "ObjectEncryptionType": "SSE_S3"  
  }  
}
```

Output:

```
{  
  "ExportId": "ADR20NPKN5LINYGb4dp7yZ"  
}
```

Untuk informasi selengkapnya, lihat [Mengekspor Jurnal Anda di Amazon QLDB](#) di Panduan Pengembang QLDB Amazon.

- Untuk detail API, lihat [ExportJournalToS3](#) di Referensi AWS CLI Perintah.

## get-block

Contoh kode berikut menunjukkan cara menggunakan `get-block`.

## AWS CLI

Contoh 1: Untuk mendapatkan blok jurnal dan bukti verifikasi menggunakan file input

get-block Contoh berikut meminta objek data blok dan bukti dari buku besar yang ditentukan. Permintaan adalah untuk alamat tip intisari tertentu dan alamat blok.

```
aws qldb get-block \
  --name vehicle-registration \
  --block-address file://myblockaddress.json \
  --digest-tip-address file://mydigesttipaddress.json
```

Isi dari myblockaddress.json:

```
{
  "IonText": "{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\",sequenceNo:100}"
}
```

Isi dari mydigesttipaddress.json:

```
{
  "IonText": "{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\",sequenceNo:123}"
}
```

Output:

```
{
  "Block": {
    "IonText": "{blockAddress:{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1
  \",sequenceNo:100},transactionId:\"FnQeJBAicTX0Ah32ZnVtSX
  \",blockTimestamp:2019-09-16T19:37:05.360Z,blockHash:
  {{NoChM92yKRuJAb/jeLd1VnYn4DHiWI071ACfic9uHc=}},entriesHash:
  {{105L0siKV14SDbuaYnH7uwXzUvqzIwUiRLXGbTyj/nY=}},previousBlockHash:
  {{7kewBXhpdBc1cZKxhVmpoMhpUGOJtWQD0iY2LPfZkYA=}},entriesHashList:
  [{{eRSwnmAM7WWANWd5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{mHVex/
  yjHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},
  {{y5cCB7r7p0AIUfsVQ1j0TqtE97b4b4oo1R0vnYyE5wWM=}},{{TvTXygML1bMe6NvEZtGkX
  +KR+W/EJl4qD1mmV77KZQg=}}],transactionInfo:{statements:[{statement:
  \"FROM VehicleRegistration AS r \\nWHERE r.VIN = '1N4AL11D75C109151'\\
  \\nINSERT INTO r.Owners.SecondaryOwners\\n  VALUE { 'PersonId' :
  'CMvdR77XP8zAglmmFDGTvt' } }],startTime:2019-09-16T19:37:05.302Z,statementDigest:
```

```
{jcgPX2vs0J0waum4qmDYtn1pCAT9xKNIzA+2k4R+mxA=}}}], documents:
{JUJgkIcNbhS2goq8RqLuZ4:{tableName:\"VehicleRegistration\", tableId:
\"BFJKdXgz9oF4wjMbuXy4G\", statements:[0]}}, revisions:[{blockAddress:
{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\", sequenceNo:100}, hash:
{mHVex/yjHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}}, data:{VIN:
\"1N4AL11D75C109151\", LicensePlateNumber:\"LEWISR261LL\", State:\"WA
\", PendingPenaltyTicketAmount:90.25, ValidFromDate:2017-08-21, ValidToDate:2020-05-11, Owners:
{PrimaryOwner:{PersonId:\"BFJKdXhnLRT27sXBnojNGW\"}, SecondaryOwners:
[{PersonId:\"CMVdR77XP8zAg1mmFDGTvt\"}]}, City:\"Everett\"}, metadata:{id:
\"JUJgkIcNbhS2goq8RqLuZ4\", version:3, txTime:2019-09-16T19:37:05.344Z, txId:
\"FnQeJBAicTX0Ah32ZnVtSX\"}}}]"}
},
"Proof": {
  "IonText": "[{{l3+EXs69K1+rehlqyWLkt+oHDlw4Zi9pCLW/t/mgTPM=}},
{{48CXG3ehPqsxCYd34EEa8Fso00RpWwA08010RJKf3Do=}}, {{9UnwnKSQT0i3ge1JMva
+tMIqCEDa0PTkwxmyHSn8UPQ=}}, {{3nW6Vryghk+7pd6wFctLufgPM6qXHyTNeCb1sCwcDaI=}},
{{Irb5fNhbRiNEQ1VPhzlnGT/ZQPadSmgfdtMYcwkN0xoI=}}, {{+3CwpYG/ytf/
vq9GidpzSx6JJiLXt1hMQWnNq0y3jfY=}}, {{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT
+qE=}}]"
}
}
```

Untuk informasi selengkapnya, lihat [Verifikasi Data di Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

Contoh 2: Untuk mendapatkan blok jurnal dan bukti verifikasi menggunakan sintaks singkatan

`get-block` Contoh berikut meminta objek data blok dan bukti dari buku besar yang ditentukan menggunakan sintaks singkatan. Permintaan adalah untuk alamat tip intisari tertentu dan alamat blok.

```
aws qldb get-block \
  --name vehicle-registration \
  --block-address 'IonText="{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\", sequenceNo:100}"' \
  --digest-tip-address 'IonText="{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1
\", sequenceNo:123}"'
```

Output:

```
{
  "Block": {
```

```

    "IonText": "{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1
\\",sequenceNo:100},transactionId:\\"FnQeJBAicTX0Ah32ZnVtSX
\\",blockTimestamp:2019-09-16T19:37:05.360Z,blockHash:
{{NoChM92yKRuJAb/jeLd1VnYn4DHiWiF071ACfic9uHc=}},entriesHash:
{{105L0siKV14SDbuaYnH7uwXzUvqzIwUiRLXGbtYj/nY=}},previousBlockHash:
{{7kewBXhpdBcLcZKxhVmpoMhPUGOJtWQD0iY2LPfZkYA=}},entriesHashList:
[{{eRSwnmAM7WWANWd5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{mHVex/
yJHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},
{{y5cCB7p0AIUfsVQ1j0TqtE97b4b4oo1R0vnYyE5wWM=}},{{TvTXygML1bMe6NvEZtGkX
+KR+W/EJl4qD1mmV77KZQg=}}],transactionInfo:{statements:[{statement:
\\\"FROM VehicleRegistration AS r \\nWHERE r.VIN = '1N4AL11D75C109151'\\
\\nINSERT INTO r.Owners.SecondaryOwners\\n    VALUE { 'PersonId' :
'CMVdR77XP8zAg1mmFDGTvt' }\\\",startTime:2019-09-16T19:37:05.302Z,statementDigest:
{{jcgPX2vs0J0waum4qmDYtn1pCAT9xKNIZa+2k4R+mxA=}}}],documents:
{{JUJgkIcNbhS2goq8RqLuZ4:{tableName:\\"VehicleRegistration\\\",tableId:
\\\"BFJKdXgzt9oF4wjMbuXy4G\\\",statements:[0]}}}],revisions:[{blockAddress:
{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\\",sequenceNo:100},hash:
{{mHVex/yJHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},data:{VIN:
\\\"1N4AL11D75C109151\\\",LicensePlateNumber:\\"LEWISR261LL\\\",State:\\"WA
\\\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:
{{PrimaryOwner:{PersonId:\\"BFJKdXhnLRT27sXBnojNGW\\\"},SecondaryOwners:
[{{PersonId:\\"CMVdR77XP8zAg1mmFDGTvt\\\"}]}],City:\\"Everett\\\"},metadata:{id:
\\\"JUJgkIcNbhS2goq8RqLuZ4\\\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:
\\\"FnQeJBAicTX0Ah32ZnVtSX\\\"}}]}}
  },
  "Proof": {
    "IonText": "[{{13+EXs69K1+rehlqyWlkt+oHD1w4Zi9pCLW/t/mgTPM=}},
{{48CXG3ehPqsxCYd34EEa8Fso00RpWwA08010RJKf3Do=}},{{9UnwnKSQT0i3ge1JMva
+tMIqCEDaOPTkwxmyHSn8UPQ=}},{{3nW6Vryghk+7pd6wFCtLufgPM6qXHyTNECb1sCwcDaI=}},
{{Irb5fNhBrNEQ1VPhzlnGT/ZQPadSmgfdtMyCwkN0xoI=}},{{+3CwPYG/ytf/
vq9GidpzSx6JJiLXt1hMQWnNq0y3jfy=}},{{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT
+qE=}}]"
  }
}

```

Untuk informasi selengkapnya, lihat [Verifikasi Data di Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [GetBlock](#) di Referensi AWS CLI Perintah.

## get-digest

Contoh kode berikut menunjukkan cara menggunakan `get-digest`.

## AWS CLI

Untuk mendapatkan intisari untuk buku besar

`get-digest` Contoh berikut meminta intisari dari buku besar yang ditentukan di blok komit terbaru dalam jurnal.

```
aws qlldb get-digest \  
  --name vehicle-registration
```

Output:

```
{  
  "Digest": "6m6BMXobbJKpMhahwVthAEsN6awgnHK62Qq5McGP1Gk=",  
  "DigestTipAddress": {  
    "IonText": "{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:123}"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Verifikasi Data di Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [GetDigest](#) di Referensi AWS CLI Perintah.

## `get-revision`

Contoh kode berikut menunjukkan cara menggunakan `get-revision`.

## AWS CLI

Contoh 1: Untuk mendapatkan revisi dokumen dan bukti verifikasi menggunakan file input

`get-revision` Contoh berikut meminta objek data revisi dan bukti dari buku besar yang ditentukan. Permintaan adalah untuk alamat tip intisari tertentu, ID dokumen, dan alamat blok revisi.

```
aws qlldb get-revision \  
  --name vehicle-registration \  
  --block-address file://myblockaddress.json \  
  --document-id JUJgkIcNbhS2goq8RqLuZ4 \  
  --revision-id 1
```

```
--digest-tip-address file://mydigesttipaddress.json
```

Isi dari myblockaddress.json:

```
{
  "IonText": "{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100}"
}
```

Isi dari mydigesttipaddress.json:

```
{
  "IonText": "{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:123}"
}
```

Output:

```
{
  "Revision": {
    "IonText": "{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100},hash:{{mHVex/yjHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},data:
    {VIN:\\"1N4AL11D75C109151\\",LicensePlateNumber:\\"LEWISR261LL\\",State:\\"WA\\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:
    {PrimaryOwner:{PersonId:\\"BFJKdXhnLRT27sXBnojNGW\\"},SecondaryOwners:
    [{PersonId:\\"CMVdR77XP8zAgImmFDGTvt\\"}]},City:\\"Everett\\"},metadata:{id:
    \\"JUJgkIcNbhS2goq8RqLuZ4\\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:
    \\"FnQeJBAicTX0Ah32ZnVtSX\\"}}}"
  },
  "Proof": {
    "IonText": "[{{eRSwnmAM7WWANWDd5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{VV1rdaNuf
    +yJZVGlmsM6gr2T52QvB08Lg+KgpjcnWAU=}},
    {{7kewBXhpdbClcZKxhVmpoMHPUG0JtWQD0iY2LPfZkYA=}},{{13+EXs69K1+rehlqyWLkt
    +oHDlw4Zi9pCLW/t/mgTPM=}},{{48CXG3ehPqsxCYd34EEa8Fso00RpWAA08010RJKf3Do=}},
    {{9UnwnKSQT0i3ge1JMVa+tMIqCEDaOPTkWxmyHSn8UPQ=}},{{3nW6Vryghk
    +7pd6wFCtLufgPM6qXHyTNeCb1sCwcDaI=}},{{Irb5fNhBrNEQ1VPhzlnGT/
    ZQPadSmgfdtMYcwkN0xoI=}},{{+3CWpYG/ytf/vq9GidpzSx6JJiLXt1hMQWNnq0y3jfY=}},
    {{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT+qE=}}]"
  }
}
```

Untuk informasi selengkapnya, lihat [Verifikasi Data di Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

Contoh 2: Untuk mendapatkan revisi dokumen dan bukti verifikasi menggunakan sintaks singkatan

`get-revision` Contoh berikut meminta objek data revisi dan bukti dari buku besar yang ditentukan menggunakan sintaks singkatan. Permintaan adalah untuk alamat tip intisari tertentu, ID dokumen, dan alamat blok revisi.

```
aws qlldb get-revision \
  --name vehicle-registration \
  --block-address 'IonText="{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100}"' \
  --document-id JUJgkIcNbhS2goq8RqLuZ4 \
  --digest-tip-address 'IonText="{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:123}"'
```

Output:

```
{
  "Revision": {
    "IonText": "{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100},hash:{{mHVex/yjHAWjFPpwhBuH2GKXmKJjK2FBa9faqoUVNtg=}},data:{{VIN:\\"1N4AL11D75C109151\\",LicensePlateNumber:\\"LEWISR261LL\\",State:\\"WA\\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:{{PrimaryOwner:{PersonId:\\"BFJKdXhnLRT27sXBnojNGW\\"},SecondaryOwners:{{PersonId:\\"CMVdR77XP8zAg1mmFDGTvt\\"}}}},City:\\"Everett\\"},metadata:{id:\\"JUJgkIcNbhS2goq8RqLuZ4\\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:\\"FnQeJBAicTX0Ah32ZnVtSX\\"}}}"
  },
  "Proof": {
    "IonText": "[{{eRSwnmAM7WWANWd5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{VV1rdaNuf+yJZVG1msM6gr2T52QvB08Lg+KgpjcnWAU=}},{{7kewBXhpdBc1cZKxhVmpoMhpUGOJtWQD0iY2LPfZkYA=}},{{13+EXs69K1+rehlqyWLkt+oHD1w4Zi9pCLW/t/mgTPM=}},{{48CXG3ehPqsxCYd34EEa8Fso00RpWWA08010RJKf3Do=}},{{9UnwnKSQT0i3ge1JMva+tMIqCEDaOPTkwxmyHSn8UPQ=}},{{3nW6Vryghk+7pd6wFcTlufgPM6qXHyTNeCb1sCwcDaI=}},{{Irb5fNhBrNEQ1VPhz1nGT/ZQPadSmgfdtMYcwkN0xoI=}},{{+3CWpYG/ytf/vq9GidpzSx6JJiLXt1hMQWNnq0y3jfY=}},{{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT+qE=}}]"
  }
}
```

Untuk informasi selengkapnya, lihat [Verifikasi Data di Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [GetRevision](#) di Referensi AWS CLI Perintah.

## list-journal-kinesis-streams-for-ledger

Contoh kode berikut menunjukkan cara menggunakan `list-journal-kinesis-streams-for-ledger`.

### AWS CLI

Untuk membuat daftar aliran jurnal untuk buku besar

`list-journal-kinesis-streams-for-ledger` Contoh berikut mencantumkan aliran jurnal untuk buku besar yang ditentukan.

```
aws qlldb list-journal-kinesis-streams-for-ledger \  
  --ledger-name myExampleLedger
```

Output:

```
{  
  "Streams": [  
    {  
      "LedgerName": "myExampleLedger",  
      "CreationTime": 1591221984.677,  
      "InclusiveStartTime": 1590710400.0,  
      "ExclusiveEndTime": 1590796799.0,  
      "RoleArn": "arn:aws:iam::123456789012:role/my-kinesis-stream-role",  
      "StreamId": "7ISCKqwe4y25YyHLzYUFaf",  
      "Arn": "arn:aws:qlldb:us-east-1:123456789012:stream/  
myExampleLedger/7ISCKqwe4y25YyHLzYUFaf",  
      "Status": "ACTIVE",  
      "KinesisConfiguration": {  
        "StreamArn": "arn:aws:kinesis:us-east-1:123456789012:stream/stream-  
for-qlldb",  
        "AggregationEnabled": true  
      },  
      "StreamName": "myExampleLedger-stream"  
    }  
  ]  
}
```



Untuk informasi selengkapnya, lihat [Streaming data jurnal dari Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [ListJournalKinesisStreamsForLedger](#) di Referensi AWS CLI Perintah.

## **list-journal-s3-exports-for-ledger**

Contoh kode berikut menunjukkan cara menggunakan `list-journal-s3-exports-for-ledger`.

### AWS CLI

Untuk membuat daftar pekerjaan ekspor jurnal untuk buku besar

`list-journal-s3-exports-for-ledger` Contoh berikut mencantumkan pekerjaan ekspor jurnal untuk buku besar yang ditentukan.

```
aws qldb list-journal-s3-exports-for-ledger \  
  --name myExampleLedger
```

### Output:

```
{  
  "JournalS3Exports": [  
    {  
      "LedgerName": "myExampleLedger",  
      "ExclusiveEndTime": 1568847599.0,  
      "ExportCreationTime": 1568847801.418,  
      "S3ExportConfiguration": {  
        "Bucket": "awsExampleBucket",  
        "Prefix": "ledgerexport1/",  
        "EncryptionConfiguration": {  
          "ObjectEncryptionType": "SSE_S3"  
        }  
      },  
      "ExportId": "ADR20NPKN5LINYGb4dp7yZ",  
      "RoleArn": "arn:aws:iam::123456789012:role/qldb-s3-export",  
      "InclusiveStartTime": 1568764800.0,  
      "Status": "IN_PROGRESS"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengeksport Jurnal Anda di Amazon QLDB](#) di Panduan Pengembang QLDB Amazon.

- Untuk detail API, lihat [ListJournalS3 ExportsForLedger](#) di Referensi AWS CLI Perintah.

## list-journal-s3-exports

Contoh kode berikut menunjukkan cara menggunakan `list-journal-s3-exports`.

### AWS CLI

Untuk daftar pekerjaan ekspor jurnal

`list-journal-s3-exports` Contoh berikut mencantumkan pekerjaan ekspor jurnal untuk semua buku besar yang terkait dengan AWS akun saat ini dan Wilayah.

```
aws qldb list-journal-s3-exports
```

Output:

```
{
  "JournalS3Exports": [
    {
      "Status": "IN_PROGRESS",
      "LedgerName": "myExampleLedger",
      "S3ExportConfiguration": {
        "EncryptionConfiguration": {
          "ObjectEncryptionType": "SSE_S3"
        },
        "Bucket": "awsExampleBucket",
        "Prefix": "ledgerexport1/"
      },
      "RoleArn": "arn:aws:iam::123456789012:role/my-s3-export-role",
      "ExportCreationTime": 1568847801.418,
      "ExportId": "ADR20NPKN5LINYGb4dp7yZ",
      "InclusiveStartTime": 1568764800.0,
      "ExclusiveEndTime": 1568847599.0
    },
    {
      "Status": "COMPLETED",
      "LedgerName": "myExampleLedger2",
      "S3ExportConfiguration": {
        "EncryptionConfiguration": {
```

```

        "ObjectEncryptionType": "SSE_S3"
      },
      "Bucket": "awsExampleBucket",
      "Prefix": "ledgerexport1/"
    },
    "RoleArn": "arn:aws:iam::123456789012:role/my-s3-export-role",
    "ExportCreationTime": 1568846847.638,
    "ExportId": "2pdvW8UQrjBAiYTMehEJDI",
    "InclusiveStartTime": 1568592000.0,
    "ExclusiveEndTime": 1568764800.0
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mengekspor Jurnal Anda di Amazon QLDB](#) di Panduan Pengembang QLDB Amazon.

- Untuk detail API, lihat [ListJournalS3Exports](#) di AWS CLI Referensi Perintah.

## list-ledgers

Contoh kode berikut menunjukkan cara menggunakan `list-ledgers`.

### AWS CLI

Untuk membuat daftar buku besar yang tersedia

`list-ledgers` Contoh berikut mencantumkan semua buku besar yang terkait dengan AWS akun saat ini dan Wilayah.

```
aws qldb list-ledgers
```

Output:

```

{
  "Ledgers": [
    {
      "State": "ACTIVE",
      "CreationDateTime": 1568839243.951,
      "Name": "myExampleLedger"
    },
    {
      "State": "ACTIVE",

```

```
        "CreationDateTime": 1568839543.557,  
        "Name": "myExampleLedger2"  
    }  
]  
}
```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Buku Besar QLDB Amazon di Panduan Pengembang QLDB Amazon](#).

- Untuk detail API, lihat [ListLedgers](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan pada buku besar

`list-tags-for-resource` Contoh berikut mencantumkan semua tag yang dilampirkan pada buku besar yang ditentukan.

```
aws qldb list-tags-for-resource \  
  --resource-arn arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger
```

Output:

```
{  
  "Tags": {  
    "IsTest": "true",  
    "Domain": "Test"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya QLDB Amazon di Panduan Pengembang Amazon QLDB](#).

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## stream-journal-to-kinesis

Contoh kode berikut menunjukkan cara menggunakan `stream-journal-to-kinesis`.

## AWS CLI

Contoh 1: Untuk mengalirkan data jurnal ke Kinesis Data Streams menggunakan file input

`stream-journal-to-kinesis` Contoh berikut membuat aliran data jurnal dalam rentang tanggal dan waktu tertentu dari buku besar dengan nama `myExampleLedger`. Aliran mengirimkan data ke aliran data Amazon Kinesis tertentu.

```
aws qlldb stream-journal-to-kinesis \  
  --ledger-name myExampleLedger \  
  --inclusive-start-time 2020-05-29T00:00:00Z \  
  --exclusive-end-time 2020-05-29T23:59:59Z \  
  --role-arn arn:aws:iam::123456789012:role/my-kinesis-stream-role \  
  --kinesis-configuration file://my-kinesis-config.json \  
  --stream-name myExampleLedger-stream
```

Isi dari `my-kinesis-config.json`:

```
{  
  "StreamArn": "arn:aws:kinesis:us-east-1:123456789012:stream/stream-for-qlldb",  
  "AggregationEnabled": true  
}
```

Output:

```
{  
  "StreamId": "7ISCKqwe4y25YyHLzYUFaf"  
}
```

Untuk informasi selengkapnya, lihat [Streaming data jurnal dari Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

Contoh 2: Untuk mengalirkan data jurnal ke Kinesis Data Streams menggunakan sintaks singkatan

`stream-journal-to-kinesis` Contoh berikut membuat aliran data jurnal dalam rentang tanggal dan waktu tertentu dari buku besar dengan nama `myExampleLedger`. Aliran mengirimkan data ke aliran data Amazon Kinesis tertentu.

```
aws qlldb stream-journal-to-kinesis \  
  --ledger-name myExampleLedger \  
  --stream-name myExampleLedger-stream
```

```
--inclusive-start-time 2020-05-29T00:00:00Z \  
--exclusive-end-time 2020-05-29T23:59:59Z \  
--role-arn arn:aws:iam::123456789012:role/my-kinesis-stream-role \  
--stream-name myExampleLedger-stream \  
--kinesis-configuration StreamArn=arn:aws:kinesis:us-east-1:123456789012:stream/  
stream-for-qldb,AggregationEnabled=true
```

Output:

```
{  
  "StreamId": "7ISckqwe4y25YyHLzYUFAf"  
}
```

Untuk informasi selengkapnya, lihat [Streaming data jurnal dari Amazon QLDB](#) di Panduan Pengembang Amazon QLDB.

- Untuk detail API, lihat [StreamJournalToKinesis](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai buku besar

`tag-resource` Contoh berikut menambahkan satu set tag ke buku besar tertentu.

```
aws qldb tag-resource \  
--resource-arn arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger \  
--tags IsTest=true,Domain=Test
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya QLDB Amazon di Panduan Pengembang Amazon QLDB](#).

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag dengan kunci tag tertentu dari buku besar tertentu.

```
aws qlldb untag-resource \  
  --resource-arn arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger \  
  --tag-keys IsTest Domain
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya QLDB Amazon di Panduan Pengembang Amazon QLDB](#).

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-ledger-permissions-mode

Contoh kode berikut menunjukkan cara menggunakan `update-ledger-permissions-mode`.

### AWS CLI

Contoh 1: Untuk memperbarui mode izin buku besar ke STANDARD

`update-ledger-permissions-mode` Contoh berikut menetapkan modus STANDARD izin untuk buku besar yang ditentukan.

```
aws qlldb update-ledger-permissions-mode \  
  --name myExampleLedger \  
  --permissions-mode STANDARD
```

Output:

```
{  
  "Name": "myExampleLedger",  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "PermissionsMode": "STANDARD"  
}
```

Contoh 2: Untuk memperbarui mode izin buku besar ke ALLOW\_ALL

`update-ledger-permissions-mode` Contoh berikut menetapkan modus `ALLOW_ALL` izin untuk buku besar yang ditentukan.

```
aws qlldb update-ledger-permissions-mode \  
  --name myExampleLedger \  
  --permissions-mode ALLOW_ALL
```

Output:

```
{  
  "Name": "myExampleLedger",  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "PermissionsMode": "ALLOW_ALL"  
}
```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Buku Besar QLDB Amazon di Panduan Pengembang QLDB Amazon](#).

- Untuk detail API, lihat [UpdateLedgerPermissionsMode](#) di Referensi AWS CLI Perintah.

## update-ledger

Contoh kode berikut menunjukkan cara menggunakan `update-ledger`.

### AWS CLI

Contoh 1: Untuk memperbarui properti perlindungan penghapusan buku besar

`update-ledger` Contoh berikut memperbarui buku besar yang ditentukan untuk menonaktifkan fitur perlindungan penghapusan.

```
aws qlldb update-ledger \  
  --name myExampleLedger \  
  --no-deletion-protection
```

Output:

```
{  
  "CreationDateTime": 1568839243.951,  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "DeletionProtection": false,  
  "Name": "myExampleLedger",
```



```
"State": "ACTIVE"  
}
```

Contoh 2: Untuk memperbarui kunci AWS KMS buku besar ke kunci yang dikelola pelanggan update-`ledger` Contoh berikut memperbarui buku besar yang ditentukan untuk menggunakan kunci KMS yang dikelola pelanggan untuk enkripsi saat istirahat.

```
aws qlldb update-ledger \  
  --name myExampleLedger \  
  --kms-key arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111
```

Output:

```
{  
  "CreationDateTime": 1568839243.951,  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "DeletionProtection": false,  
  "Name": "myExampleLedger",  
  "State": "ACTIVE",  
  "EncryptionDescription": {  
    "KmsKeyArn": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-  
cdef-EXAMPLE11111",  
    "EncryptionStatus": "UPDATING"  
  }  
}
```

Contoh 3: Untuk memperbarui kunci AWS KMS buku besar ke kunci yang dimiliki AWS update-`ledger` Contoh berikut memperbarui buku besar yang ditentukan untuk menggunakan kunci KMS yang AWS dimiliki untuk enkripsi saat istirahat.

```
aws qlldb update-ledger \  
  --name myExampleLedger \  
  --kms-key AWS_OWNED_KMS_KEY
```

Output:

```
{  
  "CreationDateTime": 1568839243.951,  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",
```

```
"DeletionProtection": false,  
"Name": "myExampleLedger",  
"State": "ACTIVE",  
"EncryptionDescription": {  
  "KmsKeyArn": "AWS_OWNED_KMS_KEY",  
  "EncryptionStatus": "UPDATING"  
}  
}
```

Untuk informasi selengkapnya, lihat [Operasi Dasar untuk Buku Besar QLDB Amazon di Panduan Pengembang QLDB Amazon](#).

- Untuk detail API, lihat [UpdateLedger](#) di Referensi AWS CLI Perintah.

## Contoh Amazon RDS menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon RDS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-option-to-option-group**

Contoh kode berikut menunjukkan cara menggunakan `add-option-to-option-group`.

AWS CLI

Untuk menambahkan opsi ke grup opsi

`add-option-to-option-group` Contoh berikut menambahkan opsi ke grup opsi yang ditentukan.

```
aws rds add-option-to-option-group \
  --option-group-name myoptiongroup \
  --options OptionName=OEM,Port=5500,DBSecurityGroupMemberships=default \
  --apply-immediately
```

Output:

```
{
  "OptionGroup": {
    "OptionGroupName": "myoptiongroup",
    "OptionGroupDescription": "Test Option Group",
    "EngineName": "oracle-ee",
    "MajorEngineVersion": "12.1",
    "Options": [
      {
        "OptionName": "Timezone",
        "OptionDescription": "Change time zone",
        "Persistent": true,
        "Permanent": false,
        "OptionSettings": [
          {
            "Name": "TIME_ZONE",
            "Value": "Australia/Sydney",
            "DefaultValue": "UTC",
            "Description": "Specifies the timezone the user wants to
change the system time to",
            "ApplyType": "DYNAMIC",
            "DataType": "STRING",
            "AllowedValues": "Africa/Cairo,Africa/Casablanca,Africa/
Harare,Africa/Lagos,Africa/Luanda,Africa/Monrovia,Africa/Nairobi,Africa/
Tripoli,Africa/Windhoek,America/Araguaina,America/Argentina/Buenos_Aires,America/
Asuncion,America/Bogota,America/Caracas,America/Chicago,America/Chihuahua,America/
Cuiaba,America/Denver,America/Detroit,America/Fortaleza,America/Godthab,America/
Guatemala,America/Halifax,America/Lima,America/Los_Angeles,America/Manaus,America/
Matamoros,America/Mexico_City,America/Monterrey,America/Montevideo,America/
New_York,America/Phoenix,America/Santiago,America/Sao_Paulo,America/Tijuana,America/
Toronto,Asia/Amman,Asia/Ashgabat,Asia/Baghdad,Asia/Baku,Asia/Bangkok,Asia/
Beirut,Asia/Calcutta,Asia/Damascus,Asia/Dhaka,Asia/Hong_Kong,Asia/Irkutsk,Asia/
Jakarta,Asia/Jerusalem,Asia/Kabul,Asia/Karachi,Asia/Kathmandu,Asia/Kolkata,Asia/
Krasnoyarsk,Asia/Magadan,Asia/Manila,Asia/Muscat,Asia/Novosibirsk,Asia/Rangoon,Asia/
```

```

Riyadh,Asia/Seoul,Asia/Shanghai,Asia/Singapore,Asia/Taipei,Asia/Tehran,Asia/
Tokyo,Asia/Ulaanbaatar,Asia/Vladivostok,Asia/Yakutsk,Asia/Yerevan,Atlantic/
Azores,Atlantic/Cape_Verde,Australia/Adelaide,Australia/Brisbane,Australia/
Darwin,Australia/Eucla,Australia/Hobart,Australia/Lord_Howe,Australia/
Perth,Australia/Sydney,Brazil/DeNoronha,Brazil/East,Canada/Newfoundland,Canada/
Saskatchewan,Etc/GMT-3,Europe/Amsterdam,Europe/Athens,Europe/Berlin,Europe/
Dublin,Europe/Helsinki,Europe/Kaliningrad,Europe/London,Europe/Madrid,Europe/
Moscow,Europe/Paris,Europe/Prague,Europe/Rome,Europe/Sarajevo,Pacific/Apia,Pacific/
Auckland,Pacific/Chatham,Pacific/Fiji,Pacific/Guam,Pacific/Honolulu,Pacific/
Kiritimati,Pacific/Marquesas,Pacific/Samoa,Pacific/Tongatapu,Pacific/Wake,US/
Alaska,US/Central,US/East-Indiana,US/Eastern,US/Pacific,UTC",
        "IsModifiable": true,
        "IsCollection": false
    }
],
"DBSecurityGroupMemberships": [],
"VpcSecurityGroupMemberships": []
},
{
    "OptionName": "OEM",
    "OptionDescription": "Oracle 12c EM Express",
    "Persistent": false,
    "Permanent": false,
    "Port": 5500,
    "OptionSettings": [],
    "DBSecurityGroupMemberships": [
        {
            "DBSecurityGroupName": "default",
            "Status": "authorized"
        }
    ],
    "VpcSecurityGroupMemberships": []
}
],
"AllowsVpcAndNonVpcInstanceMemberships": false,
"OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:myoptiongroup"
}
}

```

Untuk informasi selengkapnya, lihat [Menambahkan Opsi ke Grup Opsi](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [AddOptionToOptionGroup](#) di Referensi AWS CLI Perintah.

## add-role-to-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `add-role-to-db-cluster`.

### AWS CLI

Untuk mengaitkan peran AWS Identity and Access Management (IAM) dengan cluster DB

`add-role-to-db-cluster` Contoh berikut mengaitkan peran dengan cluster DB.

```
aws rds add-role-to-db-cluster \  
  --db-cluster-identifier mydbcluster \  
  --role-arn arn:aws:iam::123456789012:role/RDSLoadFromS3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan peran IAM dengan kluster DB MySQL Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [AddRoleToDbCluster](#) di Referensi AWS CLI Perintah.

## add-role-to-db-instance

Contoh kode berikut menunjukkan cara menggunakan `add-role-to-db-instance`.

### AWS CLI

Untuk mengaitkan peran AWS Identity and Access Management (IAM) dengan instans DB

`add-role-to-db-instance` Contoh berikut menambahkan peran ke instance Oracle DB bernama `test-instance`.

```
aws rds add-role-to-db-instance \  
  --db-instance-identifier test-instance \  
  --feature-name S3_INTEGRATION \  
  --role-arn arn:aws:iam::111122223333:role/rds-s3-integration-role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Prasyarat untuk Integrasi Oracle Amazon RDS dengan Amazon S3 di Panduan Pengguna](#) Amazon RDS.

- Untuk detail API, lihat [AddRoleToDbInstance](#) di Referensi AWS CLI Perintah.

## add-source-identifier-to-subscription

Contoh kode berikut menunjukkan cara menggunakan `add-source-identifier-to-subscription`.

### AWS CLI

Untuk menambahkan pengenal sumber ke langganan

`add-source-identifier` Contoh berikut menambahkan pengenal sumber lain ke langganan yang ada.

```
aws rds add-source-identifier-to-subscription \  
  --subscription-name my-instance-events \  
  --source-identifier test-instance-repl
```

### Output:

```
{  
  "EventSubscription": {  
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",  
    "CustSubscriptionId": "my-instance-events",  
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-  
events",  
    "Enabled": false,  
    "Status": "modifying",  
    "EventCategoriesList": [  
      "backup",  
      "recovery"  
    ],  
    "CustomerAwsId": "123456789012",  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",  
    "SourceType": "db-instance",  
    "SourceIdsList": [  
      "test-instance",  
      "test-instance-repl"  
    ]  
  }  
}
```

- Untuk detail API, lihat [AddSourceIdentifierToSubscription](#) di Referensi AWS CLI Perintah.

## add-tags-to-resource

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber daya

`add-tags-to-resource` Contoh berikut menambahkan tag ke database RDS.

```
aws rds add-tags-to-resource \
  --resource-name arn:aws:rds:us-east-1:123456789012:db:database-mysql \
  --tags "[{\"Key\": \"Name\", \"Value\": \"MyDatabase\"}, {\"Key\": \"Environment\", \"Value\": \"test\"}]"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Amazon RDS](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [AddTagsToResource](#) di Referensi AWS CLI Perintah.

## apply-pending-maintenance-action

Contoh kode berikut menunjukkan cara menggunakan `apply-pending-maintenance-action`.

### AWS CLI

Untuk menerapkan tindakan pemeliharaan yang tertunda

`apply-pending-maintenance-action` Contoh berikut menerapkan tindakan pemeliharaan tertunda untuk cluster DB.

```
aws rds apply-pending-maintenance-action \
  --resource-identifier arn:aws:rds:us-east-1:123456789012:cluster:my-db-cluster \
  --apply-action system-update \
  --opt-in-type immediate
```

Output:

```
{
  "ResourcePendingMaintenanceActions": {
    "ResourceIdentifier": "arn:aws:rds:us-east-1:123456789012:cluster:my-db-cluster",
    "PendingMaintenanceActionDetails": [
      {
        "Action": "system-update",
        "OptInStatus": "immediate",
        "CurrentApplyDate": "2021-01-23T01:07:36.100Z",
        "Description": "Upgrade to Aurora PostgreSQL 3.3.2"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Memelihara instans DB](#) di Panduan Pengguna Amazon RDS dan [Memelihara klaster DB Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ApplyPendingMaintenanceAction](#) di Referensi AWS CLI Perintah.

## authorize-db-security-group-ingress

Contoh kode berikut menunjukkan cara menggunakan `authorize-db-security-group-ingress`.

### AWS CLI

Untuk mengaitkan peran AWS Identity and Access Management (IAM) dengan instans DB

`authorize-db-security-group-ingress` Contoh berikut mengkonfigurasi grup keamanan default dengan aturan masuk untuk rentang IP CIDR 192.0.2.0/24.

```
aws rds authorize-db-security-group-ingress \
  --db-security-group-name default \
  --cidrip 192.0.2.0/24
```

### Output:

```
{
  "DBSecurityGroup": {
    "OwnerId": "123456789012",
    "DBSecurityGroupName": "default",
```



```

    "DBSecurityGroupDescription": "default",
    "EC2SecurityGroups": [],
    "IPRanges": [
      {
        "Status": "authorizing",
        "CIDRIP": "192.0.2.0/24"
      }
    ],
    "DBSecurityGroupArn": "arn:aws:rds:us-east-1:111122223333:secgrp:default"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengotorisasi Akses Jaringan ke Grup Keamanan DB dari Rentang IP](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [AuthorizeDbSecurityGroupIngress](#) di Referensi AWS CLI Perintah.

## backtrack-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `backtrack-db-cluster`.

### AWS CLI

Untuk mundur cluster Aurora DB

`backtrack-db-cluster` Contoh berikut memutarbalikkan cluster sampel cluster DB yang ditentukan ke 19 Maret 2018, pukul 10 pagi.

```
aws rds backtrack-db-cluster --db-cluster-identifier sample-cluster --backtrack-to 2018-03-19T10:00:00+00:00
```

Perintah ini mengeluarkan blok JSON yang mengakui perubahan pada sumber daya RDS.

- Untuk detail API, lihat [BacktrackDbCluster](#) di Referensi AWS CLI Perintah.

## cancel-export-task

Contoh kode berikut menunjukkan cara menggunakan `cancel-export-task`.

### AWS CLI

Untuk membatalkan ekspor snapshot ke Amazon S3

`cancel-export-task` Contoh berikut membatalkan tugas ekspor yang sedang berlangsung yang mengekspor snapshot ke Amazon S3.

```
aws rds cancel-export-task \  
  --export-task-identifier my-s3-export-1
```

Output:

```
{  
  "ExportTaskIdentifier": "my-s3-export-1",  
  "SourceArn": "arn:aws:rds:us-east-1:123456789012:snapshot:publisher-final-  
snapshot",  
  "SnapshotTime": "2019-03-24T20:01:09.815Z",  
  "S3Bucket": "mybucket",  
  "S3Prefix": "",  
  "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/export-snap-S3-role",  
  "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/abcd0000-7bfd-4594-af38-  
aabbccddeeff",  
  "Status": "CANCELING",  
  "PercentProgress": 0,  
  "TotalExtractedDataInGB": 0  
}
```

Untuk informasi selengkapnya, lihat [Membatalkan tugas ekspor snapshot](#) di Panduan Pengguna Amazon RDS atau [Membatalkan tugas ekspor snapshot di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CancelExportTask](#) di Referensi AWS CLI Perintah.

## copy-db-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `copy-db-cluster-parameter-group`.

AWS CLI

Untuk menyalin grup parameter cluster DB

`copy-db-cluster-parameter-group` Contoh berikut membuat salinan kelompok parameter cluster DB.

```
aws rds copy-db-cluster-parameter-group \  
  --source-db-cluster-parameter-group-identifier mydbclusterpg \  
  --target-db-cluster-parameter-group-identifier mydbclusterpgcopy \  
  --source-db-cluster-parameter-group-identifier mydbclusterpgcopy \  
  --target-db-cluster-parameter-group-identifier mydbclusterpgcopy
```

```
--target-db-cluster-parameter-group-description "Copy of mydbclusterpg parameter group"
```

Output:

```
{
  "DBClusterParameterGroup": {
    "DBClusterParameterGroupName": "mydbclusterpgcopy",
    "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-pg:mydbclusterpgcopy",
    "DBParameterGroupFamily": "aurora-mysql5.7",
    "Description": "Copy of mydbclusterpg parameter group"
  }
}
```

Untuk informasi selengkapnya, lihat [Menyalin Grup Parameter Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CopyDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## copy-db-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `copy-db-cluster-snapshot`.

### AWS CLI

Untuk menyalin snapshot cluster DB

`copy-db-cluster-snapshot` Contoh berikut membuat salinan snapshot cluster DB, termasuk tag-nya.

```
aws rds copy-db-cluster-snapshot \
  --source-db-cluster-snapshot-identifier arn:aws:rds:us-east-1:123456789012:cluster-snapshot:rds:myaurora-2019-06-04-09-16
  --target-db-cluster-snapshot-identifier myclustersnapshotcopy \
  --copy-tags
```

Output:

```
{
  "DBClusterSnapshot": {
```

```

    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1e"
    ],
    "DBClusterSnapshotIdentifier": "myclustersnapshotcopy",
    "DBClusterIdentifier": "myaurora",
    "SnapshotCreateTime": "2019-06-04T09:16:42.649Z",
    "Engine": "aurora-mysql",
    "AllocatedStorage": 0,
    "Status": "available",
    "Port": 0,
    "VpcId": "vpc-6594f31c",
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
    "MasterUsername": "myadmin",
    "EngineVersion": "5.7.mysql_aurora.2.04.2",
    "LicenseModel": "aurora-mysql",
    "SnapshotType": "manual",
    "PercentProgress": 100,
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:myclustersnapshotcopy",
    "IAMDatabaseAuthenticationEnabled": false
  }
}

```

Untuk informasi selengkapnya, lihat [Menyalin Snapshot di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [CopyDbClusterSnapshot](#) di Referensi AWS CLI Perintah.

## copy-db-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `copy-db-parameter-group`.

### AWS CLI

Untuk menyalin grup parameter cluster DB

`copy-db-parameter-group` Contoh berikut membuat salinan grup parameter DB.

```

aws rds copy-db-parameter-group \
  --source-db-parameter-group-identifier mydbpg \

```

```
--target-db-parameter-group-identifier mydbpgcopy \
--target-db-parameter-group-description "Copy of mydbpg parameter group"
```

Output:

```
{
  "DBParameterGroup": {
    "DBParameterGroupName": "mydbpgcopy",
    "DBParameterGroupArn": "arn:aws:rds:us-east-1:814387698303:pg:mydbpgcopy",
    "DBParameterGroupFamily": "mysql5.7",
    "Description": "Copy of mydbpg parameter group"
  }
}
```

Untuk informasi selengkapnya, lihat [Menyalin Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [CopyDbParameterGroup](#) di Referensi AWS CLI Perintah.

## copy-db-snapshot

Contoh kode berikut menunjukkan cara menggunakan `copy-db-snapshot`.

### AWS CLI

Untuk menyalin snapshot DB

`copy-db-snapshot` Contoh berikut membuat salinan snapshot DB.

```
aws rds copy-db-snapshot \
--source-db-snapshot-identifier rds:database-mysql-2019-06-06-08-38
--target-db-snapshot-identifier mydbsnapshotcopy
```

Output:

```
{
  "DBSnapshot": {
    "VpcId": "vpc-6594f31c",
    "Status": "creating",
    "Encrypted": true,
    "SourceDBSnapshotIdentifier": "arn:aws:rds:us-east-1:123456789012:snapshot:rds:database-mysql-2019-06-06-08-38",
  }
}
```

```

    "MasterUsername": "admin",
    "Iops": 1000,
    "Port": 3306,
    "LicenseModel": "general-public-license",
    "DBSnapshotArn": "arn:aws:rds:us-
east-1:123456789012:snapshot:mydbsnapshotcopy",
    "EngineVersion": "5.6.40",
    "OptionGroupName": "default:mysql-5-6",
    "ProcessorFeatures": [],
    "Engine": "mysql",
    "StorageType": "io1",
    "DbiResourceId": "db-ZI7UJ5BLKMBYFGX7FDENCKADC4",
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
    "SnapshotType": "manual",
    "IAMDatabaseAuthenticationEnabled": false,
    "SourceRegion": "us-east-1",
    "DBInstanceIdentifier": "database-mysql",
    "InstanceCreateTime": "2019-04-30T15:45:53.663Z",
    "AvailabilityZone": "us-east-1f",
    "PercentProgress": 0,
    "AllocatedStorage": 100,
    "DBSnapshotIdentifier": "mydbsnapshotcopy"
  }
}

```

Untuk informasi selengkapnya, lihat [Menyalin Snapshot](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [CopyDbSnapshot](#) di Referensi AWS CLI Perintah.

## copy-option-group

Contoh kode berikut menunjukkan cara menggunakan `copy-option-group`.

### AWS CLI

Untuk menyalin grup opsi

`copy-option-group` Contoh berikut membuat salinan grup opsi.

```

aws rds copy-option-group \
  --source-option-group-identifier myoptiongroup \
  --target-option-group-identifier new-option-group \
  --target-option-group-description "My option group copy"

```

**Output:**

```
{
  "OptionGroup": {
    "Options": [],
    "OptionGroupName": "new-option-group",
    "MajorEngineVersion": "11.2",
    "OptionGroupDescription": "My option group copy",
    "AllowsVpcAndNonVpcInstanceMemberships": true,
    "EngineName": "oracle-ee",
    "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:new-option-group"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Salinan Grup Opsi](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [CopyOptionGroup](#) di Referensi AWS CLI Perintah.

**create-blue-green-deployment**

Contoh kode berikut menunjukkan cara menggunakan `create-blue-green-deployment`.

**AWS CLI**

Contoh 1: Untuk membuat penerapan biru/hijau untuk RDS untuk instance MySQL DB

`create-blue-green-deployment` Contoh berikut membuat penyebaran biru/hijau untuk instance MySQL DB.

```
aws rds create-blue-green-deployment \
  --blue-green-deployment-name bgd-cli-test-instance \
  --source arn:aws:rds:us-east-1:123456789012:db:my-db-instance \
  --target-engine-version 8.0 \
  --target-db-parameter-group-name mysql-80-group
```

**Output:**

```
{
  "BlueGreenDeployment": {
    "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
    "BlueGreenDeploymentName": "bgd-cli-test-instance",
```

```
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3"
      }
    ],
    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "PENDING"
      },
      {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "PENDING"
      },
      {
        "Name": "CONFIGURE_BACKUPS",
        "Status": "PENDING"
      },
      {
        "Name": "CREATING_TOPOLOGY_OF_SOURCE",
        "Status": "PENDING"
      }
    ],
    "Status": "PROVISIONING",
    "CreateTime": "2022-02-25T21:18:51.183000+00:00"
  }
}
```



Untuk informasi selengkapnya, lihat [Membuat penerapan biru/hijau di Panduan Pengguna Amazon RDS](#).

Contoh 2: Untuk membuat penyebaran biru/hijau untuk cluster DB MySQL Aurora

create-blue-green-deployment Contoh berikut membuat penyebaran biru/hijau untuk cluster DB MySQL Aurora.

```
aws rds create-blue-green-deployment \  
  --blue-green-deployment-name my-blue-green-deployment \  
  --source arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster \  
  --target-engine-version 8.0 \  
  --target-db-cluster-parameter-group-name ams-80-binlog-enabled \  
  --target-db-parameter-group-name mysql-80-cluster-group
```

Output:

```
{  
  "BlueGreenDeployment": {  
    "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",  
    "BlueGreenDeploymentName": "my-blue-green-deployment",  
    "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-  
cluster",  
    "SwitchoverDetails": [  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-  
mysql-cluster",  
        "Status": "PROVISIONING"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-  
cluster-1",  
        "Status": "PROVISIONING"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-  
cluster-2",  
        "Status": "PROVISIONING"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-  
cluster-3",  
        "Status": "PROVISIONING"  
      }  
    ]  
  }  
}
```

```

    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-
excluded-member-endpoint",
      "Status": "PROVISIONING"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-
reader-endpoint",
      "Status": "PROVISIONING"
    }
  ],
  "Tasks": [
    {
      "Name": "CREATING_READ_REPLICA_OF_SOURCE",
      "Status": "PENDING"
    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "PENDING"
    },
    {
      "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
      "Status": "PENDING"
    },
    {
      "Name": "CREATE_CUSTOM_ENDPOINTS",
      "Status": "PENDING"
    }
  ],
  "Status": "PROVISIONING",
  "CreateTime": "2022-02-25T21:12:00.288000+00:00"
}
}

```

Untuk informasi selengkapnya, lihat [Membuat penerapan biru/hijau](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateBlueGreenDeployment](#) di Referensi AWS CLI Perintah.

## create-db-cluster-endpoint

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster-endpoint`.

## AWS CLI

Untuk membuat endpoint cluster DB kustom

`create-db-cluster-endpoint` Contoh berikut membuat endpoint cluster DB kustom dan mengaitkannya dengan cluster Aurora DB yang ditentukan.

```
aws rds create-db-cluster-endpoint \  
  --db-cluster-endpoint-identifier mycustomendpoint \  
  --endpoint-type reader \  
  --db-cluster-identifier mydbcluster \  
  --static-members dbinstance1 dbinstance2
```

Output:

```
{  
  "DBClusterEndpointIdentifier": "mycustomendpoint",  
  "DBClusterIdentifier": "mydbcluster",  
  "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",  
  "Endpoint": "mycustomendpoint.cluster-custom-cnpexample.us-  
east-1.rds.amazonaws.com",  
  "Status": "creating",  
  "EndpointType": "CUSTOM",  
  "CustomEndpointType": "READER",  
  "StaticMembers": [  
    "dbinstance1",  
    "dbinstance2"  
  ],  
  "ExcludedMembers": [],  
  "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-  
endpoint:mycustomendpoint"  
}
```

Untuk informasi selengkapnya, lihat [Manajemen Koneksi Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateDbClusterEndpoint](#) di Referensi AWS CLI Perintah.

### **create-db-cluster-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster-parameter-group`.

## AWS CLI

Untuk membuat grup parameter cluster DB

`create-db-cluster-parameter-group` Contoh berikut membuat kelompok parameter cluster DB.

```
aws rds create-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name mydbclusterparametergroup \  
  --db-parameter-group-family aurora5.6 \  
  --description "My new cluster parameter group"
```

Output:

```
{  
  "DBClusterParameterGroup": {  
    "DBClusterParameterGroupName": "mydbclusterparametergroup",  
    "DBParameterGroupFamily": "aurora5.6",  
    "Description": "My new cluster parameter group",  
    "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-  
pg:mydbclusterparametergroup"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Parameter Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## **create-db-cluster-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster-snapshot`.

### AWS CLI

Untuk membuat snapshot cluster DB

`create-db-cluster-snapshot` Contoh berikut membuat snapshot cluster DB.

```
aws rds create-db-cluster-snapshot \  
  --db-cluster-identifier mydbcluster \  
  --db-cluster-snapshot-identifier mydbclustersnapshot
```

**Output:**

```
{
  "DBClusterSnapshot": {
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1e"
    ],
    "DBClusterSnapshotIdentifier": "mydbclustersnapshot",
    "DBClusterIdentifier": "mydbcluster",
    "SnapshotCreateTime": "2019-06-18T21:21:00.469Z",
    "Engine": "aurora-mysql",
    "AllocatedStorage": 1,
    "Status": "creating",
    "Port": 0,
    "VpcId": "vpc-6594f31c",
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
    "MasterUsername": "myadmin",
    "EngineVersion": "5.7.mysql_aurora.2.04.2",
    "LicenseModel": "aurora-mysql",
    "SnapshotType": "manual",
    "PercentProgress": 0,
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-snapshot:mydbclustersnapshot",
    "IAMDatabaseAuthenticationEnabled": false
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Snapshot Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateDbClusterSnapshot](#) di Referensi AWS CLI Perintah.

**create-db-cluster**

Contoh kode berikut menunjukkan cara menggunakan `create-db-cluster`.

**AWS CLI**

Contoh 1: Untuk membuat cluster DB MySQL 5.7 yang kompatibel

`create-db-cluster` Contoh berikut membuat cluster DB MySQL 5.7 kompatibel menggunakan versi mesin default. Ganti kata sandi sampel `secret99` dengan kata sandi yang aman. Saat Anda menggunakan konsol untuk membuat cluster DB, Amazon RDS secara otomatis membuat instans DB penulis untuk cluster DB Anda. Namun, ketika Anda menggunakan AWS CLI untuk membuat cluster DB, Anda harus secara eksplisit membuat instance DB penulis untuk cluster DB Anda menggunakan perintah CLI. `create-db-instance` AWS

```
aws rds create-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --engine aurora-mysql \  
  --engine-version 5.7 \  
  --master-username admin \  
  --master-user-password secret99 \  
  --db-subnet-group-name default \  
  --vpc-security-group-ids sg-0b9130572daf3dc16
```

Output:

```
{  
  "DBCluster": {  
    "DBSubnetGroup": "default",  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-0b9130572daf3dc16",  
        "Status": "active"  
      }  
    ],  
    "AllocatedStorage": 1,  
    "AssociatedRoles": [],  
    "PreferredBackupWindow": "09:12-09:42",  
    "ClusterCreateTime": "2023-02-27T23:21:33.048Z",  
    "DeletionProtection": false,  
    "IAMDatabaseAuthenticationEnabled": false,  
    "ReadReplicaIdentifiers": [],  
    "EngineMode": "provisioned",  
    "Engine": "aurora-mysql",  
    "StorageEncrypted": false,  
    "MultiAZ": false,  
    "PreferredMaintenanceWindow": "mon:04:31-mon:05:01",  
    "HttpEndpointEnabled": false,  
    "BackupRetentionPeriod": 1,  
    "DbClusterResourceId": "cluster-ANPAJ4AE5446DAEXAMPLE",
```

```

    "DBClusterIdentifier": "sample-cluster",
    "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1e"
    ],
    "MasterUsername": "master",
    "EngineVersion": "5.7.mysql_aurora.2.11.1",
    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-cluster",
    "DBClusterMembers": [],
    "Port": 3306,
    "Status": "creating",
    "Endpoint": "sample-cluster.cluster-cnpxexample.us-east-1.rds.amazonaws.com",
    "DBClusterParameterGroup": "default.aurora-mysql5.7",
    "HostedZoneId": "Z2R2ITUGPM61AM",
    "ReaderEndpoint": "sample-cluster.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "CopyTagsToSnapshot": false
  }
}

```

## Contoh 2: Untuk membuat cluster DB yang kompatibel dengan PostgreSQL

`create-db-cluster` Contoh berikut membuat cluster DB yang kompatibel dengan PostgreSQL menggunakan versi mesin default. Ganti kata sandi contoh `secret99` dengan kata sandi yang aman. Saat Anda menggunakan konsol untuk membuat cluster DB, Amazon RDS secara otomatis membuat instans DB penulis untuk cluster DB Anda. Namun, ketika Anda menggunakan AWS CLI untuk membuat cluster DB, Anda harus secara eksplisit membuat instance DB penulis untuk cluster DB Anda menggunakan perintah CLI. `create-db-instance AWS`

```

aws rds create-db-cluster \
  --db-cluster-identifier sample-pg-cluster \
  --engine aurora-postgresql \
  --master-username master \
  --master-user-password secret99 \
  --db-subnet-group-name default \
  --vpc-security-group-ids sg-0b9130572daf3dc16

```

## Output:

```

{
  "DBCluster": {

```

```
    "Endpoint": "sample-pg-cluster.cluster-cnpxample.us-
east-1.rds.amazonaws.com",
    "HttpEndpointEnabled": false,
    "DBClusterMembers": [],
    "EngineMode": "provisioned",
    "CopyTagsToSnapshot": false,
    "HostedZoneId": "Z2R2ITUGPM61AM",
    "IAMDatabaseAuthenticationEnabled": false,
    "AllocatedStorage": 1,
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
        "Status": "active"
      }
    ],
    "DeletionProtection": false,
    "StorageEncrypted": false,
    "BackupRetentionPeriod": 1,
    "PreferredBackupWindow": "09:56-10:26",
    "ClusterCreateTime": "2023-02-27T23:26:08.371Z",
    "DBClusterParameterGroup": "default.aurora-postgresql13",
    "EngineVersion": "13.7",
    "Engine": "aurora-postgresql",
    "Status": "creating",
    "DBClusterIdentifier": "sample-pg-cluster",
    "MultiAZ": false,
    "Port": 5432,
    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-pg-
cluster",
    "AssociatedRoles": [],
    "DbClusterResourceId": "cluster-ANPAJ4AE5446DAEXAMPLE",
    "PreferredMaintenanceWindow": "wed:03:33-wed:04:03",
    "ReaderEndpoint": "sample-pg-cluster.cluster-ro-cnpxample.us-
east-1.rds.amazonaws.com",
    "MasterUsername": "master",
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1c"
    ],
    "ReadReplicaIdentifiers": [],
    "DBSubnetGroup": "default"
  }
```



```
}
```

Untuk informasi selengkapnya, lihat [Membuat klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [CreateDbCluster](#) di Referensi AWS CLI Perintah.

## **create-db-instance-read-replica**

Contoh kode berikut menunjukkan cara menggunakan `create-db-instance-read-replica`.

### AWS CLI

Untuk membuat instance DB, baca replika

Contoh ini membuat replika baca dari instance DB yang ada bernama `test-instance`. Replika baca diberi nama `test-instance-repl`.

```
aws rds create-db-instance-read-replica \  
  --db-instance-identifier test-instance-repl \  
  --source-db-instance-identifier test-instance
```

Output:

```
{  
  "DBInstance": {  
    "IAMDatabaseAuthenticationEnabled": false,  
    "MonitoringInterval": 0,  
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance-repl",  
    "ReadReplicaSourceDBInstanceIdentifier": "test-instance",  
    "DBInstanceIdentifier": "test-instance-repl",  
    ...some output truncated...  
  }  
}
```

- Untuk detail API, lihat [CreateDbInstanceReadReplica](#) di Referensi AWS CLI Perintah.

## **create-db-instance**

Contoh kode berikut menunjukkan cara menggunakan `create-db-instance`.

## AWS CLI

Untuk membuat instance DB

`create-db-instance` Contoh berikut menggunakan opsi yang diperlukan untuk meluncurkan instans DB baru.

```
aws rds create-db-instance \  
  --db-instance-identifier test-mysql-instance \  
  --db-instance-class db.t3.micro \  
  --engine mysql \  
  --master-username admin \  
  --master-user-password secret99 \  
  --allocated-storage 20
```

Output:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "test-mysql-instance",  
    "DBInstanceClass": "db.t3.micro",  
    "Engine": "mysql",  
    "DBInstanceStatus": "creating",  
    "MasterUsername": "admin",  
    "AllocatedStorage": 20,  
    "PreferredBackupWindow": "12:55-13:25",  
    "BackupRetentionPeriod": 1,  
    "DBSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-12345abc",  
        "Status": "active"  
      }  
    ],  
    "DBParameterGroups": [  
      {  
        "DBParameterGroupName": "default.mysql5.7",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ],  
    "DBSubnetGroup": {  
      "DBSubnetGroupName": "default",  
      "DBSubnetGroupDescription": "default",
```

```
"VpcId": "vpc-2ff2ff2f",
"SubnetGroupStatus": "Complete",
"Subnets": [
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2c"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2d"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2a"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2b"
    },
    "SubnetStatus": "Active"
  }
],
"PreferredMaintenanceWindow": "sun:08:07-sun:08:37",
"PendingModifiedValues": {
  "MasterUserPassword": "*****"
},
"MultiAZ": false,
"EngineVersion": "5.7.22",
"AutoMinorVersionUpgrade": true,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "general-public-license",
"OptionGroupMemberships": [
  {
```

```

        "OptionGroupName": "default:mysql-5-7",
        "Status": "in-sync"
    }
],
"PubliclyAccessible": true,
"StorageType": "gp2",
"DbInstancePort": 0,
"StorageEncrypted": false,
"DbiResourceId": "db-5555EXAMPLE44444444EXAMPLE",
"CACertificateIdentifier": "rds-ca-2019",
"DomainMemberships": [],
"CopyTagsToSnapshot": false,
"MonitoringInterval": 0,
"DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:test-mysql-
instance",
"IAMDatabaseAuthenticationEnabled": false,
"PerformanceInsightsEnabled": false,
"DeletionProtection": false,
"AssociatedRoles": []
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Instans Amazon RDS DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [createdBInstance](#) di Referensi Perintah.AWS CLI

## create-db-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `create-db-parameter-group`.

### AWS CLI

Untuk membuat grup parameter DB

`create-db-parameter-group` Contoh berikut membuat grup parameter DB.

```

aws rds create-db-parameter-group \
  --db-parameter-group-name mydbparametergroup \
  --db-parameter-group-family MySQL5.6 \
  --description "My new parameter group"

```

Output:

```
{
  "DBParameterGroup": {
    "DBParameterGroupName": "mydbparametergroup",
    "DBParameterGroupFamily": "mysql5.6",
    "Description": "My new parameter group",
    "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:mydbparametergroup"
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [CreateDB ParameterGroup](#) di AWS CLI Referensi Perintah.

## create-db-proxy-endpoint

Contoh kode berikut menunjukkan cara menggunakan `create-db-proxy-endpoint`.

### AWS CLI

Untuk membuat endpoint proxy DB untuk database RDS

`create-db-proxy-endpoint` Contoh berikut menciptakan titik akhir proxy DB.

```
aws rds create-db-proxy-endpoint \
  --db-proxy-name proxyExample \
  --db-proxy-endpoint-name "proxyep1" \
  --vpc-subnet-ids subnetgroup1 subnetgroup2
```

### Output:

```
{
  "DBProxyEndpoint": {
    "DBProxyEndpointName": "proxyep1",
    "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
    "DBProxyName": "proxyExample",
    "Status": "creating",
    "VpcId": "vpc-1234567",
    "VpcSecurityGroupIds": [
      "sg-1234",

```

```

        "sg-5678"
    ],
    "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
    ],
    "Endpoint": "proxyep1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",
    "TargetRole": "READ_WRITE",
    "IsDefault": false
    }
}

```

Untuk informasi selengkapnya, lihat [Membuat titik akhir proxy](#) di Panduan Pengguna Amazon RDS dan [Membuat titik akhir proxy](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateDbProxyEndpoint](#) di Referensi AWS CLI Perintah.

## create-db-proxy

Contoh kode berikut menunjukkan cara menggunakan `create-db-proxy`.

### AWS CLI

Untuk membuat proxy DB untuk database RDS

`create-db-proxy` Contoh berikut membuat proxy DB.

```

aws rds create-db-proxy \
  --db-proxy-name proxyExample \
  --engine-family MYSQL \
  --auth
  Description="proxydescription1",AuthScheme="SECRETS",SecretArn="arn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",IAMAuth="DISABLED",ClientPasswordAuthType="MYSQL"
  \
  --role-arn arn:aws:iam::123456789123:role/ProxyRole \
  --vpc-subnet-ids subnetgroup1 subnetgroup2

```

Output:

```
{
```

```

"DBProxy": {
  "DBProxyName": "proxyExample",
  "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
  "EngineFamily": "MYSQL",
  "VpcId": "vpc-1234567",
  "VpcSecuritytGroupIds": [
    "sg-1234",
    "sg-5678",
    "sg-9101"
  ],
  "VpcSubnetIds": [
    "subnetgroup1",
    "subnetgroup2"
  ],
  "Auth": "[
    {
      "Description": "proxydescription1",
      "AuthScheme": "SECRETS",
      "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxyssecret1-Abcd1e",
      "IAMAuth": "DISABLED"
    }
  ]",
  "RoleArn": "arn:aws:iam::12345678912:role/ProxyRole",
  "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
  "RequireTLS": false,
  "IdleClientTimeout": 1800,
  "DebuggingLogging": false,
  "CreateDate": "2023-04-05T16:09:33.452000+00:00",
  "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Membuat Proxy RDS](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateDbProxy](#) di Referensi AWS CLI Perintah.

## create-db-security-group

Contoh kode berikut menunjukkan cara menggunakan `create-db-security-group`.

## AWS CLI

Untuk membuat grup keamanan Amazon RDS DB

`create-db-security-group` Perintah berikut membuat grup keamanan Amazon RDS DB baru:

```
aws rds create-db-security-group --db-security-group-name mysecgroup --db-security-group-description "My Test Security Group"
```

Dalam contoh, grup keamanan DB baru diberi nama `mysecgroup` dan memiliki deskripsi.

Output:

```
{
  "DBSecurityGroup": {
    "OwnerId": "123456789012",
    "DBSecurityGroupName": "mysecgroup",
    "DBSecurityGroupDescription": "My Test Security Group",
    "VpcId": "vpc-a1b2c3d4",
    "EC2SecurityGroups": [],
    "IPRanges": [],
    "DBSecurityGroupArn": "arn:aws:rds:us-west-2:123456789012:secgrp:mysecgroup"
  }
}
```

- Untuk detail API, lihat [CreateDbSecurityGroup](#) di Referensi AWS CLI Perintah.

## `create-db-shard-group`

Contoh kode berikut menunjukkan cara menggunakan `create-db-shard-group`.

### AWS CLI

Contoh 1: Untuk membuat cluster DB utama Aurora PostgreSQL

`create-db-cluster` Contoh berikut membuat cluster DB primer Aurora PostgreSQL SQL yang kompatibel dengan Aurora Serverless v2 dan Aurora Limitless Database.

```
aws rds create-db-cluster \
  --db-cluster-identifier my-sv2-cluster \
```



```
--engine aurora-postgresql \  
--engine-version 15.2-limitless \  
--storage-type aurora-iopt1 \  
--serverless-v2-scaling-configuration MinCapacity=2,MaxCapacity=16 \  
--enable-limitless-database \  
--master-username myuser \  
--master-user-password mypassword \  
--enable-cloudwatch-logs-exports postgresql
```

## Output:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-2b",  
      "us-east-2c",  
      "us-east-2a"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DBClusterIdentifier": "my-sv2-cluster",  
    "DBClusterParameterGroup": "default.aurora-postgresql15",  
    "DBSubnetGroup": "default",  
    "Status": "creating",  
    "Endpoint": "my-sv2-cluster.cluster-cekyexample.us-  
east-2.rds.amazonaws.com",  
    "ReaderEndpoint": "my-sv2-cluster.cluster-ro-cekyexample.us-  
east-2.rds.amazonaws.com",  
    "MultiAZ": false,  
    "Engine": "aurora-postgresql",  
    "EngineVersion": "15.2-limitless",  
    "Port": 5432,  
    "MasterUsername": "myuser",  
    "PreferredBackupWindow": "06:05-06:35",  
    "PreferredMaintenanceWindow": "mon:08:25-mon:08:55",  
    "ReadReplicaIdentifiers": [],  
    "DBClusterMembers": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-#####",  
        "Status": "active"  
      }  
    ],  
  },  
}
```

```

    "HostedZoneId": "Z2XHWR1EXAMPLE",
    "StorageEncrypted": false,
    "DbClusterResourceId": "cluster-XYEDT6ML6FHIXH4Q2J1EXAMPLE",
    "DBClusterArn": "arn:aws:rds:us-east-2:123456789012:cluster:my-sv2-cluster",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "ClusterCreateTime": "2024-02-19T16:24:07.771000+00:00",
    "EnabledCloudwatchLogsExports": [
      "postgresql"
    ],
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": false,
    "CrossAccountClone": false,
    "DomainMemberships": [],
    "TagList": [],
    "StorageType": "aurora-iopt1",
    "AutoMinorVersionUpgrade": true,
    "ServerlessV2ScalingConfiguration": {
      "MinCapacity": 2.0,
      "MaxCapacity": 16.0
    },
    "NetworkType": "IPV4",
    "IOOptimizedNextAllowedModificationTime":
"2024-03-21T16:24:07.781000+00:00",
    "LimitlessDatabase": {
      "Status": "not-in-use",
      "MinRequiredACU": 96.0
    }
  }
}

```

## Contoh 2: Untuk membuat instance DB utama (penulis)

`create-db-instance` Contoh berikut membuat instance DB primer (penulis) Aurora Serverless v2. Saat Anda menggunakan konsol untuk membuat cluster DB, Amazon RDS secara otomatis membuat instans DB penulis untuk cluster DB Anda. Namun, ketika Anda menggunakan AWS CLI untuk membuat cluster DB, Anda harus secara eksplisit membuat instance DB penulis untuk cluster DB Anda menggunakan perintah CLI. `create-db-instance` AWS

```

aws rds create-db-instance \
  --db-instance-identifier my-sv2-instance \

```

```
--db-cluster-identifier my-sv2-cluster \  
--engine aurora-postgresql \  
--db-instance-class db.serverless
```

**Output:**

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "my-sv2-instance",  
    "DBInstanceClass": "db.serverless",  
    "Engine": "aurora-postgresql",  
    "DBInstanceStatus": "creating",  
    "MasterUsername": "myuser",  
    "AllocatedStorage": 1,  
    "PreferredBackupWindow": "06:05-06:35",  
    "BackupRetentionPeriod": 1,  
    "DBSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-#####",  
        "Status": "active"  
      }  
    ],  
    "DBParameterGroups": [  
      {  
        "DBParameterGroupName": "default.aurora-postgresql15",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ],  
    "DBSubnetGroup": {  
      "DBSubnetGroupName": "default",  
      "DBSubnetGroupDescription": "default",  
      "VpcId": "vpc-#####",  
      "SubnetGroupStatus": "Complete",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-#####",  
          "SubnetAvailabilityZone": {  
            "Name": "us-east-2c"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-#####",  
          "SubnetAvailabilityZone": {  
            "Name": "us-east-2c"  
          }  
        }  
      ],  
      "SubnetOutpost": {},  
      "SubnetStatus": "Active"  
    }  
  }  
}
```

```
        {
            "SubnetIdentifier": "subnet-#####",
            "SubnetAvailabilityZone": {
                "Name": "us-east-2a"
            },
            "SubnetOutpost": {},
            "SubnetStatus": "Active"
        },
        {
            "SubnetIdentifier": "subnet-#####",
            "SubnetAvailabilityZone": {
                "Name": "us-east-2b"
            },
            "SubnetOutpost": {},
            "SubnetStatus": "Active"
        }
    ]
},
"PreferredMaintenanceWindow": "fri:09:01-fri:09:31",
"PendingModifiedValues": {
    "PendingCloudwatchLogsExports": {
        "LogTypesToEnable": [
            "postgresql"
        ]
    }
},
"MultiAZ": false,
"EngineVersion": "15.2-limitless",
"AutoMinorVersionUpgrade": true,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "postgresql-license",
"OptionGroupMemberships": [
    {
        "OptionGroupName": "default:aurora-postgresql-15",
        "Status": "in-sync"
    }
],
"PubliclyAccessible": false,
"StorageType": "aurora-iopt1",
"DbInstancePort": 0,
"DBClusterIdentifier": "my-sv2-cluster",
"StorageEncrypted": false,
"DbiResourceId": "db-BIQTE3B3K3RM7M74SK5EXAMPLE",
"CACertificateIdentifier": "rds-ca-rsa2048-g1",
```

```

    "DomainMemberships": [],
    "CopyTagsToSnapshot": false,
    "MonitoringInterval": 0,
    "PromotionTier": 1,
    "DBInstanceArn": "arn:aws:rds:us-east-2:123456789012:db:my-sv2-instance",
    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": [],
    "TagList": [],
    "CustomerOwnedIpEnabled": false,
    "BackupTarget": "region",
    "NetworkType": "IPV4",
    "StorageThroughput": 0,
    "CertificateDetails": {
      "CAIdentifier": "rds-ca-rsa2048-g1"
    },
    "DedicatedLogVolume": false
  }
}

```

### Contoh 3: Untuk membuat grup pecahan DB

`create-db-shard-group` Contoh berikut membuat grup pecahan DB di cluster DB utama Aurora PostgreSQL Anda.

```

aws rds create-db-shard-group \
  --db-shard-group-identifier my-db-shard-group \
  --db-cluster-identifier my-sv2-cluster \
  --max-acu 768

```

### Output:

```

{
  "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
  "DBShardGroupIdentifier": "my-db-shard-group",
  "DBClusterIdentifier": "my-sv2-cluster",
  "MaxACU": 768.0,
  "ComputeRedundancy": 0,
  "Status": "creating",
  "PubliclyAccessible": false,
  "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Aurora Tanpa Server v2](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [CreateDbShardGroup](#) di Referensi AWS CLI Perintah.

## create-db-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-db-snapshot`.

### AWS CLI

Untuk membuat snapshot DB

`create-db-snapshot` Contoh berikut membuat snapshot DB.

```
aws rds create-db-snapshot \  
  --db-instance-identifier database-mysql \  
  --db-snapshot-identifier mydbsnapshot
```

Output:

```
{  
  "DBSnapshot": {  
    "DBSnapshotIdentifier": "mydbsnapshot",  
    "DBInstanceIdentifier": "database-mysql",  
    "Engine": "mysql",  
    "AllocatedStorage": 100,  
    "Status": "creating",  
    "Port": 3306,  
    "AvailabilityZone": "us-east-1b",  
    "VpcId": "vpc-6594f31c",  
    "InstanceCreateTime": "2019-04-30T15:45:53.663Z",  
    "MasterUsername": "admin",  
    "EngineVersion": "5.6.40",  
    "LicenseModel": "general-public-license",  
    "SnapshotType": "manual",  
    "Iops": 1000,  
    "OptionGroupName": "default:mysql-5-6",  
    "PercentProgress": 0,  
    "StorageType": "io1",  
    "Encrypted": true,  
  }  
}
```

```

    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
    "DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mydbsnapshot",
    "IAMDatabaseAuthenticationEnabled": false,
    "ProcessorFeatures": [],
    "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat Snapshot DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [CreatedBSnapshot](#) di Referensi Perintah.AWS CLI

## create-db-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `create-db-subnet-group`.

### AWS CLI

Untuk membuat grup subnet DB

`create-db-subnet-group` Contoh berikut membuat grup subnet DB yang disebut `mysubnetgroup` menggunakan subnet yang ada.

```

aws rds create-db-subnet-group \
  --db-subnet-group-name mysubnetgroup \
  --db-subnet-group-description "test DB subnet group" \
  --subnet-ids
  ["subnet-0a1dc4e1a6f123456","subnet-070dd7ecb3aaaaaaa","subnet-00f5b198bc0abcdef"]'

```

### Output:

```

{
  "DBSubnetGroup": {
    "DBSubnetGroupName": "mysubnetgroup",
    "DBSubnetGroupDescription": "test DB subnet group",
    "VpcId": "vpc-0f08e7610a1b2c3d4",
    "SubnetGroupStatus": "Complete",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-070dd7ecb3aaaaaaa",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2b"
        }
      }
    ],
  }
}

```

```

        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-00f5b198bc0abcdef",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2d"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-0a1dc4e1a6f123456",
        "SubnetAvailabilityZone": {
            "Name": "us-west-2b"
        },
        "SubnetStatus": "Active"
    }
],
    "DBSubnetGroupArn": "arn:aws:rds:us-
west-2:0123456789012:subgrp:mysubnetgroup"
}
}

```

Untuk informasi selengkapnya, lihat [Membuat Instans DB di VPC](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [CreateDbSubnetGroup](#) di Referensi AWS CLI Perintah.

## create-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `create-event-subscription`.

### AWS CLI

Untuk membuat langganan acara

`create-event-subscription` Contoh berikut membuat langganan untuk acara pencadangan dan pemulihan untuk instans DB di AWS akun saat ini. Pemberitahuan dikirim ke topik Layanan Pemberitahuan Sederhana Amazon, yang ditentukan oleh `--sns-topic-arn`.

```

aws rds create-event-subscription \
  --subscription-name my-instance-events \
  --source-type db-instance \
  --event-categories ["backup","recovery"] \

```



```
--sns-topic-arn arn:aws:sns:us-east-1:123456789012:interesting-events
```

Output:

```
{
  "EventSubscription": {
    "Status": "creating",
    "CustSubscriptionId": "my-instance-events",
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
    "EventCategoriesList": [
      "backup",
      "recovery"
    ],
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
    "CustomerAwsId": "123456789012",
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
    "SourceType": "db-instance",
    "Enabled": true
  }
}
```

- Untuk detail API, lihat [CreateEventSubscription](#) di Referensi AWS CLI Perintah.

## create-global-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-global-cluster`.

AWS CLI

Untuk membuat cluster DB global

`create-global-cluster` Contoh berikut membuat cluster DB global Aurora MySQL kompatibel baru.

```
aws rds create-global-cluster \
  --global-cluster-identifier myglobalcluster \
  --engine aurora-mysql
```

Output:

```
{
```

```
"GlobalCluster": {
  "GlobalClusterIdentifier": "myglobalcluster",
  "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",
  "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
  "Status": "available",
  "Engine": "aurora-mysql",
  "EngineVersion": "5.7.mysql_aurora.2.07.2",
  "StorageEncrypted": false,
  "DeletionProtection": false,
  "GlobalClusterMembers": []
}
```

Untuk informasi selengkapnya, lihat [Membuat database global Aurora di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [CreateGlobalCluster](#) di Referensi AWS CLI Perintah.

## create-option-group

Contoh kode berikut menunjukkan cara menggunakan `create-option-group`.

### AWS CLI

Untuk Membuat grup opsi Amazon RDS

`create-option-group` Perintah berikut membuat grup opsi Amazon RDS baru untuk Oracle Enterprise Edition versi 11.2`, is named ``MyOptionGroup dan menyertakan deskripsi.

```
aws rds create-option-group \
  --option-group-name MyOptionGroup \
  --engine-name oracle-ee \
  --major-engine-version 11.2 \
  --option-group-description "Oracle Database Manager Database Control"
```

Output:

```
{
  "OptionGroup": {
    "OptionGroupName": "myoptiongroup",
    "OptionGroupDescription": "Oracle Database Manager Database Control",
```

```

    "EngineName": "oracle-ee",
    "MajorEngineVersion": "11.2",
    "Options": [],
    "AllowsVpcAndNonVpcInstanceMemberships": true,
    "OptionGroupArn": "arn:aws:rds:us-west-2:123456789012:og:myoptiongroup"
  }
}

```

- Untuk detail API, lihat [CreateOptionGroup](#) di Referensi AWS CLI Perintah.

## delete-blue-green-deployment

Contoh kode berikut menunjukkan cara menggunakan `delete-blue-green-deployment`.

### AWS CLI

Contoh 1: Untuk menghapus sumber daya di lingkungan hijau untuk RDS untuk instance MySQL DB

`delete-blue-green-deployment` Contoh berikut menghapus sumber daya dalam lingkungan hijau untuk RDS untuk MySQL DB instance.

```

aws rds delete-blue-green-deployment \
  --blue-green-deployment-identifier bgd-v53303651eexfake \
  --delete-target

```

### Output:

```

{
  "BlueGreenDeployment": {
    "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
    "BlueGreenDeploymentName": "bgd-cli-test-instance",
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
    "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-rkfbpe",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-rkfbpe",
        "Status": "AVAILABLE"
      }
    ]
  }
}

```

```
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-j382ha",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ejv4ao",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-vlpz3t",
      "Status": "AVAILABLE"
    }
  ],
  "Tasks": [
    {
      "Name": "CREATING_READ_REPLICA_OF_SOURCE",
      "Status": "COMPLETED"
    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "COMPLETED"
    },
    {
      "Name": "CONFIGURE_BACKUPS",
      "Status": "COMPLETED"
    },
    {
      "Name": "CREATING_TOPOLOGY_OF_SOURCE",
      "Status": "COMPLETED"
    }
  ],
  "Status": "DELETING",
  "CreateTime": "2022-02-25T21:18:51.183000+00:00",
  "DeleteTime": "2022-02-25T22:25:31.331000+00:00"
```

```
}
}
```

Untuk informasi selengkapnya, lihat [Menghapus penerapan biru/hijau di Panduan Pengguna Amazon RDS](#).

Contoh 2: Untuk menghapus sumber daya di lingkungan hijau untuk cluster DB MySQL Aurora delete-blue-green-deployment Contoh berikut menghapus sumber daya di lingkungan hijau untuk cluster DB MySQL Aurora.

```
aws rds delete-blue-green-deployment \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake \
  --delete-target
```

Output:

```
{
  "BlueGreenDeployment": {
    "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
    "BlueGreenDeploymentName": "my-blue-green-deployment",
    "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
    "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1-green-gpmaxf",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2",
```

```
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2-green-j2oajq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3-green-mkxies",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-4sqjrq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-gwwzlg",
        "Status": "AVAILABLE"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "COMPLETED"
    }
],
```

```
    "Status": "DELETING",
    "CreateTime": "2022-02-25T21:12:00.288000+00:00",
    "DeleteTime": "2022-02-25T22:29:11.336000+00:00"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus penerapan biru/hijau](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteBlueGreenDeployment](#) di Referensi AWS CLI Perintah.

## delete-db-cluster-endpoint

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster-endpoint`.

### AWS CLI

Untuk menghapus titik akhir cluster DB kustom

`delete-db-cluster-endpoint` Contoh berikut menghapus titik akhir cluster DB kustom yang ditentukan.

```
aws rds delete-db-cluster-endpoint \
  --db-cluster-endpoint-identifier mycustomendpoint
```

Output:

```
{
  "DBClusterEndpointIdentifier": "mycustomendpoint",
  "DBClusterIdentifier": "mydbcluster",
  "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
  "Endpoint": "mycustomendpoint.cluster-custom-cnpxexample.us-east-1.rds.amazonaws.com",
  "Status": "deleting",
  "EndpointType": "CUSTOM",
  "CustomEndpointType": "READER",
  "StaticMembers": [
    "dbinstance1",
    "dbinstance2",
    "dbinstance3"
  ],
  "ExcludedMembers": [],
}
```

```
"DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}
```

Untuk informasi selengkapnya, lihat [Manajemen Koneksi Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteDbClusterEndpoint](#) di Referensi AWS CLI Perintah.

## **delete-db-cluster-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster-parameter-group`.

### AWS CLI

Untuk menghapus grup parameter cluster DB

`delete-db-cluster-parameter-group` Contoh berikut menghapus kelompok parameter cluster DB yang ditentukan.

```
aws rds delete-db-cluster-parameter-group \
  --db-cluster-parameter-group-name mydbclusterparametergroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB dan Grup Parameter Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## **delete-db-cluster-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster-snapshot`.

### AWS CLI

Untuk menghapus snapshot cluster DB

`delete-db-cluster-snapshot` Contoh berikut menghapus snapshot cluster DB yang ditentukan.



```
aws rds delete-db-cluster-snapshot \  
--db-cluster-snapshot-identifier mydbclustersnapshot
```

### Output:

```
{  
  "DBClusterSnapshot": {  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1b",  
      "us-east-1e"  
    ],  
    "DBClusterSnapshotIdentifier": "mydbclustersnapshot",  
    "DBClusterIdentifier": "mydbcluster",  
    "SnapshotCreateTime": "2019-06-18T21:21:00.469Z",  
    "Engine": "aurora-mysql",  
    "AllocatedStorage": 0,  
    "Status": "available",  
    "Port": 0,  
    "VpcId": "vpc-6594f31c",  
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",  
    "MasterUsername": "myadmin",  
    "EngineVersion": "5.7.mysql_aurora.2.04.2",  
    "LicenseModel": "aurora-mysql",  
    "SnapshotType": "manual",  
    "PercentProgress": 100,  
    "StorageEncrypted": true,  
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",  
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-  
snapshot:mydbclustersnapshot",  
    "IAMDatabaseAuthenticationEnabled": false  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Snapshot di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [DeleteDbClusterSnapshot](#) di Referensi AWS CLI Perintah.

## delete-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-db-cluster`.

## AWS CLI

Contoh 1: Untuk menghapus instans DB di cluster DB

`delete-db-instance` Contoh berikut menghapus instans DB akhir dalam cluster DB. Anda tidak dapat menghapus kluster DB jika berisi instance DB yang tidak dalam status menghapus. Anda tidak dapat mengambil snapshot akhir saat menghapus instans DB di cluster DB.

```
aws rds delete-db-instance \  
  --db-instance-identifier database-3
```

Output:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "database-3",  
    "DBInstanceClass": "db.r4.large",  
    "Engine": "aurora-postgresql",  
    "DBInstanceStatus": "deleting",  
  
    ...output omitted...  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Instans DB di Cluster Aurora DB](#) di Panduan Pengguna Amazon Aurora.

Contoh 2: Untuk menghapus cluster DB

`delete-db-cluster` Contoh berikut menghapus cluster DB bernama `mycluster` dan mengambil snapshot akhir bernama `mycluster-final-snapshot`. Status cluster DB tersedia saat snapshot sedang diambil. Untuk mengikuti kemajuan penghapusan, gunakan perintah `describe-db-clusters`.

```
aws rds delete-db-cluster \  
  --db-cluster-identifier mycluster \  
  --no-skip-final-snapshot \  
  --final-db-snapshot-identifier mycluster-final-snapshot
```

Output:

```
{
  "DBCluster": {
    "AllocatedStorage": 20,
    "AvailabilityZones": [
      "eu-central-1b",
      "eu-central-1c",
      "eu-central-1a"
    ],
    "BackupRetentionPeriod": 7,
    "DBClusterIdentifier": "mycluster",
    "DBClusterParameterGroup": "default.aurora-postgresql10",
    "DBSubnetGroup": "default-vpc-aa11bb22",
    "Status": "available",

    ...output omitted...

  }
}
```

Untuk informasi selengkapnya, lihat [Cluster Aurora dengan Instans DB Tunggal di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [DeleteDbCluster](#) di Referensi AWS CLI Perintah.

## **delete-db-instance-automated-backup**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-instance-automated-backup`.

### AWS CLI

Untuk menghapus cadangan otomatis yang direplikasi dari Wilayah

`delete-db-instance-automated-backup` Contoh berikut menghapus cadangan otomatis dengan Nama Sumber Daya Amazon (ARN) yang ditentukan.

```
aws rds delete-db-instance-automated-backup \
  --db-instance-automated-backups-arn "arn:aws:rds:us-west-2:123456789012:auto-
  backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
```

Output:

```
{
  "DBInstanceAutomatedBackup": {
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
    "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
    "Region": "us-east-1",
    "DBInstanceIdentifier": "new-orcl-db",
    "RestoreWindow": {},
    "AllocatedStorage": 20,
    "Status": "deleting",
    "Port": 1521,
    "AvailabilityZone": "us-east-1b",
    "VpcId": "vpc-#####",
    "InstanceCreateTime": "2020-12-04T15:28:31Z",
    "MasterUsername": "admin",
    "Engine": "oracle-se2",
    "EngineVersion": "12.1.0.2.v21",
    "LicenseModel": "bring-your-own-license",
    "OptionGroupName": "default:oracle-se2-12-1",
    "Encrypted": false,
    "StorageType": "gp2",
    "IAMDatabaseAuthenticationEnabled": false,
    "BackupRetentionPeriod": 7,
    "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-backup:ab-jkib2gfgq5rv7replzadtausbrktni2bn4example"
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus cadangan yang direplikasi di Panduan Pengguna Amazon RDS](#).

- Untuk detail API, lihat [DeleteDbInstanceAutomatedBackup](#) di Referensi AWS CLI Perintah.

## delete-db-instance

Contoh kode berikut menunjukkan cara menggunakan `delete-db-instance`.

### AWS CLI

Untuk menghapus instans DB

`delete-db-instance` Contoh berikut menghapus instance DB tertentu setelah membuat snapshot DB akhir bernama `test-instance-final-snap`

```
aws rds delete-db-instance \  
  --db-instance-identifier test-instance \  
  --final-db-snapshot-identifier test-instance-final-snap
```

Output:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "test-instance",  
    "DBInstanceStatus": "deleting",  
    ...some output truncated...  
  }  
}
```

- Untuk detail API, lihat [DeletedDBInstance](#) di Referensi Perintah AWS CLI .

## delete-db-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `delete-db-parameter-group`.

AWS CLI

Untuk menghapus grup parameter DB

commandContoh berikut menghapus grup parameter DB.

```
aws rds delete-db-parameter-group \  
  --db-parameter-group-name mydbparametergroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DeleteDB ParameterGroup](#) di Referensi AWS CLI Perintah.

## delete-db-proxy-endpoint

Contoh kode berikut menunjukkan cara menggunakan `delete-db-proxy-endpoint`.

## AWS CLI

Untuk menghapus titik akhir proxy DB untuk database RDS

`delete-db-proxy-endpoint` Contoh berikut menghapus titik akhir proxy DB untuk database target.

```
aws rds delete-db-proxy-endpoint \  
  --db-proxy-endpoint-name proxyEP1
```

Output:

```
{  
  "DBProxyEndpoint":  
    {  
      "DBProxyEndpointName": "proxyEP1",  
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-  
endpoint:prx-endpoint-0123a01b12345c0ab",  
      "DBProxyName": "proxyExample",  
      "Status": "deleting",  
      "VpcId": "vpc-1234567",  
      "VpcSecurityGroupIds": [  
        "sg-1234",  
        "sg-5678"  
      ],  
      "VpcSubnetIds": [  
        "subnetgroup1",  
        "subnetgroup2"  
      ],  
      "Endpoint": "proxyEP1.endpoint.proxy-ab0cd1efghij.us-  
east-1.rds.amazonaws.com",  
      "CreateDate": "2023-04-13T01:49:38.568000+00:00",  
      "TargetRole": "READ_ONLY",  
      "IsDefault": false  
    }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus titik akhir proxy](#) di Panduan Pengguna Amazon RDS dan [Menghapus titik akhir proxy di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteDbProxyEndpoint](#) di Referensi AWS CLI Perintah.

## delete-db-proxy

Contoh kode berikut menunjukkan cara menggunakan `delete-db-proxy`.

### AWS CLI

Untuk menghapus proxy DB untuk database RDS

`delete-db-proxy` Contoh berikut menghapus proxy DB.

```
aws rds delete-db-proxy \  
  --db-proxy-name proxyExample
```

Output:

```
{  
  "DBProxy":  
  {  
    "DBProxyName": "proxyExample",  
    "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-  
proxy:prx-0123a01b12345c0ab",  
    "Status": "deleting",  
    "EngineFamily": "PostgreSQL",  
    "VpcId": "vpc-1234567",  
    "VpcSecurityGroupIds": [  
      "sg-1234",  
      "sg-5678"  
    ],  
    "VpcSubnetIds": [  
      "subnetgroup1",  
      "subnetgroup2"  
    ],  
    "Auth": "[  
      {  
        "Description": "proxydescription`"  
        "AuthScheme": "SECRETS",  
        "SecretArn": "arn:aws:secretsmanager:us-  
west-2:123456789123:secret:proxysecret1-Abcd1e",  
        "IAMAuth": "DISABLED"  
      } ],  
    "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",  
    "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-  
east-1.rds.amazonaws.com",
```

```
    "RequireTLS": false,  
    "IdleClientTimeout": 1800,  
    "DebuggingLogging": false,  
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",  
    "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Menghapus Proxy RDS di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteDbProxy](#) di Referensi AWS CLI Perintah.

## **delete-db-security-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-security-group`.

### AWS CLI

Untuk menghapus grup keamanan DB

`delete-db-security-group` Contoh berikut menghapus grup keamanan DB bernama `mysecuritygroup`.

```
aws rds delete-db-security-group \  
  --db-security-group-name mysecuritygroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan grup keamanan DB \(platform EC2-Classic\)](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DeleteDbSecurityGroup](#) di Referensi AWS CLI Perintah.

## **delete-db-shard-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-shard-group`.

### AWS CLI

Contoh 1: Untuk menghapus grup pecahan DB tidak berhasil



`delete-db-shard-group` Contoh berikut menunjukkan kesalahan yang terjadi ketika Anda mencoba untuk menghapus grup shard DB sebelum menghapus semua database dan skema Anda.

```
aws rds delete-db-shard-group \  
  --db-shard-group-identifier limitless-test-shard-grp
```

Output:

```
An error occurred (InvalidDBShardGroupState) when calling the DeleteDBShardGroup  
operation: Unable to delete the DB shard group limitless-test-db-shard-group.  
Delete all of your Limitless Database databases and schemas, then try again.
```

Contoh 2: Untuk menghapus grup pecahan DB dengan sukses

`delete-db-shard-group` Contoh berikut menghapus grup pecahan DB setelah Anda menghapus semua database dan skema Anda, termasuk skema `public`

```
aws rds delete-db-shard-group \  
  --db-shard-group-identifier limitless-test-shard-grp
```

Output:

```
{  
  "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",  
  "DBShardGroupIdentifier": "limitless-test-shard-grp",  
  "DBClusterIdentifier": "limitless-test-cluster",  
  "MaxACU": 768.0,  
  "ComputeRedundancy": 0,  
  "Status": "deleting",  
  "PubliclyAccessible": true,  
  "Endpoint": "limitless-test-cluster.limitless-cekyexample.us-  
east-2.rds.amazonaws.com"  
}
```

Untuk informasi selengkapnya, lihat [Menghapus cluster DB Aurora dan instans DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteDbShardGroup](#) di Referensi AWS CLI Perintah.

## delete-db-snapshot

Contoh kode berikut menunjukkan cara menggunakan `delete-db-snapshot`.

### AWS CLI

Untuk menghapus snapshot DB

`delete-db-snapshot` Contoh berikut menghapus snapshot DB yang ditentukan.

```
aws rds delete-db-snapshot \  
  --db-snapshot-identifier mydbsnapshot
```

Output:

```
{  
  "DBSnapshot": {  
    "DBSnapshotIdentifier": "mydbsnapshot",  
    "DBInstanceIdentifier": "database-mysql",  
    "SnapshotCreateTime": "2019-06-18T22:08:40.702Z",  
    "Engine": "mysql",  
    "AllocatedStorage": 100,  
    "Status": "deleted",  
    "Port": 3306,  
    "AvailabilityZone": "us-east-1b",  
    "VpcId": "vpc-6594f31c",  
    "InstanceCreateTime": "2019-04-30T15:45:53.663Z",  
    "MasterUsername": "admin",  
    "EngineVersion": "5.6.40",  
    "LicenseModel": "general-public-license",  
    "SnapshotType": "manual",  
    "Iops": 1000,  
    "OptionGroupName": "default:mysql-5-6",  
    "PercentProgress": 100,  
    "StorageType": "io1",  
    "Encrypted": true,  
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",  
    "DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mydbsnapshot",  
    "IAMDatabaseAuthenticationEnabled": false,  
    "ProcessorFeatures": [],  
    "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Snapshot](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DeleteDbSnapshot](#) di Referensi AWS CLI Perintah.

## **delete-db-subnet-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-db-subnet-group`.

### AWS CLI

Untuk menghapus grup subnet DB

`delete-db-subnet-group` Contoh berikut menghapus grup subnet DB yang disebut `mysubnetgroup`

```
aws rds delete-db-subnet-group --db-subnet-group-name mysubnetgroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Instans DB di VPC](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DeleteDbSubnetGroup](#) di Referensi AWS CLI Perintah.

## **delete-event-subscription**

Contoh kode berikut menunjukkan cara menggunakan `delete-event-subscription`.

### AWS CLI

Untuk menghapus langganan acara

`delete-event-subscription` Contoh berikut menghapus langganan acara yang ditentukan.

```
aws rds delete-event-subscription --subscription-name my-instance-events
```

Output:

```
{
  "EventSubscription": {
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-events",
  }
}
```

```

    "CustomerAwsId": "123456789012",
    "Enabled": false,
    "SourceIdsList": [
      "test-instance"
    ],
    "SourceType": "db-instance",
    "EventCategoriesList": [
      "backup",
      "recovery"
    ],
    "SubscriptionCreationTime": "2018-07-31 23:22:01.893",
    "CustSubscriptionId": "my-instance-events",
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
    "Status": "deleting"
  }
}

```

- Untuk detail API, lihat [DeleteEventSubscription](#) di Referensi AWS CLI Perintah.

## delete-global-cluster

Contoh kode berikut menunjukkan cara menggunakan `delete-global-cluster`.

### AWS CLI

Untuk menghapus cluster DB global

`delete-global-cluster` Contoh berikut menghapus cluster DB global yang kompatibel dengan Aurora MySQL. Output menunjukkan cluster yang Anda hapus, tetapi `describe-global-clusters` perintah berikutnya tidak mencantumkan cluster DB itu.

```

aws rds delete-global-cluster \
  --global-cluster-identifier myglobalcluster

```

Output:

```

{
  "GlobalCluster": {
    "GlobalClusterIdentifier": "myglobalcluster",
    "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-cluster:myglobalcluster",
  }
}

```

```
    "Status": "available",
    "Engine": "aurora-mysql",
    "EngineVersion": "5.7.mysql_aurora.2.07.2",
    "StorageEncrypted": false,
    "DeletionProtection": false,
    "GlobalClusterMembers": []
  }
}
```

Untuk informasi selengkapnya, lihat [Menghapus database global Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DeleteGlobalCluster](#) di Referensi AWS CLI Perintah.

## **delete-option-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-option-group`.

### AWS CLI

Untuk menghapus grup opsi

`delete-option-group` Contoh berikut menghapus grup opsi yang ditentukan.

```
aws rds delete-option-group \  
  --option-group-name myoptiongroup
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Grup Opsi](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DeleteOptionGroup](#) di Referensi AWS CLI Perintah.

## **deregister-db-proxy-targets**

Contoh kode berikut menunjukkan cara menggunakan `deregister-db-proxy-targets`.

### AWS CLI

Untuk membatalkan pendaftaran target proxy DB dari grup target database

`deregister-db-proxy-targets` Contoh berikut menghapus hubungan antara proxy `proxyExample` dan targetnya.

```
aws rds deregister-db-proxy-targets \  
  --db-proxy-name proxyExample \  
  --db-instance-identifiers database-1
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Menghapus Proxy RDS di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [DeregisterDbProxyTargets](#) di Referensi AWS CLI Perintah.

## describe-account-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-account-attributes`.

### AWS CLI

Untuk menggambarkan atribut akun

`describe-account-attributes` Contoh berikut mengambil atribut untuk AWS akun saat ini.

```
aws rds describe-account-attributes
```

Output:

```
{  
  "AccountQuotas": [  
    {  
      "Max": 40,  
      "Used": 4,  
      "AccountQuotaName": "DBInstances"  
    },  
    {  
      "Max": 40,  
      "Used": 0,  
      "AccountQuotaName": "ReservedDBInstances"  
    },  
    {  
      "Max": 100000,  
      "Used": 40,  
      "AccountQuotaName": "AllocatedStorage"  
    },  
    {
```

```
    "Max": 25,  
    "Used": 0,  
    "AccountQuotaName": "DBSecurityGroups"  
  },  
  {  
    "Max": 20,  
    "Used": 0,  
    "AccountQuotaName": "AuthorizationsPerDBSecurityGroup"  
  },  
  {  
    "Max": 50,  
    "Used": 1,  
    "AccountQuotaName": "DBParameterGroups"  
  },  
  {  
    "Max": 100,  
    "Used": 3,  
    "AccountQuotaName": "ManualSnapshots"  
  },  
  {  
    "Max": 20,  
    "Used": 0,  
    "AccountQuotaName": "EventSubscriptions"  
  },  
  {  
    "Max": 50,  
    "Used": 1,  
    "AccountQuotaName": "DBSubnetGroups"  
  },  
  {  
    "Max": 20,  
    "Used": 1,  
    "AccountQuotaName": "OptionGroups"  
  },  
  {  
    "Max": 20,  
    "Used": 6,  
    "AccountQuotaName": "SubnetsPerDBSubnetGroup"  
  },  
  {  
    "Max": 5,  
    "Used": 0,  
    "AccountQuotaName": "ReadReplicasPerMaster"  
  },  
},
```

```

    {
      "Max": 40,
      "Used": 1,
      "AccountQuotaName": "DBClusters"
    },
    {
      "Max": 50,
      "Used": 0,
      "AccountQuotaName": "DBClusterParameterGroups"
    },
    {
      "Max": 5,
      "Used": 0,
      "AccountQuotaName": "DBClusterRoles"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeAccountAttributes](#) di Referensi AWS CLI Perintah.

## describe-blue-green-deployments

Contoh kode berikut menunjukkan cara menggunakan `describe-blue-green-deployments`.

### AWS CLI

Contoh 1: Untuk menggambarkan penerapan biru/hijau dari instance RDS DB setelah pembuatan selesai

`describe-blue-green-deployment` Contoh berikut mengambil rincian penerapan biru/hijau setelah pembuatan selesai.

```

aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-v53303651eexfake

```

### Output:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
      "BlueGreenDeploymentName": "bgd-cli-test-instance",

```



```
"Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
"Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
rkfbpe",
  "SwitchoverDetails": [
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-rkfbpe",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-j382ha",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ejv4ao",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-vlpz3t",
      "Status": "AVAILABLE"
    }
  ],
  "Tasks": [
    {
      "Name": "CREATING_READ_REPLICA_OF_SOURCE",
      "Status": "COMPLETED"
    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "COMPLETED"
    },
    {
      "Name": "CONFIGURE_BACKUPS",
```

```

        "Status": "COMPLETED"
      },
      {
        "Name": "CREATING_TOPOLOGY_OF_SOURCE",
        "Status": "COMPLETED"
      }
    ],
    "Status": "AVAILABLE",
    "CreateTime": "2022-02-25T21:18:51.183000+00:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat penerapan biru/hijau di Panduan Pengguna Amazon RDS](#).

Contoh 2: Untuk menggambarkan penyebaran biru/hijau untuk cluster DB MySQL Aurora

`describe-blue-green-deployment` Contoh berikut mengambil rincian penyebaran biru/hijau.

```

aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake

```

Output:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
      "BlueGreenDeploymentName": "my-blue-green-deployment",
      "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
      "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnuk1",
      "SwitchoverDetails": [
        {
          "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
          "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnuk1",
          "Status": "AVAILABLE"
        }
      ],
    }
  ]
}

```

```
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1-green-gpmaxf",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2-green-j2oajq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3-green-mkxies",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint-green-4sqjrq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint-green-gwzlg",
        "Status": "AVAILABLE"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    }
],
```

```

        {
            "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
            "Status": "COMPLETED"
        },
        {
            "Name": "CREATE_CUSTOM_ENDPOINTS",
            "Status": "COMPLETED"
        }
    ],
    "Status": "AVAILABLE",
    "CreateTime": "2022-02-25T21:12:00.288000+00:00"
}
]
}

```

Untuk informasi selengkapnya, lihat [Melihat penerapan biru/hijau](#) di Panduan Pengguna Amazon Aurora.

Contoh 3: Untuk menggambarkan penerapan biru/hijau untuk cluster MySQL Aurora setelah peralihan

`describe-blue-green-deployment` Contoh berikut mengambil detail tentang penerapan biru/hijau setelah lingkungan hijau dipromosikan menjadi lingkungan produksi.

```

aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake

```

Output:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
      "BlueGreenDeploymentName": "my-blue-green-deployment",
      "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-old1",
      "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
      "SwitchoverDetails": [
        {
          "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-old1",

```

```

        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint",
        "Status": "SWITCHOVER_COMPLETED"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    }
]

```

```

    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "COMPLETED"
    },
    {
      "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
      "Status": "COMPLETED"
    },
    {
      "Name": "CREATE_CUSTOM_ENDPOINTS",
      "Status": "COMPLETED"
    }
  ],
  "Status": "SWITCHOVER_COMPLETED",
  "CreateTime": "2022-02-25T22:38:49.522000+00:00"
}
]
}

```

Untuk informasi selengkapnya, lihat [Melihat penerapan biru/hijau](#) di Panduan Pengguna Amazon Aurora.

Contoh 4: Untuk menggambarkan penerapan biru/hijau gabungan

`describe-blue-green-deployment` Contoh berikut mengambil rincian penyebaran biru/hijau gabungan.

```
aws rds describe-blue-green-deployments
```

Output:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-wi89nwzgfakelccs",
      "BlueGreenDeploymentName": "my-blue-green-deployment",
      "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
      "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
      "SwitchoverDetails": [
        {

```

```
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1-green-gpmaxf",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2-green-j2oajq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3-green-mkxies",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint-green-4sqjrq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint-green-gwwzlg",
        "Status": "AVAILABLE"
    }
],
"Tasks": [
    {
```

```

        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "COMPLETED"
    }
],
"Status": "AVAILABLE",
"CreateTime": "2022-02-25T21:12:00.288000+00:00"
},
{
    "BlueGreenDeploymentIdentifier": "bgd-v5330365fake1eex",
    "BlueGreenDeploymentName": "bgd-cli-test-instance",
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-old1",
    "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
    "SwitchoverDetails": [
        {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-old1",
            "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
            "Status": "SWITCHOVER_COMPLETED"
        },
        {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-old1",
            "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
            "Status": "SWITCHOVER_COMPLETED"
        },
        {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-old1",
            "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",

```



```

        "Status": "SWITCHOVER_COMPLETED"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
        "Status": "SWITCHOVER_COMPLETED"
      }
    ],
    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
      },
      {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
      },
      {
        "Name": "CONFIGURE_BACKUPS",
        "Status": "COMPLETED"
      },
      {
        "Name": "CREATING_TOPOLOGY_OF_SOURCE",
        "Status": "COMPLETED"
      }
    ],
    "Status": "SWITCHOVER_COMPLETED",
    "CreateTime": "2022-02-25T22:33:22.225000+00:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat penerapan biru/hijau](#) di Panduan Pengguna Amazon RDS dan [Melihat penerapan biru/hijau di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeBlueGreenDeployments](#) di Referensi AWS CLI Perintah.

## describe-certificates

Contoh kode berikut menunjukkan cara menggunakan `describe-certificates`.

## AWS CLI

Untuk menggambarkan sertifikat

`describe-certificates` Contoh berikut mengambil rincian sertifikat yang terkait dengan wilayah default pengguna.

```
aws rds describe-certificates
```

Output:

```
{
  "Certificates": [
    {
      "CertificateIdentifier": "rds-ca-ecc384-g1",
      "CertificateType": "CA",
      "Thumbprint": "2ee3dcc06e50192559b13929e73484354f23387d",
      "ValidFrom": "2021-05-24T22:06:59+00:00",
      "ValidTill": "2121-05-24T23:06:59+00:00",
      "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-ecc384-g1",
      "CustomerOverride": false
    },
    {
      "CertificateIdentifier": "rds-ca-rsa4096-g1",
      "CertificateType": "CA",
      "Thumbprint": "19da4f2af579a8ae1f6a0fa77aa5befd874b4cab",
      "ValidFrom": "2021-05-24T22:03:20+00:00",
      "ValidTill": "2121-05-24T23:03:20+00:00",
      "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-rsa4096-g1",
      "CustomerOverride": false
    },
    {
      "CertificateIdentifier": "rds-ca-rsa2048-g1",
      "CertificateType": "CA",
      "Thumbprint": "7c40cb42714b6fdb2b296f9bbd0e8bb364436a76",
      "ValidFrom": "2021-05-24T21:59:00+00:00",
      "ValidTill": "2061-05-24T22:59:00+00:00",
      "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-rsa2048-g1",
      "CustomerOverride": true,
      "CustomerOverrideValidTill": "2061-05-24T22:59:00+00:00"
    },
    {
      "CertificateIdentifier": "rds-ca-2019",
```

```

    "CertificateType": "CA",
    "Thumbprint": "d40ddb29e3750dffa671c3140bbf5f478d1c8096",
    "ValidFrom": "2019-08-22T17:08:50+00:00",
    "ValidTill": "2024-08-22T17:08:50+00:00",
    "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-2019",
    "CustomerOverride": false
  }
],
"DefaultCertificateForNewLaunches": "rds-ca-rsa2048-g1"
}

```

Untuk informasi selengkapnya, lihat [Menggunakan SSL/TLS untuk mengenkripsi koneksi ke instans DB di](#) Panduan Pengguna Amazon RDS dan [Menggunakan SSL/TLS untuk mengenkripsi koneksi ke cluster DB di](#) Panduan Pengguna [Amazon Aurora](#).

- Untuk detail API, lihat [DescribeCertificates](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-backtracks

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-backtracks`.

### AWS CLI

Untuk menggambarkan backtrack untuk cluster DB

`describe-db-cluster-backtracks` Contoh berikut mengambil rincian tentang cluster DB tertentu.

```
aws rds describe-db-cluster-backtracks \
  --db-cluster-identifier mydbcluster
```

Output:

```

{
  "DBClusterBacktracks": [
    {
      "DBClusterIdentifier": "mydbcluster",
      "BacktrackIdentifier": "2f5f5294-0dd2-44c9-9f50-EXAMPLE",
      "BacktrackTo": "2021-02-12T04:59:22Z",
      "BacktrackedFrom": "2021-02-12T14:37:31.640Z",
      "BacktrackRequestCreationTime": "2021-02-12T14:36:18.819Z",
      "Status": "COMPLETED"
    }
  ],
}

```

```
{
  "DBClusterIdentifier": "mydbcluster",
  "BacktrackIdentifier": "3c7a6421-af2a-4ea3-ae95-EXAMPLE",
  "BacktrackTo": "2021-02-11T22:53:46Z",
  "BacktrackedFrom": "2021-02-12T00:09:27.006Z",
  "BacktrackRequestCreationTime": "2021-02-12T00:07:53.487Z",
  "Status": "COMPLETED"
}
```

Untuk informasi selengkapnya, lihat [Melacak kembali klaster Aurora DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbClusterBacktracks](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-endpoints`.

### AWS CLI

Contoh 1: Untuk menggambarkan titik akhir cluster DB

`describe-db-cluster-endpoints` Contoh berikut mengambil detail untuk titik akhir cluster DB Anda. Jenis cluster Aurora yang paling umum memiliki dua titik akhir. Satu titik akhir memiliki tipe `WRITER`. Anda dapat menggunakan endpoint ini untuk semua pernyataan SQL. Endpoint lainnya memiliki tipe `READER`. Anda dapat menggunakan endpoint ini hanya untuk `SELECT` dan pernyataan SQL read-only lainnya.

```
aws rds describe-db-cluster-endpoints
```

Output:

```
{
  "DBClusterEndpoints": [
    {
      "DBClusterIdentifier": "my-database-1",
      "Endpoint": "my-database-1.cluster-cnpxample.us-east-1.rds.amazonaws.com",
      "Status": "creating",
      "EndpointType": "WRITER"
    }
  ]
}
```

```

    },
    {
      "DBClusterIdentifier": "my-database-1",
      "Endpoint": "my-database-1.cluster-ro-cnpxample.us-
east-1.rds.amazonaws.com",
      "Status": "creating",
      "EndpointType": "READER"
    },
    {
      "DBClusterIdentifier": "mydbcluster",
      "Endpoint": "mydbcluster.cluster-cnpxample.us-east-1.rds.amazonaws.com",
      "Status": "available",
      "EndpointType": "WRITER"
    },
    {
      "DBClusterIdentifier": "mydbcluster",
      "Endpoint": "mydbcluster.cluster-ro-cnpxample.us-
east-1.rds.amazonaws.com",
      "Status": "available",
      "EndpointType": "READER"
    }
  ]
}

```

Contoh 2: Untuk menggambarkan titik akhir cluster DB dari satu cluster DB

`describe-db-cluster-endpoints` Contoh berikut mengambil rincian untuk titik akhir cluster DB dari satu cluster DB tertentu. Cluster Aurora Tanpa Server hanya memiliki satu titik akhir dengan tipe. WRITER

```
aws rds describe-db-cluster-endpoints \
  --db-cluster-identifier serverless-cluster
```

Output:

```

{
  "DBClusterEndpoints": [
    {
      "Status": "available",
      "Endpoint": "serverless-cluster.cluster-cnpxample.us-
east-1.rds.amazonaws.com",
      "DBClusterIdentifier": "serverless-cluster",
      "EndpointType": "WRITER"
    }
  ]
}

```

```

    ]
}

```

Untuk informasi selengkapnya, lihat [Manajemen Koneksi Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbClusterEndpoints](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-parameter-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-parameter-groups`.

### AWS CLI

Untuk menggambarkan kelompok parameter cluster DB

`describe-db-cluster-parameter-groups` Contoh berikut mengambil detail untuk grup parameter cluster DB Anda.

```
aws rds describe-db-cluster-parameter-groups
```

Output:

```

{
  "DBClusterParameterGroups": [
    {
      "DBClusterParameterGroupName": "default.aurora-mysql5.7",
      "DBParameterGroupFamily": "aurora-mysql5.7",
      "Description": "Default cluster parameter group for aurora-mysql5.7",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-pg:default.aurora-mysql5.7"
    },
    {
      "DBClusterParameterGroupName": "default.aurora-postgresql9.6",
      "DBParameterGroupFamily": "aurora-postgresql9.6",
      "Description": "Default cluster parameter group for aurora-postgresql9.6",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-pg:default.aurora-postgresql9.6"
    },
    {

```

```

    "DBClusterParameterGroupName": "default.aurora5.6",
    "DBParameterGroupFamily": "aurora5.6",
    "Description": "Default cluster parameter group for aurora5.6",
    "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora5.6"
  },
  {
    "DBClusterParameterGroupName": "mydbclusterpg",
    "DBParameterGroupFamily": "aurora-mysql5.7",
    "Description": "My DB cluster parameter group",
    "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:mydbclusterpg"
  },
  {
    "DBClusterParameterGroupName": "mydbclusterpgcopy",
    "DBParameterGroupFamily": "aurora-mysql5.7",
    "Description": "Copy of mydbclusterpg parameter group",
    "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:mydbclusterpgcopy"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB dan Grup Parameter Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbClusterParameterGroups](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-parameters`.

### AWS CLI

Contoh 1: Untuk menggambarkan parameter dalam kelompok parameter cluster DB

`describe-db-cluster-parameters` Contoh berikut mengambil rincian tentang parameter dalam kelompok parameter cluster DB.

```
aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name mydbclusterpg
```

Output:

```

{
  "Parameters": [
    {
      "ParameterName": "allow-suspicious-udfs",
      "Description": "Controls whether user-defined functions that have only
an xxx symbol for the main function can be loaded",
      "Source": "engine-default",
      "ApplyType": "static",
      "DataType": "boolean",
      "AllowedValues": "0,1",
      "IsModifiable": false,
      "ApplyMethod": "pending-reboot",
      "SupportedEngineModes": [
        "provisioned"
      ]
    },
    {
      "ParameterName": "aurora_lab_mode",
      "ParameterValue": "0",
      "Description": "Enables new features in the Aurora engine.",
      "Source": "engine-default",
      "ApplyType": "static",
      "DataType": "boolean",
      "AllowedValues": "0,1",
      "IsModifiable": true,
      "ApplyMethod": "pending-reboot",
      "SupportedEngineModes": [
        "provisioned"
      ]
    },
    ...some output truncated...
  ]
}

```

Contoh 2: Untuk mencantumkan hanya nama parameter dalam grup parameter cluster DB

`describe-db-cluster-parameters` Contoh berikut mengambil hanya nama-nama parameter dalam kelompok parameter cluster DB.

```

aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name default.aurora-mysql5.7 \
  --query 'Parameters[].{ParameterName:ParameterName}'

```



**Output:**

```
[
  {
    "ParameterName": "allow-suspicious-udfs"
  },
  {
    "ParameterName": "aurora_binlog_read_buffer_size"
  },
  {
    "ParameterName": "aurora_binlog_replication_max_yield_seconds"
  },
  {
    "ParameterName": "aurora_binlog_use_large_read_buffer"
  },
  {
    "ParameterName": "aurora_lab_mode"
  },
  ...some output truncated...
]
```

Contoh 3: Untuk menjelaskan hanya parameter yang dapat dimodifikasi dalam grup parameter cluster DB

`describe-db-cluster-parameters` Contoh berikut mengambil nama hanya parameter yang dapat Anda modifikasi dalam kelompok parameter cluster DB.

```
aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name default.aurora-mysql5.7 \
  --query 'Parameters[].[ParameterName:ParameterName,IsModifiable:IsModifiable] |
  [?IsModifiable == `true`]'
```

**Output:**

```
[
  {
    "ParameterName": "aurora_binlog_read_buffer_size",
    "IsModifiable": true
  },
  {
```

```

    "ParameterName": "aurora_binlog_replication_max_yield_seconds",
    "IsModifiable": true
  },
  {
    "ParameterName": "aurora_binlog_use_large_read_buffer",
    "IsModifiable": true
  },
  {
    "ParameterName": "aurora_lab_mode",
    "IsModifiable": true
  },
  ...some output truncated...
}
]

```

Contoh 4: Untuk menjelaskan hanya parameter Boolean yang dapat dimodifikasi dalam grup parameter cluster DB

`describe-db-cluster-parameters` Contoh berikut mengambil nama hanya parameter yang dapat Anda modifikasi dalam kelompok parameter cluster DB dan yang memiliki tipe data Boolean.

```

aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name default.aurora-mysql5.7 \
  --query 'Parameters[?]
{ParameterName:ParameterName,DataType:DataType,IsModifiable:IsModifiable} | [?
DataType == `boolean`] | [?IsModifiable == `true`]'

```

Output:

```

[
  {
    "DataType": "boolean",
    "ParameterName": "aurora_binlog_use_large_read_buffer",
    "IsModifiable": true
  },
  {
    "DataType": "boolean",
    "ParameterName": "aurora_lab_mode",
    "IsModifiable": true
  },
]

```

```

    {
      "DataType": "boolean",
      "ParameterName": "autocommit",
      "IsModifiable": true
    },
    {
      "DataType": "boolean",
      "ParameterName": "automatic_sp_privileges",
      "IsModifiable": true
    },
    ...some output truncated...
  }
]

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB dan Grup Parameter Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbClusterParameters](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-snapshot-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-snapshot-attributes`.

### AWS CLI

Untuk mendeskripsikan nama atribut dan nilai untuk snapshot cluster DB

`describe-db-cluster-snapshot-attributes` Contoh berikut mengambil rincian nama atribut dan nilai untuk snapshot cluster DB yang ditentukan.

```
aws rds describe-db-cluster-snapshot-attributes \
  --db-cluster-snapshot-identifier myclustersnapshot
```

Output:

```

{
  "DBClusterSnapshotAttributesResult": {
    "DBClusterSnapshotIdentifier": "myclustersnapshot",
    "DBClusterSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [

```

```
    "123456789012"  
  ]  
} ]  
}
```

Untuk informasi selengkapnya, lihat [Berbagi Snapshot Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbClusterSnapshotAttributes](#) di Referensi AWS CLI Perintah.

## describe-db-cluster-snapshots

Contoh kode berikut menunjukkan cara menggunakan `describe-db-cluster-snapshots`.

### AWS CLI

Untuk menggambarkan snapshot cluster DB untuk cluster DB

`describe-db-cluster-snapshots` Contoh berikut mengambil rincian untuk snapshot cluster DB untuk cluster DB yang ditentukan.

```
aws rds describe-db-cluster-snapshots \  
  --db-cluster-identifier mydbcluster
```

### Output:

```
{  
  "DBClusterSnapshots": [  
    {  
      "AvailabilityZones": [  
        "us-east-1a",  
        "us-east-1b",  
        "us-east-1e"  
      ],  
      "DBClusterSnapshotIdentifier": "myclustersnapshotcopy",  
      "DBClusterIdentifier": "mydbcluster",  
      "SnapshotCreateTime": "2019-06-04T09:16:42.649Z",  
      "Engine": "aurora-mysql",  
      "AllocatedStorage": 0,  
      "Status": "available",  
      "Port": 0,  
    }  
  ]  
}
```

```

    "VpcId": "vpc-6594f31c",
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
    "MasterUsername": "myadmin",
    "EngineVersion": "5.7.mysql_aurora.2.04.2",
    "LicenseModel": "aurora-mysql",
    "SnapshotType": "manual",
    "PercentProgress": 100,
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
AKIAIOSFODNN7EXAMPLE",
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:814387698303:cluster-
snapshot:myclustersnapshotcopy",
    "IAMDatabaseAuthenticationEnabled": false
  },
  {
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1e"
    ],
    "DBClusterSnapshotIdentifier": "rds:mydbcluster-2019-06-20-09-16",
    "DBClusterIdentifier": "mydbcluster",
    "SnapshotCreateTime": "2019-06-20T09:16:26.569Z",
    "Engine": "aurora-mysql",
    "AllocatedStorage": 0,
    "Status": "available",
    "Port": 0,
    "VpcId": "vpc-6594f31c",
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
    "MasterUsername": "myadmin",
    "EngineVersion": "5.7.mysql_aurora.2.04.2",
    "LicenseModel": "aurora-mysql",
    "SnapshotType": "automated",
    "PercentProgress": 100,
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:814387698303:key/
AKIAIOSFODNN7EXAMPLE",
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:rds:mydbcluster-2019-06-20-09-16",
    "IAMDatabaseAuthenticationEnabled": false
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Snapshot Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbClusterSnapshots](#) di Referensi AWS CLI Perintah.

## describe-db-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-db-clusters`.

### AWS CLI

Contoh 1: Untuk menggambarkan cluster DB

`describe-db-clusters` Contoh berikut mengambil rincian cluster DB tertentu.

```
aws rds describe-db-clusters \
  --db-cluster-identifier mydbcluster
```

Output:

```
{
  "DBClusters": [
    {
      "AllocatedStorage": 1,
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1e"
      ],
      "BackupRetentionPeriod": 1,
      "DatabaseName": "mydbcluster",
      "DBClusterIdentifier": "mydbcluster",
      "DBClusterParameterGroup": "default.aurora-mysql5.7",
      "DBSubnetGroup": "default",
      "Status": "available",
      "EarliestRestorableTime": "2019-06-19T09:16:28.210Z",
      "Endpoint": "mydbcluster.cluster-cnpeexample.us-
east-1.rds.amazonaws.com",
      "ReaderEndpoint": "mydbcluster.cluster-ro-cnpeexample.us-
east-1.rds.amazonaws.com",
      "MultiAZ": true,
      "Engine": "aurora-mysql",
      "EngineVersion": "5.7.mysql_aurora.2.04.2",
```

```
"LatestRestorableTime": "2019-06-20T22:38:14.908Z",
"Port": 3306,
"MasterUsername": "myadmin",
"PreferredBackupWindow": "09:09-09:39",
"PreferredMaintenanceWindow": "sat:04:09-sat:04:39",
"ReadReplicaIdentifiers": [],
"DBClusterMembers": [
  {
    "DBInstanceIdentifier": "dbinstance3",
    "IsClusterWriter": false,
    "DBClusterParameterGroupStatus": "in-sync",
    "PromotionTier": 1
  },
  {
    "DBInstanceIdentifier": "dbinstance1",
    "IsClusterWriter": false,
    "DBClusterParameterGroupStatus": "in-sync",
    "PromotionTier": 1
  },
  {
    "DBInstanceIdentifier": "dbinstance2",
    "IsClusterWriter": false,
    "DBClusterParameterGroupStatus": "in-sync",
    "PromotionTier": 1
  },
  {
    "DBInstanceIdentifier": "mydbcluster",
    "IsClusterWriter": false,
    "DBClusterParameterGroupStatus": "in-sync",
    "PromotionTier": 1
  },
  {
    "DBInstanceIdentifier": "mydbcluster-us-east-1b",
    "IsClusterWriter": false,
    "DBClusterParameterGroupStatus": "in-sync",
    "PromotionTier": 1
  },
  {
    "DBInstanceIdentifier": "mydbcluster",
    "IsClusterWriter": true,
    "DBClusterParameterGroupStatus": "in-sync",
    "PromotionTier": 1
  }
],
```

```

    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z2R2ITUGPM61AM",
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:814387698303:key/
AKIAIOSFODNN7EXAMPLE",
    "DbClusterResourceId": "cluster-AKIAIOSFODNN7EXAMPLE",
    "DBClusterArn": "arn:aws:rds:us-
east-1:123456789012:cluster:mydbcluster",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false
  }
]
}

```

Contoh 2: Untuk daftar atribut tertentu dari semua cluster DB

`describe-db-clusters` Contoh berikut hanya mengambil `DBClusterIdentifier`, `Endpoint`, dan `ReaderEndpoint` atribut semua cluster DB Anda di Wilayah saat ini AWS .

```

aws rds describe-db-clusters \
  --query 'DBClusters[.]
{DBClusterIdentifier:DBClusterIdentifier,Endpoint:Endpoint,ReaderEndpoint:ReaderEndpoint}'

```

Output:

```

[
  {
    "Endpoint": "cluster-57-2020-05-01-2270.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
    "ReaderEndpoint": "cluster-57-2020-05-01-2270.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
    "DBClusterIdentifier": "cluster-57-2020-05-01-2270"
  },
  {

```



```

    "Endpoint": "cluster-57-2020-05-01-4615.cluster-cnpxexample.us-
east-1.rds.amazonaws.com",
    "ReaderEndpoint": "cluster-57-2020-05-01-4615.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "DBClusterIdentifier": "cluster-57-2020-05-01-4615"
  },
  {
    "Endpoint": "pg2-cluster.cluster-cnpxexample.us-east-1.rds.amazonaws.com",
    "ReaderEndpoint": "pg2-cluster.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "DBClusterIdentifier": "pg2-cluster"
  },
  ...output omitted...
}
]

```

Contoh 3: Untuk daftar cluster DB dengan atribut tertentu

`describe-db-clusters` Contoh berikut hanya mengambil `DBClusterIdentifier` dan `Engine` atribut cluster DB Anda yang menggunakan mesin `aurora-postgresql` DB.

```

aws rds describe-db-clusters \
  --query 'DBClusters[].{DBClusterIdentifier:DBClusterIdentifier,Engine:Engine} |
[?Engine == `aurora-postgresql`]'

```

Output:

```

[
  {
    "Engine": "aurora-postgresql",
    "DBClusterIdentifier": "pg2-cluster"
  }
]

```

Untuk informasi selengkapnya, lihat [Cluster DB Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [DescribeDbClusters](#) di Referensi AWS CLI Perintah.

## `describe-db-engine-versions`

Contoh kode berikut menunjukkan cara menggunakan `describe-db-engine-versions`.

## AWS CLI

Untuk menggambarkan versi mesin DB untuk mesin MySQL DB

`describe-db-engine-versions` Contoh berikut menampilkan rincian tentang masing-masing versi mesin DB untuk mesin DB yang ditentukan.

```
aws rds describe-db-engine-versions \  
--engine mysql
```

Output:

```
{  
  "DBEngineVersions": [  
    {  
      "Engine": "mysql",  
      "EngineVersion": "5.5.46",  
      "DBParameterGroupFamily": "mysql5.5",  
      "DBEngineDescription": "MySQL Community Edition",  
      "DBEngineVersionDescription": "MySQL 5.5.46",  
      "ValidUpgradeTarget": [  
        {  
          "Engine": "mysql",  
          "EngineVersion": "5.5.53",  
          "Description": "MySQL 5.5.53",  
          "AutoUpgrade": false,  
          "IsMajorVersionUpgrade": false  
        },  
        {  
          "Engine": "mysql",  
          "EngineVersion": "5.5.54",  
          "Description": "MySQL 5.5.54",  
          "AutoUpgrade": false,  
          "IsMajorVersionUpgrade": false  
        },  
        {  
          "Engine": "mysql",  
          "EngineVersion": "5.5.57",  
          "Description": "MySQL 5.5.57",  
          "AutoUpgrade": false,  
          "IsMajorVersionUpgrade": false  
        }  
      ],  
      ...some output truncated...  
    }  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Apa itu Amazon Relational Database Service \(Amazon RDS\)?](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeDB EngineVersions](#) di Referensi AWS CLI Perintah.

## describe-db-instance-automated-backups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-instance-automated-backups`.

### AWS CLI

Untuk menggambarkan backup otomatis untuk instans DB

`describe-db-instance-automated-backups` Contoh berikut menampilkan rincian tentang backup otomatis untuk instans DB tertentu. Detailnya termasuk cadangan otomatis yang direplikasi di Wilayah lain. AWS

```
aws rds describe-db-instance-automated-backups \
  --db-instance-identifier new-orcl-db
```

### Output:

```
{
  "DBInstanceAutomatedBackups": [
    {
      "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
      "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
      "Region": "us-east-1",
      "DBInstanceIdentifier": "new-orcl-db",
      "RestoreWindow": {
        "EarliestTime": "2020-12-07T21:05:20.939Z",
        "LatestTime": "2020-12-07T21:05:20.939Z"
      },
      "AllocatedStorage": 20,
      "Status": "replicating",
      "Port": 1521,
      "InstanceCreateTime": "2020-12-04T15:28:31Z",
```

```

    "MasterUsername": "admin",
    "Engine": "oracle-se2",
    "EngineVersion": "12.1.0.2.v21",
    "LicenseModel": "bring-your-own-license",
    "OptionGroupName": "default:oracle-se2-12-1",
    "Encrypted": false,
    "StorageType": "gp2",
    "IAMDatabaseAuthenticationEnabled": false,
    "BackupRetentionPeriod": 14,
    "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-
west-2:123456789012:auto-backup:ab-jkib2gfg5rv7replzadausbrktni2bn4example"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menemukan informasi tentang cadangan yang direplikasi di Panduan Pengguna Amazon RDS](#).

- Untuk detail API, lihat [DescribeDbInstanceAutomatedBackups](#) di Referensi AWS CLI Perintah.

## describe-db-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-db-instances`.

### AWS CLI

Untuk menggambarkan instance DB

`describe-db-instances` Contoh berikut mengambil rincian tentang instans DB tertentu.

```

aws rds describe-db-instances \
  --db-instance-identifier mydbinstancecf

```

Output:

```

{
  "DBInstances": [
    {
      "DBInstanceIdentifier": "mydbinstancecf",
      "DBInstanceClass": "db.t3.small",
      "Engine": "mysql",
      "DBInstanceStatus": "available",

```

```

        "MasterUsername": "masterawsuser",
        "Endpoint": {
            "Address": "mydbinstancecf.abcxample.us-east-1.rds.amazonaws.com",
            "Port": 3306,
            "HostedZoneId": "Z2R2ITUGPM61AM"
        },
        ...some output truncated...
    }
]
}

```

- Untuk detail API, lihat [DescribedInstances](#) di Referensi Perintah.AWS CLI

## describe-db-log-files

Contoh kode berikut menunjukkan cara menggunakan `describe-db-log-files`.

### AWS CLI

Untuk mendeskripsikan file log untuk instance DB

`describe-db-log-files` Contoh berikut mengambil rincian tentang file log untuk instance DB tertentu.

```
aws rds describe-db-log-files -\
    -db-instance-identifier test-instance
```

Output:

```

{
  "DescribeDBLogFiles": [
    {
      "Size": 0,
      "LastWritten": 1533060000000,
      "LogFileName": "error/mysql-error-running.log"
    },
    {
      "Size": 2683,
      "LastWritten": 1532994300000,
      "LogFileName": "error/mysql-error-running.log.0"
    },
    {

```

```

        "Size": 107,
        "LastWritten": 1533057300000,
        "LogFileName": "error/mysql-error-running.log.18"
    },
    {
        "Size": 13105,
        "LastWritten": 1532991000000,
        "LogFileName": "error/mysql-error-running.log.23"
    },
    {
        "Size": 0,
        "LastWritten": 1533061200000,
        "LogFileName": "error/mysql-error.log"
    },
    {
        "Size": 3519,
        "LastWritten": 1532989252000,
        "LogFileName": "mysqlUpgrade"
    }
]
}

```

- Untuk detail API, lihat [DescribeDbLogFiles](#) di Referensi AWS CLI Perintah.

## describe-db-parameter-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-parameter-groups`.

### AWS CLI

Untuk menggambarkan grup parameter DB Anda

`describe-db-parameter-groups` Contoh berikut mengambil rincian tentang grup parameter DB Anda.

```
aws rds describe-db-parameter-groups
```

Output:

```

{
  "DBParameterGroups": [
    {

```

```

        "DBParameterGroupName": "default.aurora-mysql5.7",
        "DBParameterGroupFamily": "aurora-mysql5.7",
        "Description": "Default parameter group for aurora-mysql5.7",
        "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora-mysql5.7"
    },
    {
        "DBParameterGroupName": "default.aurora-postgresql9.6",
        "DBParameterGroupFamily": "aurora-postgresql9.6",
        "Description": "Default parameter group for aurora-postgresql9.6",
        "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora-postgresql9.6"
    },
    {
        "DBParameterGroupName": "default.aurora5.6",
        "DBParameterGroupFamily": "aurora5.6",
        "Description": "Default parameter group for aurora5.6",
        "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora5.6"
    },
    {
        "DBParameterGroupName": "default.mariadb10.1",
        "DBParameterGroupFamily": "mariadb10.1",
        "Description": "Default parameter group for mariadb10.1",
        "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.mariadb10.1"
    },
    ...some output truncated...
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeDB ParameterGroups](#) di Referensi AWS CLI Perintah.

## describe-db-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-db-parameters`.

### AWS CLI

Untuk menggambarkan parameter dalam kelompok parameter DB

`describe-db-parameters` Contoh berikut mengambil rincian kelompok parameter DB yang ditentukan.

```
aws rds describe-db-parameters \  
  --db-parameter-group-name mydbpg
```

Output:

```
{  
  "Parameters": [  
    {  
      "ParameterName": "allow-suspicious-udfs",  
      "Description": "Controls whether user-defined functions that have only  
an xxx symbol for the main function can be loaded",  
      "Source": "engine-default",  
      "ApplyType": "static",  
      "DataType": "boolean",  
      "AllowedValues": "0,1",  
      "IsModifiable": false,  
      "ApplyMethod": "pending-reboot"  
    },  
    {  
      "ParameterName": "auto_generate_certs",  
      "Description": "Controls whether the server autogenerates SSL key and  
certificate files in the data directory, if they do not already exist.",  
      "Source": "engine-default",  
      "ApplyType": "static",  
      "DataType": "boolean",  
      "AllowedValues": "0,1",  
      "IsModifiable": false,  
      "ApplyMethod": "pending-reboot"  
    },  
    ...some output truncated...  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [AWS CLI DescribedBParameters](#) di Referensi Perintah.



## describe-db-proxies

Contoh kode berikut menunjukkan cara menggunakan `describe-db-proxies`.

### AWS CLI

Untuk menggambarkan proxy DB untuk database RDS

`describe-db-proxies` Contoh berikut mengembalikan informasi tentang proxy DB.

```
aws rds describe-db-proxies
```

Output:

```
{
  "DBProxies": [
    {
      "DBProxyName": "proxyExample1",
      "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-proxy:prx-0123a01b12345c0ab",
      "Status": "available",
      "EngineFamily": "PostgreSQL",
      "VpcId": "vpc-1234567",
      "VpcSecurityGroupIds": [
        "sg-1234"
      ],
      "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
      ],
      "Auth": "[
        {
          "Description": "proxydescription1",
          "AuthScheme": "SECRETS",
          "SecretArn": "arn:aws:secretsmanager:us-west-2:123456789123:secret:secretName-1234f",
          "IAMAuth": "DISABLED"
        }
      ]",
      "RoleArn": "arn:aws:iam::12345678912??:role/ProxyPostgreSQLRole",
      "Endpoint": "proxyExample1.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
      "RequireTLS": false,
    }
  ]
}
```

```

    "IdleClientTimeout": 1800,
    "DebuggingLogging": false,
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",
    "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
  },
  {
    "DBProxyName": "proxyExample2",
    "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-1234a12b23456c1ab",
    "Status": "available",
    "EngineFamily": "PostgreSQL",
    "VpcId": "sg-1234567",
    "VpcSecurityGroupIds": [
      "sg-1234"
    ],
    "VpcSubnetIds": [
      "subnetgroup1",
      "subnetgroup2"
    ],
    "Auth": "[
      {
        "Description": "proxydescription2"
        "AuthScheme": "SECRETS",
        "SecretArn": "aarn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",
        "IAMAuth": "DISABLED"
      }
    ]",
    "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
    "Endpoint": "proxyExample2.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
    "RequireTLS": false,
    "IdleClientTimeout": 1800,
    "DebuggingLogging": false,
    "CreateDate": "2022-01-05T16:19:33.452000+00:00",
    "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Melihat Proxy RDS](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbProxies](#) di Referensi AWS CLI Perintah.

## describe-db-proxy-endpoints

Contoh kode berikut menunjukkan cara menggunakan `describe-db-proxy-endpoints`.

### AWS CLI

Untuk menggambarkan titik akhir proxy DB

`describe-db-proxy-endpoints` Contoh berikut mengembalikan informasi tentang titik akhir proxy DB.

```
aws rds describe-db-proxy-endpoints
```

Output:

```
{
  "DBProxyEndpoints": [
    {
      "DBProxyEndpointName": "proxyEndpoint1",
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
      "DBProxyName": "proxyExample",
      "Status": "available",
      "VpcId": "vpc-1234567",
      "VpcSecurityGroupIds": [
        "sg-1234"
      ],
      "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
      ],
      "Endpoint": "proxyEndpoint1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
      "CreateDate": "2023-04-05T16:09:33.452000+00:00",
      "TargetRole": "READ_WRITE",
      "IsDefault": false
    },
    {
      "DBProxyEndpointName": "proxyEndpoint2",
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-4567a01b12345c0ab",
      "DBProxyName": "proxyExample2",
      "Status": "available",
```

```

        "VpcId": "vpc1234567",
        "VpcSecurityGroupIds": [
            "sg-5678"
        ],
        "VpcSubnetIds": [
            "subnetgroup1",
            "subnetgroup2"
        ],
        "Endpoint": "proxyEndpoint2.endpoint.proxy-cd1ef2klmnop.us-
east-1.rds.amazonaws.com",
        "CreateDate": "2023-04-05T16:09:33.452000+00:00",
        "TargetRole": "READ_WRITE",
        "IsDefault": false
    }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat titik akhir proxy](#) di Panduan Pengguna Amazon RDS dan [Membuat titik akhir proxy](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbProxyEndpoints](#) di Referensi AWS CLI Perintah.

## describe-db-proxy-target-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-proxy-target-groups`.

### AWS CLI

Untuk menggambarkan titik akhir proxy DB

`describe-db-proxy-target-groups` Contoh berikut mengembalikan informasi tentang kelompok target proxy DB.

```
aws rds describe-db-proxy-target-groups \
  --db-proxy-name proxyExample
```

Output:

```
{
  "TargetGroups": [
    {
      "DBProxyName": "proxyExample",
      "TargetGroupName": "default",

```

```

    "TargetGroupArn": "arn:aws:rds:us-east-1:123456789012:target-group:prx-
tg-0123a01b12345c0ab",
    "IsDefault": true,
    "Status": "available",
    "ConnectionPoolConfig": {
      "MaxConnectionsPercent": 100,
      "MaxIdleConnectionsPercent": 50,
      "ConnectionBorrowTimeout": 120,
      "SessionPinningFilters": []
    },
    "CreateDate": "2023-05-02T18:41:19.495000+00:00",
    "UpdateDate": "2023-05-02T18:41:21.762000+00:00"
  }
}

```

Untuk informasi selengkapnya, lihat [Melihat Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Melihat Proxy RDS](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbProxyTargetGroups](#) di Referensi AWS CLI Perintah.

## describe-db-proxy-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-db-proxy-targets`.

### AWS CLI

Untuk menggambarkan target proxy DB

`describe-db-proxy-targets` Contoh berikut mengembalikan informasi tentang target proxy DB.

```
aws rds describe-db-proxy-targets \
  --db-proxy-name proxyExample
```

Output:

```

{
  "Targets": [
    {
      "Endpoint": "database1.ab0cd1efghij.us-east-1.rds.amazonaws.com",
      "TrackedClusterId": "database1",
      "RdsResourceId": "database1-instance-1",
      "Port": 3306,
    }
  ]
}

```

```
    "Type": "RDS_INSTANCE",
    "Role": "READ_WRITE",
    "TargetHealth": {
      "State": "UNAVAILABLE",
      "Reason": "PENDING_PROXY_CAPACITY",
      "Description": "DBProxy Target is waiting for proxy to scale to
desired capacity"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Melihat proxy RDS](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeDbProxyTargets](#) di Referensi AWS CLI Perintah.

## describe-db-recommendations

Contoh kode berikut menunjukkan cara menggunakan `describe-db-recommendations`.

### AWS CLI

Contoh 1: Untuk daftar semua rekomendasi DB

`describe-db-recommendations` Contoh berikut mencantumkan semua rekomendasi DB di AWS akun Anda.

```
aws rds describe-db-recommendations
```

Output:

```
{
  "DBRecommendations": [
    {
      "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
      "TypeId": "config_recommendation::old_minor_version",
      "Severity": "informational",
      "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
      "Status": "active",
      "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
      "UpdatedTime": "2024-02-21T23:14:19+00:00",
    }
  ]
}
```

```
    "Detection": "***[resource-name]** is not running the latest minor DB
engine version",
    "Recommendation": "Upgrade to latest engine version",
    "Description": "Your database resources aren't running the latest minor
DB engine version. The latest minor version contains the latest security fixes and
other improvements.",
    "RecommendedActions": [
      {
        "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
        "Operation": "modifyDbInstance",
        "Parameters": [
          {
            "Key": "EngineVersion",
            "Value": "5.7.44"
          },
          {
            "Key": "DBInstanceIdentifier",
            "Value": "database-1"
          }
        ],
        "ApplyModes": [
          "immediately",
          "next-maintenance-window"
        ],
        "Status": "ready",
        "ContextAttributes": [
          {
            "Key": "Recommended value",
            "Value": "5.7.44"
          },
          {
            "Key": "Current engine version",
            "Value": "5.7.42"
          }
        ]
      }
    ],
    "Category": "security",
    "Source": "RDS",
    "TypeDetection": "***[resource-count] resources** are not running the
latest minor DB engine version",
    "TypeRecommendation": "Upgrade to latest engine version",
    "Impact": "Reduced database performance and data security at risk",
```

```

    "AdditionalInfo": "We recommend that you maintain your database with the
latest DB engine minor version as this version includes the latest security and
functionality fixes. The DB engine minor version upgrades contain only the changes
which are backward-compatible with earlier minor versions of the same major version
of the DB engine.",
    "Links": [
      {
        "Text": "Upgrading an RDS DB instance engine version",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
      },
      {
        "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon Aurora",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
      },
      {
        "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon RDS",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Melihat dan menanggapi rekomendasi Amazon RDS](#) di Panduan Pengguna Amazon RDS [dan Melihat serta menanggapi rekomendasi Amazon RDS di Panduan Pengguna Amazon Aurora](#).

Contoh 2: Untuk daftar rekomendasi DB tingkat keparahan tinggi

`describe-db-recommendations` Contoh berikut mencantumkan rekomendasi DB tingkat keparahan tinggi di AWS akun Anda.

```
aws rds describe-db-recommendations \
  --filters Name=severity,Values=high
```

Output:

```
{
```



```
"DBRecommendations": [
  {
    "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
    "TypeId": "config_recommendation::rds_extended_support",
    "Severity": "high",
    "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
    "Status": "active",
    "CreatedTime": "2024-02-21T23:14:19.392000+00:00",
    "UpdatedTime": "2024-02-21T23:14:19+00:00",
    "Detection": "Your databases will be auto-enrolled to RDS Extended
Support on February 29",
    "Recommendation": "Upgrade your major version before February 29, 2024
to avoid additional charges",
    "Description": "Your PostgreSQL 11 and MySQL 5.7 databases will be
automatically enrolled into RDS Extended Support on February 29, 2024. To avoid
the increase in charges due to RDS Extended Support, we recommend upgrading your
databases to a newer major engine version before February 29, 2024.\n\nTo learn more
about the RDS Extended Support pricing, refer to the pricing page.",
    "RecommendedActions": [
      {
        "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
        "Parameters": [],
        "ApplyModes": [
          "manual"
        ],
        "Status": "ready",
        "ContextAttributes": []
      }
    ],
    "Category": "cost optimization",
    "Source": "RDS",
    "TypeDetection": "Your database will be auto-enrolled to RDS Extended
Support on February 29",
    "TypeRecommendation": "Upgrade your major version before February 29,
2024 to avoid additional charges",
    "Impact": "Increase in charges due to RDS Extended Support",
    "AdditionalInfo": "With Amazon RDS Extended Support, you can continue
running your database on a major engine version past the RDS end of standard
support date for an additional cost. This paid feature gives you more time to
upgrade to a supported major engine version.\n\nDuring Extended Support, Amazon RDS
will supply critical CVE patches and bug fixes.",
    "Links": [
      {
        "Text": "Amazon RDS Extended Support pricing for RDS for MySQL",
```

```

        "Url": "https://aws.amazon.com/rds/mysql/pricing/"
    },
    {
        "Text": "Amazon RDS Extended Support for RDS for MySQL and
PostgreSQL databases",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
extended-support.html"
    },
    {
        "Text": "Amazon RDS Extended Support pricing for Amazon Aurora
PostgreSQL",
        "Url": "https://aws.amazon.com/rds/aurora/pricing/"
    },
    {
        "Text": "Amazon RDS Extended Support for Aurora PostgreSQL
databases",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/extended-support.html"
    },
    {
        "Text": "Amazon RDS Extended Support pricing for RDS for
PostgreSQL",
        "Url": "https://aws.amazon.com/rds/postgresql/pricing/"
    }
]
}
]
}

```

Untuk informasi selengkapnya, lihat [Melihat dan menanggapi rekomendasi Amazon RDS](#) di Panduan Pengguna Amazon RDS [dan Melihat serta menanggapi rekomendasi Amazon RDS di Panduan Pengguna Amazon Aurora](#).

Contoh 3: Untuk daftar rekomendasi DB untuk instans DB tertentu

`describe-db-recommendations` Contoh berikut mencantumkan semua rekomendasi DB untuk instance DB tertentu.

```
aws rds describe-db-recommendations \
  --filters Name=dbi-resource-id,Values=database-1
```

Output:

```
{
  "DBRecommendations": [
    {
      "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
      "TypeId": "config_recommendation::old_minor_version",
      "Severity": "informational",
      "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
      "Status": "active",
      "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
      "UpdatedTime": "2024-02-21T23:14:19+00:00",
      "Detection": "***[resource-name]** is not running the latest minor DB
engine version",
      "Recommendation": "Upgrade to latest engine version",
      "Description": "Your database resources aren't running the latest minor
DB engine version. The latest minor version contains the latest security fixes and
other improvements.",
      "RecommendedActions": [
        {
          "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
          "Operation": "modifyDbInstance",
          "Parameters": [
            {
              "Key": "EngineVersion",
              "Value": "5.7.44"
            },
            {
              "Key": "DBInstanceIdentifier",
              "Value": "database-1"
            }
          ],
          "ApplyModes": [
            "immediately",
            "next-maintenance-window"
          ],
          "Status": "ready",
          "ContextAttributes": [
            {
              "Key": "Recommended value",
              "Value": "5.7.44"
            },
            {
              "Key": "Current engine version",
              "Value": "5.7.42"
            }
          ]
        }
      ]
    }
  ]
}
```

```

    }
  ]
}
],
"Category": "security",
"Source": "RDS",
"TypeDetection": "**[resource-count] resources** are not running the
latest minor DB engine version",
"TypeRecommendation": "Upgrade to latest engine version",
"Impact": "Reduced database performance and data security at risk",
"AdditionalInfo": "We recommend that you maintain your database with the
latest DB engine minor version as this version includes the latest security and
functionality fixes. The DB engine minor version upgrades contain only the changes
which are backward-compatible with earlier minor versions of the same major version
of the DB engine.",
"Links": [
  {
    "Text": "Upgrading an RDS DB instance engine version",
    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
  },
  {
    "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon Aurora",
    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
  },
  {
    "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon RDS",
    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
  }
]
}
]
}

```

Untuk informasi selengkapnya, lihat [Melihat dan menanggapi rekomendasi Amazon RDS](#) di Panduan Pengguna Amazon RDS [dan Melihat serta menanggapi rekomendasi Amazon RDS di Panduan Pengguna Amazon Aurora](#).

Contoh 4: Untuk daftar semua rekomendasi DB aktif

`describe-db-recommendations` Contoh berikut mencantumkan semua rekomendasi DB aktif di AWS akun Anda.

```
aws rds describe-db-recommendations \  
  --filters Name=status,Values=active
```

Output:

```
{  
  "DBRecommendations": [  
    {  
      "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",  
      "TypeId": "config_recommendation::old_minor_version",  
      "Severity": "informational",  
      "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",  
      "Status": "active",  
      "CreatedTime": "2024-02-21T23:14:19.292000+00:00",  
      "UpdatedTime": "2024-02-21T23:14:19+00:00",  
      "Detection": "***[resource-name]** is not running the latest minor DB  
engine version",  
      "Recommendation": "Upgrade to latest engine version",  
      "Description": "Your database resources aren't running the latest minor  
DB engine version. The latest minor version contains the latest security fixes and  
other improvements.",  
      "RecommendedActions": [  
        {  
          "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",  
          "Operation": "modifyDbInstance",  
          "Parameters": [  
            {  
              "Key": "EngineVersion",  
              "Value": "5.7.44"  
            },  
            {  
              "Key": "DBInstanceIdentifier",  
              "Value": "database-1"  
            }  
          ],  
          "ApplyModes": [  
            "immediately",  
            "next-maintenance-window"  
          ],  
          "Status": "ready",
```

```
        "ContextAttributes": [
            {
                "Key": "Recommended value",
                "Value": "5.7.44"
            },
            {
                "Key": "Current engine version",
                "Value": "5.7.42"
            }
        ]
    },
    "Category": "security",
    "Source": "RDS",
    "TypeDetection": "***[resource-count] resources** are not running the
latest minor DB engine version",
    "TypeRecommendation": "Upgrade to latest engine version",
    "Impact": "Reduced database performance and data security at risk",
    "AdditionalInfo": "We recommend that you maintain your database with the
latest DB engine minor version as this version includes the latest security and
functionality fixes. The DB engine minor version upgrades contain only the changes
which are backward-compatible with earlier minor versions of the same major version
of the DB engine.",
    "Links": [
        {
            "Text": "Upgrading an RDS DB instance engine version",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
        },
        {
            "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon Aurora",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
        },
        {
            "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon RDS",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
        }
    ]
}
]
```

```
}
```

Untuk informasi selengkapnya, lihat [Melihat dan menanggapi rekomendasi Amazon RDS](#) di Panduan Pengguna Amazon RDS [dan Melihat serta menanggapi rekomendasi Amazon RDS di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [DescribeDbRecommendations](#) di Referensi AWS CLI Perintah.

## describe-db-security-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-security-groups`.

### AWS CLI

Untuk daftar grup keamanan DB

`describe-db-security-groups` Contoh berikut mencantumkan grup keamanan DB.

```
aws rds describe-db-security-groups
```

Output:

```
{
  "DBSecurityGroups": [
    {
      "OwnerId": "123456789012",
      "DBSecurityGroupName": "default",
      "DBSecurityGroupDescription": "default",
      "EC2SecurityGroups": [],
      "IPRanges": [],
      "DBSecurityGroupArn": "arn:aws:rds:us-west-1:111122223333:secgrp:default"
    },
    {
      "OwnerId": "123456789012",
      "DBSecurityGroupName": "mysecgroup",
      "DBSecurityGroupDescription": "My Test Security Group",
      "VpcId": "vpc-1234567f",
      "EC2SecurityGroups": [],
      "IPRanges": [],
      "DBSecurityGroupArn": "arn:aws:rds:us-west-1:111122223333:secgrp:mysecgroup"
    }
  ]
}
```

```

    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Daftar Grup Keamanan DB yang Tersedia](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeDbSecurityGroups](#) di Referensi AWS CLI Perintah.

## describe-db-shard-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-db-shard-groups`.

### AWS CLI

Contoh 1: Untuk menggambarkan kelompok pecahan DB

`describe-db-shard-groups` Contoh berikut mengambil rincian grup shard DB Anda.

```
aws rds describe-db-shard-groups
```

Output:

```

{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
      "DBShardGroupIdentifier": "limitless-test-shard-grp",
      "DBClusterIdentifier": "limitless-test-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": true,
      "Endpoint": "limitless-test-cluster.limitless-cekyexample.us-east-2.rds.amazonaws.com"
    },
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",

```



```

        "PubliclyAccessible": false,
        "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-
east-2.rds.amazonaws.com"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Cluster DB Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [DescribeDbShardGroups](#) di Referensi AWS CLI Perintah.

## describe-db-snapshot-attributes

Contoh kode berikut menunjukkan cara menggunakan `describe-db-snapshot-attributes`.

### AWS CLI

Untuk mendeskripsikan nama atribut dan nilai untuk snapshot DB

`describe-db-snapshot-attributes` Contoh berikut menjelaskan nama atribut dan nilai untuk snapshot DB.

```

aws rds describe-db-snapshot-attributes \
  --db-snapshot-identifier mydbsnapshot

```

Output:

```

{
  "DBSnapshotAttributesResult": {
    "DBSnapshotIdentifier": "mydbsnapshot",
    "DBSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [
          "123456789012",
          "210987654321"
        ]
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Berbagi Snapshot DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeDbSnapshotAttributes](#) di Referensi AWS CLI Perintah.

## describe-db-snapshots

Contoh kode berikut menunjukkan cara menggunakan `describe-db-snapshots`.

### AWS CLI

Contoh 1: Untuk menggambarkan snapshot DB untuk instance DB

`describe-db-snapshots` Contoh berikut mengambil rincian snapshot DB untuk instance DB.

```
aws rds describe-db-snapshots \  
  --db-snapshot-identifier mydbsnapshot
```

Output:

```
{  
  "DBSnapshots": [  
    {  
      "DBSnapshotIdentifier": "mydbsnapshot",  
      "DBInstanceIdentifier": "mysqldb",  
      "SnapshotCreateTime": "2018-02-08T22:28:08.598Z",  
      "Engine": "mysql",  
      "AllocatedStorage": 20,  
      "Status": "available",  
      "Port": 3306,  
      "AvailabilityZone": "us-east-1f",  
      "VpcId": "vpc-6594f31c",  
      "InstanceCreateTime": "2018-02-08T22:24:55.973Z",  
      "MasterUsername": "mysqladmin",  
      "EngineVersion": "5.6.37",  
      "LicenseModel": "general-public-license",  
      "SnapshotType": "manual",  
      "OptionGroupName": "default:mysql-5-6",  
      "PercentProgress": 100,  
      "StorageType": "gp2",  
      "Encrypted": false,  
      "DBSnapshotArn": "arn:aws:rds:us-  
east-1:123456789012:snapshot:mydbsnapshot",  
      "IAMDatabaseAuthenticationEnabled": false,  
    }  
  ]  
}
```

```

        "ProcessorFeatures": [],
        "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat Snapshot DB](#) di Panduan Pengguna Amazon RDS.

Contoh 2: Untuk menemukan jumlah snapshot manual yang diambil

`describe-db-snapshots` Contoh berikut menggunakan `length` operator dalam `--query` opsi untuk mengembalikan jumlah snapshot manual yang telah diambil di AWS Wilayah tertentu.

```

aws rds describe-db-snapshots \
  --snapshot-type manual \
  --query "length(*[].{DBSnapshots:SnapshotType})" \
  --region eu-central-1

```

Output:

```
35
```

Untuk informasi selengkapnya, lihat [Membuat Snapshot DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribedBSnapshots](#) di Referensi Perintah.AWS CLI

## **describe-db-subnet-groups**

Contoh kode berikut menunjukkan cara menggunakan `describe-db-subnet-groups`.

AWS CLI

Untuk menggambarkan grup subnet DB

`describe-db-subnet-groups` Contoh berikut mengambil rincian kelompok subnet DB tertentu.

```
aws rds describe-db-subnet-groups
```

Output:

```
{
  "DBSubnetGroups": [

```

```
{
  "DBSubnetGroupName": "mydbsubnetgroup",
  "DBSubnetGroupDescription": "My DB Subnet Group",
  "VpcId": "vpc-971c12ee",
  "SubnetGroupStatus": "Complete",
  "Subnets": [
    {
      "SubnetIdentifier": "subnet-d8c8e7f4",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1a"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-718fdc7d",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1f"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-cbc8e7e7",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1a"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-0ccde220",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1a"
      },
      "SubnetStatus": "Active"
    }
  ],
  "DBSubnetGroupArn": "arn:aws:rds:us-east-1:123456789012:subgrp:mydbsubnetgroup"
}
```

Untuk informasi selengkapnya, lihat [VPC Amazon Virtual Private Cloud dan Amazon RDS](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeDbSubnetGroups](#) di Referensi AWS CLI Perintah.

## describe-engine-default-cluster-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-engine-default-cluster-parameters`.

### AWS CLI

Untuk menggambarkan mesin default dan informasi parameter sistem untuk mesin database Aurora

`describe-engine-default-cluster-parameters` Contoh berikut mengambil rincian mesin default dan informasi parameter sistem untuk cluster Aurora DB dengan kompatibilitas MySQL 5.7.

```
aws rds describe-engine-default-cluster-parameters \
  --db-parameter-group-family aurora-mysql5.7
```

Output:

```
{
  "EngineDefaults": {
    "Parameters": [
      {
        "ParameterName": "aurora_load_from_s3_role",
        "Description": "IAM role ARN used to load data from AWS S3",
        "Source": "engine-default",
        "ApplyType": "dynamic",
        "DataType": "string",
        "IsModifiable": true,
        "SupportedEngineModes": [
          "provisioned"
        ]
      },
      ...some output truncated...
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB dan Grup Parameter Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeEngineDefaultClusterParameters](#) di Referensi AWS CLI Perintah.

## describe-engine-default-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-engine-default-parameters`.

### AWS CLI

Untuk menggambarkan mesin default dan informasi parameter sistem untuk mesin database

`describe-engine-default-parameters` Contoh berikut mengambil rincian untuk mesin default dan informasi parameter sistem untuk MySQL 5.7 instans DB.

```
aws rds describe-engine-default-parameters \  
  --db-parameter-group-family mysql5.7
```

### Output:

```
{  
  "EngineDefaults": {  
    "Parameters": [  
      {  
        "ParameterName": "allow-suspicious-udfs",  
        "Description": "Controls whether user-defined functions that have  
only an xxx symbol for the main function can be loaded",  
        "Source": "engine-default",  
        "ApplyType": "static",  
        "DataType": "boolean",  
        "AllowedValues": "0,1",  
        "IsModifiable": false  
      },  
      ...some output truncated...  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeEngineDefaultParameters](#) di Referensi AWS CLI Perintah.

## describe-event-categories

Contoh kode berikut menunjukkan cara menggunakan `describe-event-categories`.

### AWS CLI

Untuk menggambarkan kategori acara

`describe-event-categories` Contoh berikut mengambil rincian tentang kategori acara untuk semua sumber acara yang tersedia.

```
aws rds describe-event-categories
```

Output:

```
{
  "EventCategoriesMapList": [
    {
      "SourceType": "db-instance",
      "EventCategories": [
        "deletion",
        "read replica",
        "failover",
        "restoration",
        "maintenance",
        "low storage",
        "configuration change",
        "backup",
        "creation",
        "availability",
        "recovery",
        "failure",
        "backtrack",
        "notification"
      ]
    },
    {
      "SourceType": "db-security-group",
      "EventCategories": [
        "configuration change",
        "failure"
      ]
    }
  ]
}
```

```

    ]
  },
  {
    "SourceType": "db-parameter-group",
    "EventCategories": [
      "configuration change"
    ]
  },
  {
    "SourceType": "db-snapshot",
    "EventCategories": [
      "deletion",
      "creation",
      "restoration",
      "notification"
    ]
  },
  {
    "SourceType": "db-cluster",
    "EventCategories": [
      "failover",
      "failure",
      "notification"
    ]
  },
  {
    "SourceType": "db-cluster-snapshot",
    "EventCategories": [
      "backup"
    ]
  }
]
}

```

- Untuk detail API, lihat [DescribeEventCategories](#) di Referensi AWS CLI Perintah.

## describe-event-subscriptions

Contoh kode berikut menunjukkan cara menggunakan `describe-event-subscriptions`.

### AWS CLI

Untuk menggambarkan langganan acara



Contoh ini menjelaskan semua langganan acara Amazon RDS untuk akun saat ini AWS .

```
aws rds describe-event-subscriptions
```

Output:

```
{
  "EventSubscriptionsList": [
    {
      "EventCategoriesList": [
        "backup",
        "recovery"
      ],
      "Enabled": true,
      "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-
instance-events",
      "Status": "creating",
      "SourceType": "db-instance",
      "CustomerAwsId": "123456789012",
      "SubscriptionCreationTime": "2018-07-31 23:22:01.893",
      "CustSubscriptionId": "my-instance-events",
      "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events"
    },
    ...some output truncated...
  ]
}
```

- Untuk detail API, lihat [DescribeEventSubscriptions](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Untuk menggambarkan peristiwa

`describe-events` Contoh berikut mengambil rincian untuk peristiwa yang telah terjadi untuk instans DB tertentu.

```
aws rds describe-events \
```

```
--source-identifier test-instance \
--source-type db-instance
```

### Output:

```
{
  "Events": [
    {
      "SourceType": "db-instance",
      "SourceIdentifier": "test-instance",
      "EventCategories": [
        "backup"
      ],
      "Message": "Backing up DB instance",
      "Date": "2018-07-31T23:09:23.983Z",
      "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
    },
    {
      "SourceType": "db-instance",
      "SourceIdentifier": "test-instance",
      "EventCategories": [
        "backup"
      ],
      "Message": "Finished DB Instance backup",
      "Date": "2018-07-31T23:15:13.049Z",
      "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## describe-export-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-export-tasks`.

### AWS CLI

Untuk menggambarkan tugas ekspor snapshot

`describe-export-tasks` Contoh berikut mengembalikan informasi tentang ekspor snapshot ke Amazon S3.

```
aws rds describe-export-tasks
```

**Output:**

```
{
  "ExportTasks": [
    {
      "ExportTaskIdentifier": "test-snapshot-export",
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:test-
snapshot",
      "SnapshotTime": "2020-03-02T18:26:28.163Z",
      "TaskStartTime": "2020-03-02T18:57:56.896Z",
      "TaskEndTime": "2020-03-02T19:10:31.985Z",
      "S3Bucket": "mybucket",
      "S3Prefix": "",
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
      "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
abcd0000-7fca-4128-82f2-aabbccddeeff",
      "Status": "COMPLETE",
      "PercentProgress": 100,
      "TotalExtractedDataInGB": 0
    },
    {
      "ExportTaskIdentifier": "my-s3-export",
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-
test",
      "SnapshotTime": "2020-03-27T20:48:42.023Z",
      "S3Bucket": "mybucket",
      "S3Prefix": "",
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
      "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
abcd0000-7fca-4128-82f2-aabbccddeeff",
      "Status": "STARTING",
      "PercentProgress": 0,
      "TotalExtractedDataInGB": 0
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memantau Ekspor Snapshot](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeExportTasks](#) di Referensi AWS CLI Perintah.

## describe-global-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-global-clusters`.

### AWS CLI

Untuk menggambarkan cluster DB global

`describe-global-clusters` Contoh berikut mencantumkan cluster DB global Aurora di Wilayah saat ini. AWS

```
aws rds describe-global-clusters
```

Output:

```
{
  "GlobalClusters": [
    {
      "GlobalClusterIdentifier": "myglobalcluster",
      "GlobalClusterResourceId": "cluster-f5982077e3b5aabb",
      "GlobalClusterArn": "arn:aws:rds::123456789012:global-cluster:myglobalcluster",
      "Status": "available",
      "Engine": "aurora-mysql",
      "EngineVersion": "5.7.mysql_aurora.2.07.2",
      "StorageEncrypted": false,
      "DeletionProtection": false,
      "GlobalClusterMembers": []
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola database global Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [DescribeGlobalClusters](#) di Referensi AWS CLI Perintah.

## describe-option-group-options

Contoh kode berikut menunjukkan cara menggunakan `describe-option-group-options`.

## AWS CLI

Untuk menjelaskan semua opsi yang tersedia

`describe-option-group-options` Contoh berikut mencantumkan dua opsi untuk contoh Oracle Database 19c.

```
aws rds describe-option-group-options \  
  --engine-name oracle-ee \  
  --major-engine-version 19 \  
  --max-items 2
```

Output:

```
{  
  "OptionGroupOptions": [  
    {  
      "Name": "APEX",  
      "Description": "Oracle Application Express Runtime Environment",  
      "EngineName": "oracle-ee",  
      "MajorEngineVersion": "19",  
      "MinimumRequiredMinorEngineVersion": "0.0.0.ru-2019-07.rur-2019-07.r1",  
      "PortRequired": false,  
      "OptionsDependedOn": [],  
      "OptionsConflictsWith": [],  
      "Persistent": false,  
      "Permanent": false,  
      "RequiresAutoMinorEngineVersionUpgrade": false,  
      "VpcOnly": false,  
      "SupportsOptionVersionDowngrade": false,  
      "OptionGroupOptionSettings": [],  
      "OptionGroupOptionVersions": [  
        {  
          "Version": "19.1.v1",  
          "IsDefault": true  
        },  
        {  
          "Version": "19.2.v1",  
          "IsDefault": false  
        }  
      ]  
    },  
    {
```

```

    "Name": "APEX-DEV",
    "Description": "Oracle Application Express Development Environment",
    "EngineName": "oracle-ee",
    "MajorEngineVersion": "19",
    "MinimumRequiredMinorEngineVersion": "0.0.0.ru-2019-07.rur-2019-07.r1",
    "PortRequired": false,
    "OptionsDependedOn": [
        "APEX"
    ],
    "OptionsConflictsWith": [],
    "Persistent": false,
    "Permanent": false,
    "RequiresAutoMinorEngineVersionUpgrade": false,
    "VpcOnly": false,
    "OptionGroupOptionSettings": []
  }
],
"NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

Untuk informasi selengkapnya, lihat [Daftar Pengaturan Opsi dan Opsi untuk Grup Opsi](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeOptionGroupOptions](#) di Referensi AWS CLI Perintah.

## describe-option-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-option-groups`.

### AWS CLI

Untuk mendeskripsikan grup opsi yang tersedia

`describe-option-groups` Contoh berikut mencantumkan grup opsi untuk contoh Oracle Database 19c.

```

aws rds describe-option-groups \
  --engine-name oracle-ee \
  --major-engine-version 19

```

Output:

```
{
```

```

"OptionGroupsList": [
  {
    "OptionGroupName": "default:oracle-ee-19",
    "OptionGroupDescription": "Default option group for oracle-ee 19",
    "EngineName": "oracle-ee",
    "MajorEngineVersion": "19",
    "Options": [],
    "AllowsVpcAndNonVpcInstanceMemberships": true,
    "OptionGroupArn": "arn:aws:rds:us-west-1:111122223333:og:default:oracle-
ee-19"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Daftar Pengaturan Opsi dan Opsi untuk Grup Opsi](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeOptionGroups](#) di Referensi AWS CLI Perintah.

## describe-orderable-db-instance-options

Contoh kode berikut menunjukkan cara menggunakan `describe-orderable-db-instance-options`.

### AWS CLI

Untuk menjelaskan opsi instans DB yang dapat dipesan

`describe-orderable-db-instance-options` Contoh berikut mengambil rincian tentang opsi yang dapat dipesan untuk instance DB yang menjalankan mesin MySQL DB.

```

aws rds describe-orderable-db-instance-options \
  --engine mysql

```

Output:

```

{
  "OrderableDBInstanceOptions": [
    {
      "MinStorageSize": 5,
      "ReadReplicaCapable": true,
      "MaxStorageSize": 6144,

```

```

    "AvailabilityZones": [
      {
        "Name": "us-east-1a"
      },
      {
        "Name": "us-east-1b"
      },
      {
        "Name": "us-east-1c"
      },
      {
        "Name": "us-east-1d"
      }
    ],
    "SupportsIops": false,
    "AvailableProcessorFeatures": [],
    "MultiAZCapable": true,
    "DBInstanceClass": "db.m1.large",
    "Vpc": true,
    "StorageType": "gp2",
    "LicenseModel": "general-public-license",
    "EngineVersion": "5.5.46",
    "SupportsStorageEncryption": false,
    "SupportsEnhancedMonitoring": true,
    "Engine": "mysql",
    "SupportsIAMDatabaseAuthentication": false,
    "SupportsPerformanceInsights": false
  }
]
...some output truncated...
}

```

- Untuk detail API, lihat [DescribeOrderableDB InstanceOptions](#) di Referensi AWS CLI Perintah.

## describe-pending-maintenance-actions

Contoh kode berikut menunjukkan cara menggunakan `describe-pending-maintenance-actions`.

### AWS CLI

Untuk membuat daftar sumber daya dengan setidaknya satu tindakan pemeliharaan yang tertunda



`describe-pending-maintenance-actions` Contoh berikut mencantumkan tindakan pemeliharaan tertunda untuk instance DB.

```
aws rds describe-pending-maintenance-actions
```

Output:

```
{
  "PendingMaintenanceActions": [
    {
      "ResourceIdentifier": "arn:aws:rds:us-
west-2:123456789012:cluster:global-db1-cl1",
      "PendingMaintenanceActionDetails": [
        {
          "Action": "system-update",
          "Description": "Upgrade to Aurora PostgreSQL 2.4.2"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mempertahankan Instans DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribePendingMaintenanceActions](#) di Referensi AWS CLI Perintah.

## **describe-reserved-db-instances-offerings**

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-db-instances-offerings`.

AWS CLI

Untuk menggambarkan penawaran instans DB yang dipesan

`describe-reserved-db-instances-offerings` Contoh berikut mengambil rincian tentang opsi instans DB yang dicadangkan untuk `oracle`.

```
aws rds describe-reserved-db-instances-offerings \
```

```
--product-description oracle
```

Output:

```
{
  "ReservedDBInstancesOfferings": [
    {
      "CurrencyCode": "USD",
      "UsagePrice": 0.0,
      "ProductDescription": "oracle-se2(li)",
      "ReservedDBInstancesOfferingId": "005bdee3-9ef4-4182-aa0c-58ef7cb6c2f8",
      "MultiAZ": true,
      "DBInstanceClass": "db.m4.xlarge",
      "OfferingType": "Partial Upfront",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": 0.594,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "FixedPrice": 4089.0,
      "Duration": 31536000
    },
    ...some output truncated...
  ]
}
```

- Untuk detail API, lihat [DescribeReservedDbInstancesOfferings](#) di Referensi AWS CLI Perintah.

## describe-reserved-db-instances

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-db-instances`.

### AWS CLI

Untuk menggambarkan instans DB yang dicadangkan

`describe-reserved-db-instances` Contoh berikut mengambil rincian tentang instans DB yang dicadangkan di akun saat ini AWS .

```
aws rds describe-reserved-db-instances
```

Output:

```
{
  "ReservedDBInstances": [
    {
      "ReservedDBInstanceId": "myreservedinstance",
      "ReservedDBInstancesOfferingId": "12ab34cd-59af-4b2c-a660-1abcdef23456",
      "DBInstanceClass": "db.t3.micro",
      "StartTime": "2020-06-01T13:44:21.436Z",
      "Duration": 31536000,
      "FixedPrice": 0.0,
      "UsagePrice": 0.0,
      "CurrencyCode": "USD",
      "DBInstanceCount": 1,
      "ProductDescription": "sqlserver-ex(li)",
      "OfferingType": "No Upfront",
      "MultiAZ": false,
      "State": "payment-pending",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": 0.014,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "ReservedDBInstanceArn": "arn:aws:rds:us-west-2:123456789012:ri:myreservedinstance",
      "LeaseId": "a1b2c3d4-6b69-4a59-be89-5e11aa446666"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Instans DB Cadangan untuk Amazon RDS](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [DescribeReservedDbInstances](#) di Referensi AWS CLI Perintah.

## describe-source-regions

Contoh kode berikut menunjukkan cara menggunakan `describe-source-regions`.

### AWS CLI

Untuk mendeskripsikan daerah sumber

`describe-source-regions` Contoh berikut mengambil rincian tentang semua AWS Wilayah sumber. Ini juga menunjukkan bahwa pencadangan otomatis hanya dapat direplikasi dari AS Barat (Oregon) ke AWS Wilayah tujuan, AS Timur (Virginia N.).

```
aws rds describe-source-regions \  
  --region us-east-1
```

Output:

```
{  
  "SourceRegions": [  
    {  
      "RegionName": "af-south-1",  
      "Endpoint": "https://rds.af-south-1.amazonaws.com",  
      "Status": "available",  
      "SupportsDBInstanceAutomatedBackupsReplication": false  
    },  
    {  
      "RegionName": "ap-east-1",  
      "Endpoint": "https://rds.ap-east-1.amazonaws.com",  
      "Status": "available",  
      "SupportsDBInstanceAutomatedBackupsReplication": false  
    },  
    {  
      "RegionName": "ap-northeast-1",  
      "Endpoint": "https://rds.ap-northeast-1.amazonaws.com",  
      "Status": "available",  
      "SupportsDBInstanceAutomatedBackupsReplication": true  
    },  
    {  
      "RegionName": "ap-northeast-2",  
      "Endpoint": "https://rds.ap-northeast-2.amazonaws.com",  
      "Status": "available",  
      "SupportsDBInstanceAutomatedBackupsReplication": true  
    },  
    {  
      "RegionName": "ap-northeast-3",  
      "Endpoint": "https://rds.ap-northeast-3.amazonaws.com",  
      "Status": "available",  
      "SupportsDBInstanceAutomatedBackupsReplication": false  
    },  
    {  
      "RegionName": "ap-south-1",
```

```
    "Endpoint": "https://rds.ap-south-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "ap-southeast-1",
    "Endpoint": "https://rds.ap-southeast-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "ap-southeast-2",
    "Endpoint": "https://rds.ap-southeast-2.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "ap-southeast-3",
    "Endpoint": "https://rds.ap-southeast-3.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": false
  },
  {
    "RegionName": "ca-central-1",
    "Endpoint": "https://rds.ca-central-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-north-1",
    "Endpoint": "https://rds.eu-north-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-south-1",
    "Endpoint": "https://rds.eu-south-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": false
  },
  {
    "RegionName": "eu-west-1",
    "Endpoint": "https://rds.eu-west-1.amazonaws.com",
    "Status": "available",
```

```
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-west-2",
    "Endpoint": "https://rds.eu-west-2.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-west-3",
    "Endpoint": "https://rds.eu-west-3.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "me-central-1",
    "Endpoint": "https://rds.me-central-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": false
  },
  {
    "RegionName": "me-south-1",
    "Endpoint": "https://rds.me-south-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": false
  },
  {
    "RegionName": "sa-east-1",
    "Endpoint": "https://rds.sa-east-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "us-east-2",
    "Endpoint": "https://rds.us-east-2.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "us-west-1",
    "Endpoint": "https://rds.us-west-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
},
```

```
{
  "RegionName": "us-west-2",
  "Endpoint": "https://rds.us-west-2.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": true
}
]
```

Untuk informasi selengkapnya, lihat [Menemukan informasi tentang cadangan yang direplikasi di Panduan Pengguna Amazon RDS](#).

- Untuk detail API, lihat [DescribeSourceRegions](#) di Referensi AWS CLI Perintah.

## **describe-valid-db-instance-modifications**

Contoh kode berikut menunjukkan cara menggunakan `describe-valid-db-instance-modifications`.

### AWS CLI

Untuk menjelaskan modifikasi yang valid untuk instans DB

`describe-valid-db-instance-modifications` Contoh berikut mengambil rincian tentang modifikasi yang valid untuk instans DB tertentu.

```
aws rds describe-valid-db-instance-modifications \
  --db-instance-identifier test-instance
```

### Output:

```
{
  "ValidDBInstanceModificationsMessage": {
    "ValidProcessorFeatures": [],
    "Storage": [
      {
        "StorageSize": [
          {
            "Step": 1,
            "To": 20,
            "From": 20
          },
          {
```

```
        "Step": 1,
        "To": 6144,
        "From": 22
      }
    ],
    "ProvisionedIops": [
      {
        "Step": 1,
        "To": 0,
        "From": 0
      }
    ],
    "IopsToStorageRatio": [
      {
        "To": 0.0,
        "From": 0.0
      }
    ],
    "StorageType": "gp2"
  },
  {
    "StorageSize": [
      {
        "Step": 1,
        "To": 6144,
        "From": 100
      }
    ],
    "ProvisionedIops": [
      {
        "Step": 1,
        "To": 40000,
        "From": 1000
      }
    ],
    "IopsToStorageRatio": [
      {
        "To": 50.0,
        "From": 1.0
      }
    ],
    "StorageType": "io1"
  },
  {
```



```

        "StorageSize": [
            {
                "Step": 1,
                "To": 20,
                "From": 20
            },
            {
                "Step": 1,
                "To": 3072,
                "From": 22
            }
        ],
        "ProvisionedIops": [
            {
                "Step": 1,
                "To": 0,
                "From": 0
            }
        ],
        "IopsToStorageRatio": [
            {
                "To": 0.0,
                "From": 0.0
            }
        ],
        "StorageType": "magnetic"
    }
]
}

```

- Untuk detail API, lihat [DescribeValidDbInstanceModifications](#) di Referensi AWS CLI Perintah.

## download-db-log-file-portion

Contoh kode berikut menunjukkan cara menggunakan `download-db-log-file-portion`.

### AWS CLI

Untuk mengunduh file log DB

`download-db-log-file-portion` Contoh berikut hanya mengunduh bagian terbaru dari file log Anda, menyimpannya ke file lokal bernama `tail.txt`.

```
aws rds download-db-log-file-portion \  
  --db-instance-identifier test-instance \  
  --log-file-name log.txt \  
  --output text > tail.txt
```

Untuk mengunduh seluruh file, Anda harus menyertakan `--starting-token 0` parameter. Contoh berikut menyimpan output ke file lokal bernama `full.txt`.

```
aws rds download-db-log-file-portion \  
  --db-instance-identifier test-instance \  
  --log-file-name log.txt \  
  --starting-token 0 \  
  --output text > full.txt
```

File yang disimpan mungkin berisi baris kosong. Mereka muncul di akhir setiap bagian dari file log saat sedang diunduh. Ini umumnya tidak menyebabkan masalah dalam analisis file log Anda.

- Untuk detail API, lihat [DownloadDbLogFilePortion](#) di Referensi AWS CLI Perintah.

## generate-auth-token

Contoh kode berikut menunjukkan cara menggunakan `generate-auth-token`.

### AWS CLI

Untuk menghasilkan token otentikasi

`generate-db-auth-token` Contoh berikut menghasilkan token otentikasi untuk digunakan dengan otentikasi database IAM.

```
aws rds generate-db-auth-token \  
  --hostname aurmysql-test.cdgmuiadpid.us-west-2.rds.amazonaws.com \  
  --port 3306 \  
  --region us-east-1 \  
  --username jane_doe
```

Output:

```
aurmysql-test.cdgmuiadpid.us-west-2.rds.amazonaws.com:3306/?  
Action=connect&DBUser=jane_doe&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
```

```
Credential=AKIAIESZCNJ30EXAMPLE%2F20180731%2Fus-east-1%2Frds-db%2Faws4_request&X-
Amz-Date=20180731T235209Z&X-Amz-Expires=900&X-Amz-SignedHeaders=host&X-Amz-
Signature=5a8753ebEXAMPLEa2c724e5667797EXAMPLE9d6ec6e3f427191fa41aeEXAMPLE
```

- Untuk detail API, lihat [GenerateAuthToken](#) di Referensi AWS CLI Perintah.

## generate-db-auth-token

Contoh kode berikut menunjukkan cara menggunakan `generate-db-auth-token`.

### AWS CLI

Untuk menghasilkan token otentikasi IAM

`generate-db-auth-token` Contoh berikut menghasilkan token otentikasi IAM untuk terhubung ke database.

```
aws rds generate-db-auth-token \
  --hostname mydb.123456789012.us-east-1.rds.amazonaws.com \
  --port 3306 \
  --region us-east-1 \
  --username db_user
```

Output:

```
mydb.123456789012.us-east-1.rds.amazonaws.com:3306/?
Action=connect&DBUser=db_user&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAIEXAMPLE%2Fus-east-1%2Frds-db%2Faws4_request&X-Amz-
Date=20210123T011543Z&X-Amz-Expires=900&X-Amz-SignedHeaders=host&X-Amz-
Signature=88987EXAMPLE1EXAMPLE2EXAMPLE3EXAMPLE4EXAMPLE5EXAMPLE6
```

Untuk informasi selengkapnya, lihat [Menyambungkan ke instans DB menggunakan autentikasi IAM](#) di Panduan Pengguna Amazon RDS dan [Menyambungkan ke cluster DB menggunakan autentikasi IAM di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [GenerateDbAuthToken](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk mencantumkan tag pada sumber daya Amazon RDS

`list-tags-for-resource` Contoh berikut mencantumkan semua tag pada instance DB.

```
aws rds list-tags-for-resource \  
  --resource-name arn:aws:rds:us-east-1:123456789012:db:orc11
```

Output:

```
{  
  "TagList": [  
    {  
      "Key": "Environment",  
      "Value": "test"  
    },  
    {  
      "Key": "Name",  
      "Value": "MyDatabase"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Amazon RDS](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **modify-certificates**

Contoh kode berikut menunjukkan cara menggunakan `modify-certificates`.

## AWS CLI

Untuk mengganti sementara sertifikat SSL/TLS default sistem untuk instans DB baru

`modify-certificates` Contoh berikut untuk sementara mengesampingkan sertifikat SSL/TLS system-default untuk instance DB baru.

```
aws rds modify-certificates \  
  --certificate-identifier rds-ca-2019
```

**Output:**

```
{
  "Certificate": {
    "CertificateIdentifier": "rds-ca-2019",
    "CertificateType": "CA",
    "Thumbprint": "EXAMPLE123456789012",
    "ValidFrom": "2019-09-19T18:16:53Z",
    "ValidTill": "2024-08-22T17:08:50Z",
    "CertificateArn": "arn:aws:rds:us-east-1::cert:rds-ca-2019",
    "CustomerOverride": true,
    "CustomerOverrideValidTill": "2024-08-22T17:08:50Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Memutar sertifikat SSL/TLS Anda di Panduan Pengguna Amazon RDS](#) dan [Memutar sertifikat SSL/TLS Anda di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [ModifyCertificates](#) di Referensi AWS CLI Perintah.

**modify-current-db-cluster-capacity**

Contoh kode berikut menunjukkan cara menggunakan `modify-current-db-cluster-capacity`.

**AWS CLI**

Untuk menskalakan kapasitas cluster DB Aurora Serverless

`modify-current-db-cluster-capacity` Contoh berikut menskalakan kapasitas cluster DB Tanpa Server Aurora menjadi 8.

```
aws rds modify-current-db-cluster-capacity \
  --db-cluster-identifier mydbcluster \
  --capacity 8
```

**Output:**

```
{
  "DBClusterIdentifier": "mydbcluster",
  "PendingCapacity": 8,
  "CurrentCapacity": 1,
  "SecondsBeforeTimeout": 300,
```

```
"TimeoutAction": "ForceApplyCapacityChange"
}
```

Untuk informasi selengkapnya, lihat [Menskalakan kapasitas klaster V1 DB Tanpa Server Aurora secara manual di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyCurrentDbClusterCapacity](#) di Referensi AWS CLI Perintah.

## modify-db-cluster-endpoint

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster-endpoint`.

### AWS CLI

Untuk memodifikasi titik akhir cluster DB kustom

`modify-db-cluster-endpoint` Contoh berikut memodifikasi titik akhir cluster DB kustom yang ditentukan.

```
aws rds modify-db-cluster-endpoint \
  --db-cluster-endpoint-identifier mycustomendpoint \
  --static-members dbinstance1 dbinstance2 dbinstance3
```

Output:

```
{
  "DBClusterEndpointIdentifier": "mycustomendpoint",
  "DBClusterIdentifier": "mydbcluster",
  "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
  "Endpoint": "mycustomendpoint.cluster-custom-cnpexample.us-east-1.rds.amazonaws.com",
  "Status": "modifying",
  "EndpointType": "CUSTOM",
  "CustomEndpointType": "READER",
  "StaticMembers": [
    "dbinstance1",
    "dbinstance2",
    "dbinstance3"
  ],
  "ExcludedMembers": [],
  "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:mycustomendpoint"
```

```
}
```

Untuk informasi selengkapnya, lihat [Manajemen Koneksi Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbClusterEndpoint](#) di Referensi AWS CLI Perintah.

## **modify-db-cluster-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster-parameter-group`.

### AWS CLI

Untuk memodifikasi parameter dalam kelompok parameter cluster DB

`modify-db-cluster-parameter-group` Contoh berikut memodifikasi nilai-nilai parameter dalam kelompok parameter cluster DB.

```
aws rds modify-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name mydbclusterpg \  
  --parameters  
  "ParameterName=server_audit_logging,ParameterValue=1,ApplyMethod=immediate" \  
  
  "ParameterName=server_audit_logs_upload,ParameterValue=1,ApplyMethod=immediate"
```

Output:

```
{  
  "DBClusterParameterGroupName": "mydbclusterpg"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup parameter DB dan grup parameter cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## **modify-db-cluster-snapshot-attribute**

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster-snapshot-attribute`.

## AWS CLI

Untuk memodifikasi atribut snapshot cluster DB

`modify-db-cluster-snapshot-attribute` Contoh berikut membuat perubahan pada atribut snapshot cluster DB yang ditentukan.

```
aws rds modify-db-cluster-snapshot-attribute \  
  --db-cluster-snapshot-identifier myclustersnapshot \  
  --attribute-name restore \  
  --values-to-add 123456789012
```

Output:

```
{  
  "DBClusterSnapshotAttributesResult": {  
    "DBClusterSnapshotIdentifier": "myclustersnapshot",  
    "DBClusterSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "123456789012"  
        ]  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memulihkan dari Snapshot Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbClusterSnapshotAttribute](#) di Referensi AWS CLI Perintah.

## `modify-db-cluster`

Contoh kode berikut menunjukkan cara menggunakan `modify-db-cluster`.

### AWS CLI

Contoh 1: Untuk memodifikasi cluster DB

`modify-db-cluster` Contoh berikut mengubah kata sandi pengguna utama untuk cluster DB bernama `cluster-2` dan menetapkan periode retensi cadangan menjadi 14 hari. `--apply-`



`immediatelyParameter` menyebabkan perubahan segera dilakukan, alih-alih menunggu hingga jendela pemeliharaan berikutnya.

```
aws rds modify-db-cluster \  
  --db-cluster-identifier cluster-2 \  
  --backup-retention-period 14 \  
  --master-user-password newpassword99 \  
  --apply-immediately
```

Output:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "eu-central-1b",  
      "eu-central-1c",  
      "eu-central-1a"  
    ],  
    "BackupRetentionPeriod": 14,  
    "DatabaseName": "",  
    "DBClusterIdentifier": "cluster-2",  
    "DBClusterParameterGroup": "default.aurora5.6",  
    "DBSubnetGroup": "default-vpc-2305ca49",  
    "Status": "available",  
    "EarliestRestorableTime": "2020-06-03T02:07:29.637Z",  
    "Endpoint": "cluster-2.cluster-#####.eu-central-1.rds.amazonaws.com",  
    "ReaderEndpoint": "cluster-2.cluster-ro-#####.eu-  
central-1.rds.amazonaws.com",  
    "MultiAZ": false,  
    "Engine": "aurora",  
    "EngineVersion": "5.6.10a",  
    "LatestRestorableTime": "2020-06-04T15:11:25.748Z",  
    "Port": 3306,  
    "MasterUsername": "admin",  
    "PreferredBackupWindow": "01:55-02:25",  
    "PreferredMaintenanceWindow": "thu:21:14-thu:21:44",  
    "ReadReplicaIdentifiers": [],  
    "DBClusterMembers": [  
      {  
        "DBInstanceIdentifier": "cluster-2-instance-1",  
        "IsClusterWriter": true,  
        "DBClusterParameterGroupStatus": "in-sync",
```

```

        "PromotionTier": 1
      }
    ],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-20a5c047",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z1RLNU0EXAMPLE",
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:eu-central-1:123456789012:key/d1bd7c8f-5cdb-49ca-8a62-a1b2c3d4e5f6",
    "DbClusterResourceId": "cluster-AGJ7XI77XVIS6FUXHU1EXAMPLE",
    "DBClusterArn": "arn:aws:rds:eu-central-1:123456789012:cluster:cluster-2",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "ClusterCreateTime": "2020-04-03T14:44:02.764Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": true,
    "CrossAccountClone": false,
    "DomainMemberships": []
  }
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Cluster DB Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

Contoh 2: Untuk mengaitkan grup keamanan VPC dengan cluster DB

`modify-db-instance` Contoh berikut mengaitkan grup keamanan VPC tertentu dan menghapus grup keamanan DB dari cluster DB.

```

aws rds modify-db-cluster \
  --db-cluster-identifier dbName \
  --vpc-security-group-ids sg-ID

```

Output:

```

{
  "DBCluster": {

```

```
    "AllocatedStorage": 1,
    "AvailabilityZones": [
      "us-west-2c",
      "us-west-2b",
      "us-west-2a"
    ],
    "BackupRetentionPeriod": 1,
    "DBClusterIdentifier": "dbName",
    "DBClusterParameterGroup": "default.aurora-mysql8.0",
    "DBSubnetGroup": "default",
    "Status": "available",
    "EarliestRestorableTime": "2024-02-15T01:12:13.966000+00:00",
    "Endpoint": "dbName.cluster-abcdefghji.us-west-2.rds.amazonaws.com",
    "ReaderEndpoint": "dbName.cluster-ro-abcdefghji.us-
west-2.rds.amazonaws.com",
    "MultiAZ": false,
    "Engine": "aurora-mysql",
    "EngineVersion": "8.0.mysql_aurora.3.04.1",
    "LatestRestorableTime": "2024-02-15T02:25:33.696000+00:00",
    "Port": 3306,
    "MasterUsername": "admin",
    "PreferredBackupWindow": "10:59-11:29",
    "PreferredMaintenanceWindow": "thu:08:54-thu:09:24",
    "ReadReplicaIdentifiers": [],
    "DBClusterMembers": [
      {
        "DBInstanceIdentifier": "dbName-instance-1",
        "IsClusterWriter": true,
        "DBClusterParameterGroupStatus": "in-sync",
        "PromotionTier": 1
      }
    ],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-ID",
        "Status": "active"
      }
    ],
    ...output omitted...
  }
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses dengan grup keamanan](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbCluster](#) di Referensi AWS CLI Perintah.

## modify-db-instance

Contoh kode berikut menunjukkan cara menggunakan `modify-db-instance`.

### AWS CLI

Contoh 1: Untuk memodifikasi instans DB

`modify-db-instance` Contoh berikut mengaitkan grup opsi dan grup parameter dengan instance Microsoft SQL Server DB yang kompatibel. `--apply-immediately` menyebabkan grup opsi dan parameter segera dikaitkan, alih-alih menunggu hingga jendela pemeliharaan berikutnya.

```
aws rds modify-db-instance \  
  --db-instance-identifier database-2 \  
  --option-group-name test-se-2017 \  
  --db-parameter-group-name test-sqlserver-se-2017 \  
  --apply-immediately
```

Output:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "database-2",  
    "DBInstanceClass": "db.r4.large",  
    "Engine": "sqlserver-se",  
    "DBInstanceStatus": "available",  
  
    ...output omitted...  
  
    "DBParameterGroups": [  
      {  
        "DBParameterGroupName": "test-sqlserver-se-2017",  
        "ParameterApplyStatus": "applying"  
      }  
    ],  
    "AvailabilityZone": "us-west-2d",
```

```

...output omitted...

"MultiAZ": true,
"EngineVersion": "14.00.3281.6.v1",
"AutoMinorVersionUpgrade": false,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "license-included",
"OptionGroupMemberships": [
  {
    "OptionGroupName": "test-se-2017",
    "Status": "pending-apply"
  }
],
"CharacterSetName": "SQL_Latin1_General_CP1_CI_AS",
"SecondaryAvailabilityZone": "us-west-2c",
"PubliclyAccessible": true,
"StorageType": "gp2",

...output omitted...

"DeletionProtection": false,
"AssociatedRoles": [],
"MaxAllocatedStorage": 1000
}
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi Instans Amazon RDS DB](#) di Panduan Pengguna Amazon RDS.

Contoh 2: Untuk mengaitkan grup keamanan VPC dengan instans DB

`modify-db-instance` Contoh berikut mengaitkan grup keamanan VPC tertentu dan menghapus grup keamanan DB dari instans DB:

```

aws rds modify-db-instance \
  --db-instance-identifier dbName \
  --vpc-security-group-ids sg-ID

```

Output:

```

{
  "DBInstance": {
    "DBInstanceIdentifier": "dbName",

```

```
"DBInstanceClass": "db.t3.micro",
"Engine": "mysql",
"DBInstanceStatus": "available",
"MasterUsername": "admin",
"Endpoint": {
  "Address": "dbName.abcdefghijkl.us-west-2.rds.amazonaws.com",
  "Port": 3306,
  "HostedZoneId": "ABCDEFGHIJK1234"
},
"AllocatedStorage": 20,
"InstanceCreateTime": "2024-02-15T00:37:58.793000+00:00",
"PreferredBackupWindow": "11:57-12:27",
"BackupRetentionPeriod": 7,
"DBSecurityGroups": [],
"VpcSecurityGroups": [
  {
    "VpcSecurityGroupId": "sg-ID",
    "Status": "active"
  }
],
... output omitted ...
"MultiAZ": false,
"EngineVersion": "8.0.35",
"AutoMinorVersionUpgrade": true,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "general-public-license",

... output omitted ...
}
}
```

Untuk informasi selengkapnya, lihat [Mengontrol akses dengan grup keamanan](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ModifyDBInstance](#) di Referensi Perintah.AWS CLI

## **modify-db-parameter-group**

Contoh kode berikut menunjukkan cara menggunakan `modify-db-parameter-group`.

### AWS CLI

Untuk memodifikasi grup parameter DB

`modify-db-parameter-group` Contoh berikut mengubah nilai parameter dalam kelompok `clr` `enabled` parameter DB. `--apply-immediately` Parameter menyebabkan grup parameter DB segera dimodifikasi, alih-alih menunggu hingga jendela pemeliharaan berikutnya.

```
aws rds modify-db-parameter-group \  
  --db-parameter-group-name test-sqlserver-se-2017 \  
  --parameters "ParameterName='clr  
enabled',ParameterValue=1,ApplyMethod=immediate"
```

Output:

```
{  
  "DBParameterGroupName": "test-sqlserver-se-2017"  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Parameter dalam Grup Parameter DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ModifyDB ParameterGroup](#) di AWS CLI Referensi Perintah.

## **modify-db-proxy-endpoint**

Contoh kode berikut menunjukkan cara menggunakan `modify-db-proxy-endpoint`.

AWS CLI

Untuk memodifikasi titik akhir proxy DB untuk database RDS

`modify-db-proxy-endpoint` Contoh berikut memodifikasi titik akhir proxy DB `proxyEndpoint` untuk mengatur waktu baca menjadi 65 detik.

```
aws rds modify-db-proxy-endpoint \  
  --db-proxy-endpoint-name proxyEndpoint \  
  --cli-read-timeout 65
```

Output:

```
{  
  "DBProxyEndpoint":  
    {  
      "DBProxyEndpointName": "proxyEndpoint",
```

```

    "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
    "DBProxyName": "proxyExample",
    "Status": "available",
    "VpcId": "vpc-1234567",
    "VpcSecurityGroupIds": [
        "sg-1234"
    ],
    "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
    ],
    "Endpoint": "proxyEndpoint.endpoint.proxyExample-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",
    "TargetRole": "READ_WRITE",
    "IsDefault": "false"
}
}

```

Untuk informasi selengkapnya, lihat [Memodifikasi titik akhir proxy](#) di Panduan Pengguna Amazon RDS dan [Memodifikasi titik akhir proxy di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbProxyEndpoint](#) di Referensi AWS CLI Perintah.

## modify-db-proxy-target-group

Contoh kode berikut menunjukkan cara menggunakan `modify-db-proxy-target-group`.

### AWS CLI

Untuk memodifikasi titik akhir proxy DB

`modify-db-proxy-target-group` Contoh berikut memodifikasi grup target proxy DB untuk mengatur koneksi maksimum hingga 80 persen dan koneksi idle maksimum menjadi 10 persen.

```

aws rds modify-db-proxy-target-group \
  --target-group-name default \
  --db-proxy-name proxyExample \
  --connection-pool-config MaxConnectionsPercent=80,MaxIdleConnectionsPercent=10

```

Output:



```
{
  "DBProxyTargetGroup":
    {
      "DBProxyName": "proxyExample",
      "TargetGroupName": "default",
      "TargetGroupArn": "arn:aws:rds:us-east-1:123456789012:target-group:prx-
tg-0123a01b12345c0ab",
      "IsDefault": true,
      "Status": "available",
      "ConnectionPoolConfig": {
        "MaxConnectionsPercent": 80,
        "MaxIdleConnectionsPercent": 10,
        "ConnectionBorrowTimeout": 120,
        "SessionPinningFilters": []
      },
      "CreateDate": "2023-05-02T18:41:19.495000+00:00",
      "UpdateDate": "2023-05-02T18:41:21.762000+00:00"
    }
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Memodifikasi Proxy RDS di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [ModifyDbProxyTargetGroup](#) di Referensi AWS CLI Perintah.

## modify-db-proxy

Contoh kode berikut menunjukkan cara menggunakan `modify-db-proxy`.

### AWS CLI

Untuk memodifikasi proxy DB untuk database RDS

`modify-db-proxy` Contoh berikut memodifikasi proxy DB bernama `proxyExample` memerlukan SSL untuk koneksinya.

```
aws rds modify-db-proxy \
  --db-proxy-name proxyExample \
  --require-tls
```

Output:

```
{
  "DBProxy":
    {
      "DBProxyName": "proxyExample",
      "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
      "Status": "modifying"
      "EngineFamily": "PostgreSQL",
      "VpcId": "sg-1234567",
      "VpcSecurityGroupIds": [
        "sg-1234"
      ],
      "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
      ],
      "Auth": "[
        {
          "Description": "proxydescription1",
          "AuthScheme": "SECRETS",
          "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxysecret1-Abcd1e",
          "IAMAuth": "DISABLED"
        }
      ]",
      "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
      "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
      "RequireTLS": true,
      "IdleClientTimeout": 1800,
      "DebuggingLogging": false,
      "CreateDate": "2023-04-05T16:09:33.452000+00:00",
      "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
    }
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi Proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Membuat Proxy RDS](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbProxy](#) di Referensi AWS CLI Perintah.

## modify-db-shard-group

Contoh kode berikut menunjukkan cara menggunakan `modify-db-shard-group`.

## AWS CLI

Contoh 1: Untuk memodifikasi grup pecahan DB

`modify-db-shard-group` Contoh berikut mengubah kapasitas maksimum grup shard DB.

```
aws rds modify-db-shard-group \  
  --db-shard-group-identifier my-db-shard-group \  
  --max-acu 1000
```

Output:

```
{  
  "DBShardGroups": [  
    {  
      "DBShardGroupResourceId": "shardgroup-a6e3a02226aa243e2ac6c7a1234567890",  
      "DBShardGroupIdentifier": "my-db-shard-group",  
      "DBClusterIdentifier": "my-sv2-cluster",  
      "MaxACU": 768.0,  
      "ComputeRedundancy": 0,  
      "Status": "available",  
      "PubliclyAccessible": false,  
      "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Cluster DB Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

Contoh 2: Untuk menggambarkan grup pecahan DB Anda

`describe-db-shard-groups` Contoh berikut mengambil rincian grup shard DB Anda setelah Anda menjalankan perintah `modify-db-shard-group` Kapasitas maksimum grup `my-db-shard-group` pecahan DB sekarang 1000 unit kapasitas Aurora (ACU).

```
aws rds describe-db-shard-groups
```

Output:

```
{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
      "DBShardGroupIdentifier": "limitless-test-shard-grp",
      "DBClusterIdentifier": "limitless-test-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": true,
      "Endpoint": "limitless-test-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
    },
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",
      "MaxACU": 1000.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": false,
      "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Cluster DB Amazon Aurora di Panduan](#) Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyDbShardGroup](#) di Referensi AWS CLI Perintah.

## modify-db-snapshot-attribute

Contoh kode berikut menunjukkan cara menggunakan `modify-db-snapshot-attribute`.

### AWS CLI

Contoh 1: Untuk mengaktifkan dua AWS akun untuk memulihkan snapshot DB

`modify-db-snapshot-attribute` Contoh berikut memberikan izin ke dua AWS akun, dengan pengidentifikasi 111122223333 dan 444455556666, untuk mengembalikan snapshot DB bernama `mydbsnapshot`

```
aws rds modify-db-snapshot-attribute \  
  --db-snapshot-identifier mydbsnapshot \  
  --attribute-name restore \  
  --values-to-add {"111122223333","444455556666"}
```

### Output:

```
{  
  "DBSnapshotAttributesResult": {  
    "DBSnapshotIdentifier": "mydbsnapshot",  
    "DBSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "111122223333",  
          "444455556666"  
        ]  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Berbagi Snapshot](#) di Panduan Pengguna Amazon RDS.

Contoh 2: Untuk mencegah AWS akun memulihkan snapshot DB

`modify-db-snapshot-attribute` Contoh berikut menghapus izin dari AWS akun tertentu untuk mengembalikan snapshot DB bernama `mydbsnapshot`. Saat menentukan satu akun, pengenal akun tidak dapat dikelilingi oleh tanda kutip atau tanda kurung gigi.

```
aws rds modify-db-snapshot-attribute \  
  --db-snapshot-identifier mydbsnapshot \  
  --attribute-name restore \  
  --values-to-remove 444455556666
```

### Output:

```
{  
  "DBSnapshotAttributesResult": {  
    "DBSnapshotIdentifier": "mydbsnapshot",  
    "DBSnapshotAttributes": [  

```

```

    {
      "AttributeName": "restore",
      "AttributeValues": [
        "111122223333"
      ]
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Berbagi Snapshot](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ModifyDbSnapshotAttribute](#) di Referensi AWS CLI Perintah.

## modify-db-snapshot-attributes

Contoh kode berikut menunjukkan cara menggunakan `modify-db-snapshot-attributes`.

### AWS CLI

Untuk memodifikasi atribut snapshot DB

`modify-db-snapshot-attribute` Contoh berikut mengizinkan dua pengidentifikasi AWS akun, 111122223333 dan 444455556666, untuk mengembalikan snapshot DB bernama `mydbsnapshot`

```

aws rds modify-db-snapshot-attribute \
  --db-snapshot-identifier mydbsnapshot \
  --attribute-name restore \
  --values-to-add '["111122223333","444455556666"]'

```

Output:

```

{
  "DBSnapshotAttributesResult": {
    "DBSnapshotIdentifier": "mydbsnapshot",
    "DBSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [
          "111122223333",
          "444455556666"
        ]
      }
    ]
  }
}

```

```

    ]
  }
]
}
}

```

Untuk informasi selengkapnya, lihat [Berbagi Snapshot](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ModifyDbSnapshotAttributes](#) di Referensi AWS CLI Perintah.

## modify-db-snapshot

Contoh kode berikut menunjukkan cara menggunakan `modify-db-snapshot`.

### AWS CLI

Untuk memodifikasi snapshot DB

`modify-db-snapshot` Contoh berikut memutakhirkan snapshot PostgreSQL 10.6 bernama PostgreSQL 11.7. `db5-snapshot-upg-test` Versi mesin DB baru ditampilkan setelah snapshot selesai ditingkatkan dan statusnya tersedia.

```

aws rds modify-db-snapshot \
  --db-snapshot-identifier db5-snapshot-upg-test \
  --engine-version 11.7

```

Output:

```

{
  "DBSnapshot": {
    "DBSnapshotIdentifier": "db5-snapshot-upg-test",
    "DBInstanceIdentifier": "database-5",
    "SnapshotCreateTime": "2020-03-27T20:49:17.092Z",
    "Engine": "postgres",
    "AllocatedStorage": 20,
    "Status": "upgrading",
    "Port": 5432,
    "AvailabilityZone": "us-west-2a",
    "VpcId": "vpc-2ff27557",
    "InstanceCreateTime": "2020-03-27T19:59:04.735Z",
    "MasterUsername": "postgres",
    "EngineVersion": "10.6",

```

```

    "LicenseModel": "postgresql-license",
    "SnapshotType": "manual",
    "OptionGroupName": "default:postgres-11",
    "PercentProgress": 100,
    "StorageType": "gp2",
    "Encrypted": false,
    "DBSnapshotArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-
upg-test",
    "IAMDatabaseAuthenticationEnabled": false,
    "ProcessorFeatures": [],
    "DbiResourceId": "db-GJMF75LM42IL6BTFRE4UZJ5YM4"
  }
}

```

Untuk informasi selengkapnya, lihat [Memutakhirkan Snapshot PostgreSQL DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ModifyDbSnapshot](#) di Referensi AWS CLI Perintah.

## modify-db-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `modify-db-subnet-group`.

### AWS CLI

Untuk memodifikasi grup subnet DB

`modify-db-subnet-group` Contoh berikut menambahkan subnet dengan ID `subnet-08e41f9e230222222` ke grup subnet DB bernama `mysubnetgroup` Untuk menyimpan subnet yang ada di grup subnet, sertakan ID mereka sebagai nilai dalam opsi. `--subnet-ids` Pastikan untuk memiliki subnet dengan setidaknya dua Availability Zone yang berbeda dalam grup subnet DB.

```

aws rds modify-db-subnet-group \
  --db-subnet-group-name mysubnetgroup \
  --subnet-ids
  ["subnet-0a1dc4e1a6f123456", "subnet-070dd7ecb3aaaaaaa", "subnet-00f5b198bc0abcdef", "subnet-

```

Output:

```
{
```



```
"DBSubnetGroup": {
  "DBSubnetGroupName": "mysubnetgroup",
  "DBSubnetGroupDescription": "test DB subnet group",
  "VpcId": "vpc-0f08e7610a1b2c3d4",
  "SubnetGroupStatus": "Complete",
  "Subnets": [
    {
      "SubnetIdentifier": "subnet-08e41f9e230222222",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2a"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-070dd7ecb3aaaaaaaa",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-00f5b198bc0abcdef",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2d"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-0a1dc4e1a6f123456",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      },
      "SubnetStatus": "Active"
    }
  ],
  "DBSubnetGroupArn": "arn:aws:rds:us-
west-2:534026745191:subgrp:mysubnetgroup"
}
```

Untuk informasi selengkapnya, lihat [Langkah 3: Membuat Grup Subnet DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [ModifyDbSubnetGroup](#) di Referensi AWS CLI Perintah.

## modify-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `modify-event-subscription`.

### AWS CLI

Untuk memodifikasi langganan acara

`modify-event-subscription` Contoh berikut menonaktifkan langganan acara yang ditentukan, sehingga tidak lagi menerbitkan pemberitahuan ke topik Layanan Pemberitahuan Sederhana Amazon yang ditentukan.

```
aws rds modify-event-subscription \
  --subscription-name my-instance-events \
  --no-enabled
```

Output:

```
{
  "EventSubscription": {
    "EventCategoriesList": [
      "backup",
      "recovery"
    ],
    "CustomerAwsId": "123456789012",
    "SourceType": "db-instance",
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
    "CustSubscriptionId": "my-instance-events",
    "Status": "modifying",
    "Enabled": false
  }
}
```

- Untuk detail API, lihat [ModifyEventSubscription](#) di Referensi AWS CLI Perintah.

## modify-global-cluster

Contoh kode berikut menunjukkan cara menggunakan `modify-global-cluster`.

## AWS CLI

Untuk memodifikasi cluster DB global

`modify-global-cluster` Contoh berikut memungkinkan perlindungan penghapusan untuk cluster DB global yang kompatibel dengan Aurora MySQL.

```
aws rds modify-global-cluster \  
  --global-cluster-identifier myglobalcluster \  
  --deletion-protection
```

Output:

```
{  
  "GlobalCluster": {  
    "GlobalClusterIdentifier": "myglobalcluster",  
    "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",  
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-  
cluster:myglobalcluster",  
    "Status": "available",  
    "Engine": "aurora-mysql",  
    "EngineVersion": "5.7.mysql_aurora.2.07.2",  
    "StorageEncrypted": false,  
    "DeletionProtection": true,  
    "GlobalClusterMembers": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola database global Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ModifyGlobalCluster](#) di Referensi AWS CLI Perintah.

## **promote-read-replica-db-cluster**

Contoh kode berikut menunjukkan cara menggunakan `promote-read-replica-db-cluster`.

## AWS CLI

Untuk mempromosikan replika baca cluster DB

`promote-read-replica-db-cluster` Contoh berikut mempromosikan replika baca yang ditentukan untuk menjadi cluster DB mandiri.

```
aws rds promote-read-replica-db-cluster \  
  --db-cluster-identifier mydbcluster-1
```

Output:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1b",  
      "us-east-1c"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DatabaseName": "",  
    "DBClusterIdentifier": "mydbcluster-1",  
    ...some output truncated...  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mempromosikan replika baca menjadi klaster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [PromoteReadReplicaDbCluster](#) di Referensi AWS CLI Perintah.

## **promote-read-replica**

Contoh kode berikut menunjukkan cara menggunakan `promote-read-replica`.

### AWS CLI

Untuk mempromosikan replika baca

`promote-read-replica` Contoh berikut mempromosikan replika baca yang ditentukan untuk menjadi instance DB mandiri.

```
aws rds promote-read-replica \  
  --db-instance-identifier test-instance-repl
```

Output:

```
{
```

```
"DBInstance": {
  "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance-repl",
  "StorageType": "standard",
  "ReadReplicaSourceDBInstanceIdentifier": "test-instance",
  "DBInstanceStatus": "modifying",
  ...some output truncated...
}
```

- Untuk detail API, lihat [PromoteReadReplica](#) di Referensi AWS CLI Perintah.

## **purchase-reserved-db-instance**

Contoh kode berikut menunjukkan cara menggunakan `purchase-reserved-db-instance`.

### AWS CLI

Untuk membeli penawaran instans DB yang dipesan

`purchase-reserved-db-instances-offering` Contoh berikut membeli penawaran instans DB cadangan. `reserved-db-instances-offering-id` Harus berupa ID penawaran yang valid, seperti yang dikembalikan oleh `describe-reserved-db-instances-offering` perintah.

```
aws rds purchase-reserved-db-instances -penawaran -- reserved-db-instances-offering id
438012d3-4a52-4cc7-b2e3-8dff72e0e706
```

- Untuk detail API, lihat [PurchaseReservedDbInstance](#) di Referensi AWS CLI Perintah.

## **purchase-reserved-db-instances-offerings**

Contoh kode berikut menunjukkan cara menggunakan `purchase-reserved-db-instances-offerings`.

### AWS CLI

Contoh 1: Untuk menemukan instans DB cadangan untuk dibeli

`describe-reserved-db-instances-offerings` Contoh berikut mencantumkan instance MySQL DB cadangan yang tersedia dengan kelas instans `db.t2.micro` dan durasi satu tahun. ID penawaran diperlukan untuk membeli instans DB yang dipesan.

```
aws rds describe-reserved-db-instances-offerings \  
  --product-description mysql \  
  --db-instance-class db.t2.micro \  
  --duration 1
```

Output:

```
{  
  "ReservedDBInstancesOfferings": [  
    {  
      "ReservedDBInstancesOfferingId": "8ba30be1-b9ec-447f-8f23-6114e3f4c7b4",  
      "DBInstanceClass": "db.t2.micro",  
      "Duration": 31536000,  
      "FixedPrice": 51.0,  
      "UsagePrice": 0.0,  
      "CurrencyCode": "USD",  
      "ProductDescription": "mysql",  
      "OfferingType": "Partial Upfront",  
      "MultiAZ": false,  
      "RecurringCharges": [  
        {  
          "RecurringChargeAmount": 0.006,  
          "RecurringChargeFrequency": "Hourly"  
        }  
      ]  
    },  
    ... some output truncated ...  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Instans DB Cadangan untuk Amazon RDS](#) di Panduan Pengguna Amazon RDS.

Contoh 2: Untuk membeli instans DB yang dicadangkan

`purchase-reserved-db-instances-offering` Contoh berikut menunjukkan cara membeli penawaran instans DB cadangan dari contoh sebelumnya.

```
aws rds purchase-reserved-db-instances -penawaran -- reserved-db-instances-offering id  
8ba30be1-b9ec-447f-8f23-6114e3f4c7b4
```

Output:

```
{
  "ReservedDBInstance": {
    "ReservedDBInstanceId": "ri-2020-06-29-16-54-57-670",
    "ReservedDBInstancesOfferingId": "8ba30be1-b9ec-447f-8f23-6114e3f4c7b4",
    "DBInstanceClass": "db.t2.micro",
    "StartTime": "2020-06-29T16:54:57.670Z",
    "Duration": 31536000,
    "FixedPrice": 51.0,
    "UsagePrice": 0.0,
    "CurrencyCode": "USD",
    "DBInstanceCount": 1,
    "ProductDescription": "mysql",
    "OfferingType": "Partial Upfront",
    "MultiAZ": false,
    "State": "payment-pending",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.006,
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "ReservedDBInstanceArn": "arn:aws:rds:us-west-2:123456789012:ri:ri-2020-06-29-16-54-57-670"
  }
}
```

Untuk informasi selengkapnya, lihat [Instans DB Cadangan untuk Amazon RDS](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [PurchaseReservedDbInstancesOfferings](#) di Referensi AWS CLI Perintah.

## reboot-db-instance

Contoh kode berikut menunjukkan cara menggunakan `reboot-db-instance`.

### AWS CLI

Untuk me-reboot instance DB

`reboot-db-instance` Contoh berikut memulai reboot dari instance DB yang ditentukan.

```
aws rds reboot-db-instance \
```

```
--db-instance-identifier test-mysql-instance
```

Output:

```
{
  "DBInstance": {
    "DBInstanceIdentifier": "test-mysql-instance",
    "DBInstanceClass": "db.t3.micro",
    "Engine": "mysql",
    "DBInstanceStatus": "rebooting",
    "MasterUsername": "admin",
    "Endpoint": {
      "Address": "test-mysql-instance.#####.us-
west-2.rds.amazonaws.com",
      "Port": 3306,
      "HostedZoneId": "Z1PVIF0EXAMPLE"
    },
    ... output omitted...
  }
}
```

Untuk informasi selengkapnya, lihat [Mem-boot Ulang Instans DB](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [RebootDBInstance](#) di Referensi Perintah AWS CLI .

## reboot-db-shard-group

Contoh kode berikut menunjukkan cara menggunakan `reboot-db-shard-group`.

AWS CLI

Contoh 1: Untuk me-reboot grup shard DB

`reboot-db-shard-group` Contoh berikut reboot grup shard DB.

```
aws rds reboot-db-shard-group \
  --db-shard-group-identifier my-db-shard-group
```

Output:



```
{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",
      "MaxACU": 1000.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": false,
      "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mem-boot ulang cluster Amazon Aurora DB atau instans DB Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

Contoh 2: Untuk menggambarkan grup pecahan DB Anda

`describe-db-shard-groups` Contoh berikut mengambil rincian grup shard DB Anda setelah Anda menjalankan perintah `reboot-db-shard-group` Grup pecahan DB sekarang `my-db-shard-group` sedang reboot.

```
aws rds describe-db-shard-groups
```

Output:

```
{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
      "DBShardGroupIdentifier": "limitless-test-shard-grp",
      "DBClusterIdentifier": "limitless-test-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": true,
      "Endpoint": "limitless-test-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
    },
  ],
}
```

```

    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",
      "MaxACU": 1000.0,
      "ComputeRedundancy": 0,
      "Status": "rebooting",
      "PubliclyAccessible": false,
      "Endpoint": "my-sv2-cluster.limitless-cekyexample.us-east-2.rds.amazonaws.com"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Mem-boot ulang cluster Amazon Aurora DB atau instans DB Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [RebootDbShardGroup](#) di Referensi AWS CLI Perintah.

## register-db-proxy-targets

Contoh kode berikut menunjukkan cara menggunakan `register-db-proxy-targets`.

### AWS CLI

Untuk mendaftarkan proxy DB dengan database

`register-db-proxy-targets` Contoh berikut menciptakan hubungan antara database dan proxy.

```

aws rds register-db-proxy-targets \
  --db-proxy-name proxyExample \
  --db-cluster-identifiers database-5

```

Output:

```

{
  "DBProxyTargets": [
    {
      "RdsResourceId": "database-5",
      "Port": 3306,
      "Type": "TRACKED_CLUSTER",
      "TargetHealth": {

```

```

        "State": "REGISTERING"
      }
    },
    {
      "Endpoint": "database-5instance-1.ab0cd1efghij.us-
east-1.rds.amazonaws.com",
      "RdsResourceId": "database-5",
      "Port": 3306,
      "Type": "RDS_INSTANCE",
      "TargetHealth": {
        "State": "REGISTERING"
      }
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Membuat proxy RDS](#) di Panduan Pengguna Amazon RDS dan [Membuat proxy RDS](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [RegisterDbProxyTargets](#) di Referensi AWS CLI Perintah.

## remove-from-global-cluster

Contoh kode berikut menunjukkan cara menggunakan `remove-from-global-cluster`.

### AWS CLI

Untuk melepaskan cluster sekunder Aurora dari cluster database global Aurora

`remove-from-global-cluster` Contoh berikut melepaskan cluster sekunder Aurora dari cluster database global Aurora. Cluster berubah dari read-only menjadi cluster mandiri dengan kemampuan baca-tulis.

```

aws rds remove-from-global-cluster \
  --region us-west-2 \
  --global-cluster-identifier myglobalcluster \
  --db-cluster-identifier arn:aws:rds:us-west-2:123456789012:cluster:DB-1

```

Output:

```

{
  "GlobalCluster": {

```

```

    "GlobalClusterIdentifier": "myglobalcluster",
    "GlobalClusterResourceId": "cluster-abc123def456gh",
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
    "Status": "available",
    "Engine": "aurora-postgresql",
    "EngineVersion": "10.11",
    "StorageEncrypted": true,
    "DeletionProtection": false,
    "GlobalClusterMembers": [
      {
        "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:js-
global-cluster",
        "Readers": [
          "arn:aws:rds:us-west-2:123456789012:cluster:DB-1"
        ],
        "IsWriter": true
      },
      {
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:DB-1",
        "Readers": [],
        "IsWriter": false,
        "GlobalWriteForwardingStatus": "disabled"
      }
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Menghapus klaster dari database global Amazon Aurora di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [RemoveFromGlobalCluster](#) di Referensi AWS CLI Perintah.

## remove-option-from-option-group

Contoh kode berikut menunjukkan cara menggunakan `remove-option-from-option-group`.

### AWS CLI

Untuk menghapus opsi dari grup opsi

`remove-option-from-option-group` Contoh berikut menghapus OEM opsi dari `optiongroup`.

```
aws rds remove-option-from-option-group \  
  --option-group-name myoptiongroup \  
  --options OEM \  
  --apply-immediately
```

Output:

```
{  
  "OptionGroup": {  
    "OptionGroupName": "myoptiongroup",  
    "OptionGroupDescription": "Test",  
    "EngineName": "oracle-ee",  
    "MajorEngineVersion": "19",  
    "Options": [],  
    "AllowsVpcAndNonVpcInstanceMemberships": true,  
    "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:myoptiongroup"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Opsi dari Grup Opsi](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [RemoveOptionFromOptionGroup](#) di Referensi AWS CLI Perintah.

## **remove-role-from-db-cluster**

Contoh kode berikut menunjukkan cara menggunakan `remove-role-from-db-cluster`.

### AWS CLI

Untuk memisahkan peran AWS Identity and Access Management (IAM) dari cluster DB

`remove-role-from-db-cluster` Contoh berikut menghapus peran dari cluster DB.

```
aws rds remove-role-from-db-cluster \  
  --db-cluster-identifier mydbcluster \  
  --role-arn arn:aws:iam::123456789012:role/RDSLoadFromS3
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan peran IAM dengan kluster DB MySQL Amazon Aurora](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [RemoveRoleFromDbCluster](#) di Referensi AWS CLI Perintah.

## **remove-role-from-db-instance**

Contoh kode berikut menunjukkan cara menggunakan `remove-role-from-db-instance`.

### AWS CLI

Memisahkan peran AWS Identity and Access Management (IAM) and Access Management (IAM) dari instans DB

`remove-role-from-db-instance` Contoh berikut menghapus peran bernama `rds-s3-integration-role` dari instance Oracle DB bernama `test-instance`.

```
aws rds remove-role-from-db-instance \  
  --db-instance-identifier test-instance \  
  --feature-name S3_INTEGRATION \  
  --role-arn arn:aws:iam::111122223333:role/rds-s3-integration-role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan Integrasi Server SQL RDS dengan S3 di Panduan Pengguna](#) Amazon RDS.

- Untuk detail API, lihat [RemoveRoleFromDbInstance](#) di Referensi AWS CLI Perintah.

## **remove-source-identifier-from-subscription**

Contoh kode berikut menunjukkan cara menggunakan `remove-source-identifier-from-subscription`.

### AWS CLI

Untuk menghapus pengenalan sumber dari langganan

`remove-source-identifier` Contoh berikut menghapus pengenalan sumber yang ditentukan dari langganan yang ada.

```
aws rds remove-source-identifier-from-subscription \  
  --subscription-name my-instance-events \  
  --source-identifier my-instance-events
```

```
--source-identifier test-instance-repl
```

Output:

```
{
  "EventSubscription": {
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
    "EventCategoriesList": [
      "backup",
      "recovery"
    ],
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
    "Status": "modifying",
    "CustSubscriptionId": "my-instance-events",
    "CustomerAwsId": "123456789012",
    "SourceIdsList": [
      "test-instance"
    ],
    "SourceType": "db-instance",
    "Enabled": false
  }
}
```

- Untuk detail API, lihat [RemoveSourceIdentifierFromSubscription](#) di Referensi AWS CLI Perintah.

## **remove-tags-from-resource**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya

`remove-tags-from-resource` Contoh berikut menghapus tag dari sumber daya.

```
aws rds remove-tags-from-resource \
  --resource-name arn:aws:rds:us-east-1:123456789012:db:mydbinstance \
  --tag-keys Name Environment
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Amazon RDS](#) di Panduan Pengguna Amazon RDS dan Menandai sumber daya [Amazon RDS di](#) Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [RemoveTagsFromResource](#) di Referensi AWS CLI Perintah.

## reset-db-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `reset-db-cluster-parameter-group`.

### AWS CLI

Contoh 1: Untuk mengatur ulang semua parameter ke nilai defaultnya

`reset-db-cluster-parameter-group` Contoh berikut me-reset semua nilai parameter dalam kelompok parameter cluster DB yang dibuat pelanggan ke nilai default mereka.

```
aws rds reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name mydbclpg \  
  --reset-all-parameters
```

Output:

```
{  
  "DBClusterParameterGroupName": "mydbclpg"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup parameter DB dan grup parameter cluster DB](#) di Panduan Pengguna Amazon Aurora.

Contoh 2: Untuk mengatur ulang parameter tertentu ke nilai defaultnya

`reset-db-cluster-parameter-group` Contoh berikut me-reset nilai parameter untuk parameter tertentu ke nilai default mereka dalam kelompok parameter cluster DB yang dibuat pelanggan.

```
aws rds reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name mydbclpgy \  
  --parameters "ParameterName=max_connections,ApplyMethod=immediate" \  
  "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Output:



```
{
  "DBClusterParameterGroupName": "mydbclpg"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup parameter DB dan grup parameter cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ResetDbClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## reset-db-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `reset-db-parameter-group`.

### AWS CLI

Contoh 1: Untuk mengatur ulang semua parameter ke nilai defaultnya

`reset-db-parameter-group` Contoh berikut me-reset semua nilai parameter dalam grup parameter DB yang dibuat pelanggan ke nilai default mereka.

```
aws rds reset-db-parameter-group \
  --db-parameter-group-name mypg \
  --reset-all-parameters
```

Output:

```
{
  "DBParameterGroupName": "mypg"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup parameter DB](#) di Panduan Pengguna Amazon RDS dan [Bekerja dengan grup parameter DB dan grup parameter cluster DB](#) di Panduan Pengguna Amazon Aurora.

Contoh 2: Untuk mengatur ulang parameter tertentu ke nilai defaultnya

`reset-db-parameter-group` Contoh berikut me-reset nilai parameter untuk parameter tertentu ke nilai default mereka dalam kelompok parameter DB yang dibuat pelanggan.

```
aws rds reset-db-parameter-group \
  --db-parameter-group-name mypg \
```

```
--parameters "ParameterName=max_connections,ApplyMethod=immediate" \  
             "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Output:

```
{  
  "DBParameterGroupName": "mypg"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan grup parameter DB](#) di Panduan Pengguna Amazon RDS dan [Bekerja dengan grup parameter DB dan grup parameter cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [ResetDbParameterGroup](#) di Referensi AWS CLI Perintah.

### restore-db-cluster-from-s3

Contoh kode berikut menunjukkan cara menggunakan `restore-db-cluster-from-s3`.

#### AWS CLI

Untuk memulihkan cluster Amazon Aurora DB dari Amazon S3

`restore-db-cluster-from-s3` Contoh berikut mengembalikan kluster DB yang kompatibel dengan Amazon Aurora MySQL versi 5.7 dari file cadangan MySQL 5.7 DB di Amazon S3.

```
aws rds restore-db-cluster-from-s3 \  
  --db-cluster-identifier cluster-s3-restore \  
  --engine aurora-mysql \  
  --master-username admin \  
  --master-user-password mypassword \  
  --s3-bucket-name mybucket \  
  --s3-prefix test-backup \  
  --s3-ingestion-role-arn arn:aws:iam::123456789012:role/service-role/TestBackup \  
  --source-engine mysql \  
  --source-engine-version 5.7.28
```

Output:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
  }  
}
```

```
"AvailabilityZones": [
  "us-west-2c",
  "us-west-2a",
  "us-west-2b"
],
"BackupRetentionPeriod": 1,
"DBClusterIdentifier": "cluster-s3-restore",
"DBClusterParameterGroup": "default.aurora-mysql5.7",
"DBSubnetGroup": "default",
"Status": "creating",
"Endpoint": "cluster-s3-restore.cluster-co3xyzabc123.us-
west-2.rds.amazonaws.com",
"ReaderEndpoint": "cluster-s3-restore.cluster-ro-co3xyzabc123.us-
west-2.rds.amazonaws.com",
"MultiAZ": false,
"Engine": "aurora-mysql",
"EngineVersion": "5.7.12",
"Port": 3306,
"MasterUsername": "admin",
"PreferredBackupWindow": "11:15-11:45",
"PreferredMaintenanceWindow": "thu:12:19-thu:12:49",
"ReadReplicaIdentifiers": [],
"DBClusterMembers": [],
"VpcSecurityGroups": [
  {
    "VpcSecurityGroupId": "sg-#####",
    "Status": "active"
  }
],
"HostedZoneId": "Z1PVIF0EXAMPLE",
"StorageEncrypted": false,
"DbClusterResourceId": "cluster-SU5THYQQH0WCXZZDGXREXAMPLE",
"DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:cluster-s3-
restore",
"AssociatedRoles": [],
"IAMDatabaseAuthenticationEnabled": false,
"ClusterCreateTime": "2020-07-27T14:22:08.095Z",
"EngineMode": "provisioned",
"DeletionProtection": false,
"HttpEndpointEnabled": false,
"CopyTagsToSnapshot": false,
"CrossAccountClone": false,
"DomainMemberships": []
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Memigrasi Data dari MySQL dengan Menggunakan Bucket Amazon S3 di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [RestoreDbClusterFromS3](#) di Referensi AWS CLI Perintah.

## restore-db-cluster-from-snapshot

Contoh kode berikut menunjukkan cara menggunakan `restore-db-cluster-from-snapshot`.

### AWS CLI

Untuk memulihkan cluster DB dari snapshot

Berikut ini `restore-db-cluster-from-snapshot` mengembalikan klaster Aurora PostgreSQL DB yang kompatibel dengan PostgreSQL versi 10.7 dari snapshot cluster DB bernama `test-instance-snapshot`

```
aws rds restore-db-cluster-from-snapshot \
  --db-cluster-identifier newdbcluster \
  --snapshot-identifier test-instance-snapshot \
  --engine aurora-postgresql \
  --engine-version 10.7
```

Output:

```
{
  "DBCluster": {
    "AllocatedStorage": 1,
    "AvailabilityZones": [
      "us-west-2c",
      "us-west-2a",
      "us-west-2b"
    ],
    "BackupRetentionPeriod": 7,
    "DatabaseName": "",
    "DBClusterIdentifier": "newdbcluster",
    "DBClusterParameterGroup": "default.aurora-postgresql10",
    "DBSubnetGroup": "default",
    "Status": "creating",
```

```

    "Endpoint": "newdbcluster.cluster-#####.us-west-2.rds.amazonaws.com",
    "ReaderEndpoint": "newdbcluster.cluster-ro-#####.us-
west-2.rds.amazonaws.com",
    "MultiAZ": false,
    "Engine": "aurora-postgresql",
    "EngineVersion": "10.7",
    "Port": 5432,
    "MasterUsername": "postgres",
    "PreferredBackupWindow": "09:33-10:03",
    "PreferredMaintenanceWindow": "sun:12:22-sun:12:52",
    "ReadReplicaIdentifiers": [],
    "DBClusterMembers": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-#####",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z1PVIF0EXAMPLE",
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/287364e4-33e3-4755-a3b0-
a1b2c3d4e5f6",
    "DbClusterResourceId": "cluster-5DSB5IFQDDUVAWOUWM1EXAMPLE",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:newdbcluster",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "ClusterCreateTime": "2020-06-05T15:06:58.634Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": false,
    "CrossAccountClone": false,
    "DomainMemberships": []
  }
}

```

Untuk informasi selengkapnya, lihat [Memulihkan dari Snapshot Cluster DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [RestoreDbClusterFromSnapshot](#) di Referensi AWS CLI Perintah.

## restore-db-cluster-to-point-in-time

Contoh kode berikut menunjukkan cara menggunakan `restore-db-cluster-to-point-in-time`.

### AWS CLI

Untuk mengembalikan cluster DB ke waktu tertentu

`restore-db-cluster-to-point-in-time` Contoh berikut mengembalikan cluster DB bernama `database-4` ke waktu terbaru yang mungkin. Menggunakan `copy-on-write` as the `restore-type` mengembalikan cluster DB baru sebagai tiruan dari cluster DB sumber.

```
aws rds restore-db-cluster-to-point-in-time \  
  --source-db-cluster-identifier database-4 \  
  --db-cluster-identifier sample-cluster-clone \  
  --restore-type copy-on-write \  
  --use-latest-restorable-time
```

### Output:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-west-2c",  
      "us-west-2a",  
      "us-west-2b"  
    ],  
    "BackupRetentionPeriod": 7,  
    "DatabaseName": "",  
    "DBClusterIdentifier": "sample-cluster-clone",  
    "DBClusterParameterGroup": "default.aurora-postgresql10",  
    "DBSubnetGroup": "default",  
    "Status": "creating",  
    "Endpoint": "sample-cluster-clone.cluster-#####.us-west-2.rds.amazonaws.com",  
    "ReaderEndpoint": "sample-cluster-clone.cluster-ro-#####.us-west-2.rds.amazonaws.com",  
    "MultiAZ": false,  
    "Engine": "aurora-postgresql",  
    "EngineVersion": "10.7",  
    "Port": 5432,
```

```

    "MasterUsername": "postgres",
    "PreferredBackupWindow": "09:33-10:03",
    "PreferredMaintenanceWindow": "sun:12:22-sun:12:52",
    "ReadReplicaIdentifiers": [],
    "DBClusterMembers": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-#####",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z1PVIIF0EXAMPLE",
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/287364e4-33e3-4755-a3b0-
a1b2c3d4e5f6",
    "DbClusterResourceId": "cluster-BIZ77GDSA2XBSTNPFW1EXAMPLE",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster-
clone",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "CloneGroupId": "8d19331a-099a-45a4-b4aa-11aa22bb33cc44dd",
    "ClusterCreateTime": "2020-03-10T19:57:38.967Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": false,
    "CrossAccountClone": false
  }
}

```

Untuk informasi selengkapnya, lihat [Memulihkan Cluster DB ke Waktu Tertentu](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [RestoreDbClusterToPointInTime](#) di Referensi AWS CLI Perintah.

## restore-db-instance-from-db-snapshot

Contoh kode berikut menunjukkan cara menggunakan `restore-db-instance-from-db-snapshot`.

### AWS CLI

Untuk mengembalikan instans DB dari snapshot DB

`restore-db-instance-from-db-snapshot` Contoh berikut membuat instance DB baru bernama `db7-new-instance` dengan kelas instance `db.t3.small` DB dari snapshot DB yang ditentukan. Instance DB sumber dari mana snapshot diambil menggunakan kelas instans DB yang tidak digunakan lagi, jadi Anda tidak dapat memutakhirkannya.

```
aws rds restore-db-instance-from-db-snapshot \  
  --db-instance-identifier db7-new-instance \  
  --db-snapshot-identifier db7-test-snapshot \  
  --db-instance-class db.t3.small
```

Output:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "db7-new-instance",  
    "DBInstanceClass": "db.t3.small",  
    "Engine": "mysql",  
    "DBInstanceStatus": "creating",  
  
    ...output omitted...  
  
    "PreferredMaintenanceWindow": "mon:07:37-mon:08:07",  
    "PendingModifiedValues": {},  
    "MultiAZ": false,  
    "EngineVersion": "5.7.22",  
    "AutoMinorVersionUpgrade": true,  
    "ReadReplicaDBInstanceIdentifiers": [],  
    "LicenseModel": "general-public-license",  
  
    ...output omitted...  
  
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:db7-new-instance",  
    "IAMDatabaseAuthenticationEnabled": false,  
    "PerformanceInsightsEnabled": false,  
    "DeletionProtection": false,  
    "AssociatedRoles": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memulihkan dari Snapshot DB](#) di Panduan Pengguna Amazon RDS.



- Untuk detail API, lihat [RestoreDbInstanceFromDbSnapshot](#) di Referensi AWS CLI Perintah.

### **restore-db-instance-from-s3**

Contoh kode berikut menunjukkan cara menggunakan `restore-db-instance-from-s3`.

#### AWS CLI

Untuk memulihkan instans DB dari cadangan di Amazon S3

`restore-db-instance-from-s3` Contoh berikut membuat instance DB baru bernama `restored-test-instance` dari cadangan yang ada di bucket `my-backups` S3.

```
aws rds restore-db-instance-from-s3 \  
  --db-instance-identifier restored-test-instance \  
  --allocated-storage 250 --db-instance-class db.m4.large --engine mysql \  
  --master-username master --master-user-password secret99 \  
  --s3-bucket-name my-backups --s3-ingestion-role-arn  
arn:aws:iam::123456789012:role/my-role \  
  --source-engine mysql --source-engine-version 5.6.27
```

- Untuk detail API, lihat [RestoreDbInstanceFromS3](#) di Referensi AWS CLI Perintah.

### **restore-db-instance-to-point-in-time**

Contoh kode berikut menunjukkan cara menggunakan `restore-db-instance-to-point-in-time`.

#### AWS CLI

Contoh 1: Untuk mengembalikan instans DB ke titik waktu

`restore-db-instance-to-point-in-time` Contoh berikut mengembalikan `test-instance` ke instance DB baru bernama `restored-test-instance`, pada waktu yang ditentukan.

```
aws rds restore-db-instance-to-point-in-time \  
  --source-db-instance-identifier test-instance \  
  --target-db-instance restored-test-instance \  
  --restore-time 2018-07-30T23:45:00.000Z
```

**Output:**

```
{
  "DBInstance": {
    "AllocatedStorage": 20,
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:restored-test-
instance",
    "DBInstanceStatus": "creating",
    "DBInstanceIdentifier": "restored-test-instance",
    ...some output omitted...
  }
}
```

Untuk informasi selengkapnya, lihat [Memulihkan instans DB ke waktu yang ditentukan](#) di Panduan Pengguna Amazon RDS.

Contoh 2: Untuk mengembalikan instans DB ke waktu tertentu dari cadangan yang direplikasi  
`restore-db-instance-to-point-in-time` Contoh berikut mengembalikan instance Oracle DB ke waktu yang ditentukan dari cadangan otomatis yang direplikasi.

```
aws rds restore-db-instance-to-point-in-time \
  --source-db-instance-automated-backups-arn "arn:aws:rds:us-
west-2:123456789012:auto-backup:ab-jkib2gfg5rv7replzadausbrktni2bn4example" \
  --target-db-instance-identifier myorclinstance-from-replicated-backup \
  --restore-time 2020-12-08T18:45:00.000Z
```

**Output:**

```
{
  "DBInstance": {
    "DBInstanceIdentifier": "myorclinstance-from-replicated-backup",
    "DBInstanceClass": "db.t3.micro",
    "Engine": "oracle-se2",
    "DBInstanceStatus": "creating",
    "MasterUsername": "admin",
    "DBName": "ORCL",
    "AllocatedStorage": 20,
    "PreferredBackupWindow": "07:45-08:15",
    "BackupRetentionPeriod": 14,
    ... some output omitted ...
    "DbiResourceId": "db-KGLXG75BGVIWKQT7NQ4EXAMPLE",
    "CACertificateIdentifier": "rds-ca-2019",
  }
}
```

```

    "DomainMemberships": [],
    "CopyTagsToSnapshot": false,
    "MonitoringInterval": 0,
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:myorclinstance-from-
replicated-backup",
    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": [],
    "TagList": []
  }
}

```

Untuk informasi selengkapnya, lihat [Memulihkan ke waktu tertentu dari cadangan yang direplikasi](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [RestoreDbInstanceToPointInTime](#) di Referensi AWS CLI Perintah.

## start-activity-stream

Contoh kode berikut menunjukkan cara menggunakan `start-activity-stream`.

### AWS CLI

Untuk memulai aliran aktivitas database

`start-activity-stream` Contoh berikut memulai aliran aktivitas asinkron untuk memantau cluster Aurora bernama `my-pg-cluster`

```

aws rds start-activity-stream \
  --region us-east-1 \
  --mode async \
  --kms-key-id arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-
h123-456i789jk0l1 \
  --resource-arn arn:aws:rds:us-east-1:1234567890123:cluster:my-pg-cluster \
  --apply-immediately

```

Output:

```

{
  "KmsKeyId": "arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-
h123-456i789jk0l1",
  "KinesisStreamName": "aws-rds-das-cluster-0ABCDEFGH1JKL2M3NOPQ3R4S",

```

```
"Status": "starting",
"Mode": "async",
"ApplyImmediately": true
}
```

Untuk informasi selengkapnya, lihat [Memulai aliran aktivitas database](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [StartActivityStream](#) di Referensi AWS CLI Perintah.

## start-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `start-db-cluster`.

### AWS CLI

Untuk memulai cluster DB

`start-db-cluster` Contoh berikut memulai cluster DB dan instance DB-nya.

```
aws rds start-db-cluster \
  --db-cluster-identifier mydbcluster
```

Output:

```
{
  "DBCluster": {
    "AllocatedStorage": 1,
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1e",
      "us-east-1b"
    ],
    "BackupRetentionPeriod": 1,
    "DatabaseName": "mydb",
    "DBClusterIdentifier": "mydbcluster",
    ...some output truncated...
  }
}
```

Untuk informasi selengkapnya, lihat [Menghentikan dan memulai klaster Amazon Aurora DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [StartDbCluster](#) di Referensi AWS CLI Perintah.

## start-db-instance-automated-backups-replication

Contoh kode berikut menunjukkan cara menggunakan `start-db-instance-automated-backups-replication`.

### AWS CLI

Untuk mengaktifkan pencadangan otomatis lintas wilayah

`start-db-instance-automated-backups-replication` Contoh berikut mereplikasi backup otomatis dari instans DB di Wilayah AS Timur (Virginia N.) ke AS Barat (Oregon). Periode retensi cadangan adalah 14 hari.

```
aws rds start-db-instance-automated-backups-replication \  
  --region us-west-2 \  
  --source-db-instance-arn "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db" \  
  --backup-retention-period 14
```

### Output:

```
{  
  "DBInstanceAutomatedBackup": {  
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",  
    "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",  
    "Region": "us-east-1",  
    "DBInstanceIdentifier": "new-orcl-db",  
    "RestoreWindow": {},  
    "AllocatedStorage": 20,  
    "Status": "pending",  
    "Port": 1521,  
    "InstanceCreateTime": "2020-12-04T15:28:31Z",  
    "MasterUsername": "admin",  
    "Engine": "oracle-se2",  
    "EngineVersion": "12.1.0.2.v21",  
    "LicenseModel": "bring-your-own-license",  
    "OptionGroupName": "default:oracle-se2-12-1",  
    "Encrypted": false,  
    "StorageType": "gp2",  
    "IAMDatabaseAuthenticationEnabled": false,  
    "BackupRetentionPeriod": 14,
```

```

    "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengaktifkan pencadangan otomatis lintas wilayah di Panduan Pengguna Amazon RDS](#).

- Untuk detail API, lihat [StartDbInstanceAutomatedBackupsReplication](#) di Referensi AWS CLI Perintah.

## start-db-instance

Contoh kode berikut menunjukkan cara menggunakan `start-db-instance`.

### AWS CLI

Untuk memulai instance DB

`start-db-instance` Contoh berikut memulai instance DB tertentu.

```

aws rds start-db-instance \
  --db-instance-identifier test-instance

```

Output:

```

{
  "DBInstance": {
    "DBInstanceStatus": "starting",
    ...some output truncated...
  }
}

```

- Untuk detail API, lihat [StartDbInstance](#) di Referensi AWS CLI Perintah.

## start-export-task

Contoh kode berikut menunjukkan cara menggunakan `start-export-task`.

### AWS CLI

Untuk mengekspor snapshot ke Amazon S3

`start-export-task` Contoh berikut mengekspor snapshot DB bernama `db5-snapshot-test` ke bucket Amazon S3 bernama `mybucket`

```
aws rds start-export-task \  
  --export-task-identifier my-s3-export \  
  --source-arn arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-test \  
  --s3-bucket-name mybucket \  
  --iam-role-arn arn:aws:iam::123456789012:role/service-role/ExportRole \  
  --kms-key-id arn:aws:kms:us-west-2:123456789012:key/abcd0000-7fca-4128-82f2-  
  aabbccddeeff
```

Output:

```
{  
  "ExportTaskIdentifier": "my-s3-export",  
  "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-test",  
  "SnapshotTime": "2020-03-27T20:48:42.023Z",  
  "S3Bucket": "mybucket",  
  "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/abcd0000-7fca-4128-82f2-  
  aabbccddeeff",  
  "Status": "STARTING",  
  "PercentProgress": 0,  
  "TotalExtractedDataInGB": 0  
}
```

Untuk informasi selengkapnya, lihat [Mengekspor Snapshot ke Bucket Amazon S3 di Panduan Pengguna Amazon RDS](#).

- Untuk detail API, lihat [StartExportTask](#) di Referensi AWS CLI Perintah.

## stop-activity-stream

Contoh kode berikut menunjukkan cara menggunakan `stop-activity-stream`.

### AWS CLI

Untuk menghentikan aliran aktivitas database

`stop-activity-stream` Contoh berikut menghentikan aliran aktivitas di cluster Aurora bernama `my-pg-cluster`

```
aws rds stop-activity-stream \  
  --region us-east-1 \  
  --resource-arn arn:aws:rds:us-east-1:1234567890123:cluster:my-pg-cluster \  
  --apply-immediately
```

Output:

```
{  
  "KmsKeyId": "arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-  
h123-456i789jk011",  
  "KinesisStreamName": "aws-rds-das-cluster-0ABCDEFGH11JKLM2NOPQ3R4S",  
  "Status": "stopping"  
}
```

Untuk informasi selengkapnya, lihat [Menghentikan aliran aktivitas](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [StopActivityStream](#) di Referensi AWS CLI Perintah.

## stop-db-cluster

Contoh kode berikut menunjukkan cara menggunakan `stop-db-cluster`.

### AWS CLI

Untuk menghentikan cluster DB

`stop-db-cluster` Contoh berikut menghentikan cluster DB dan instance DB-nya.

```
aws rds stop-db-cluster \  
  --db-cluster-identifier mydbcluster
```

Output:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1e",  
      "us-east-1b"  
    ],  
  },  
}
```



```

    "BackupRetentionPeriod": 1,
    "DatabaseName": "mydb",
    "DBClusterIdentifier": "mydbcluster",
    ...some output truncated...
  }
}

```

Untuk informasi selengkapnya, lihat [Menghentikan dan memulai klaster Amazon Aurora DB](#) di Panduan Pengguna Amazon Aurora.

- Untuk detail API, lihat [StopDbCluster](#) di Referensi AWS CLI Perintah.

## stop-db-instance-automated-backups-replication

Contoh kode berikut menunjukkan cara menggunakan `stop-db-instance-automated-backups-replication`.

### AWS CLI

Untuk berhenti mereplikasi cadangan otomatis

Berikut ini `stop-db-instance-automated-backups-replication` mengakhiri replikasi cadangan otomatis ke Wilayah Barat AS (Oregon). Pencadangan yang direplikasi dipertahankan sesuai dengan periode retensi cadangan yang ditetapkan.

```

aws rds stop-db-instance-automated-backups-replication \
  --region us-west-2 \
  --source-db-instance-arn "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db"

```

Output:

```

{
  "DBInstanceAutomatedBackup": {
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
    "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
    "Region": "us-east-1",
    "DBInstanceIdentifier": "new-orcl-db",
    "RestoreWindow": {
      "EarliestTime": "2020-12-04T23:13:21.030Z",
      "LatestTime": "2020-12-07T19:59:57Z"
    },
    "AllocatedStorage": 20,
  }
}

```

```
"Status": "replicating",
"Port": 1521,
"InstanceCreateTime": "2020-12-04T15:28:31Z",
"MasterUsername": "admin",
"Engine": "oracle-se2",
"EngineVersion": "12.1.0.2.v21",
"LicenseModel": "bring-your-own-license",
"OptionGroupName": "default:oracle-se2-12-1",
"Encrypted": false,
"StorageType": "gp2",
"IAMDatabaseAuthenticationEnabled": false,
"BackupRetentionPeriod": 7,
"DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
}
```

Untuk informasi selengkapnya, lihat [Menghentikan replikasi pencadangan otomatis](#) di Panduan Pengguna Amazon RDS.

- Untuk detail API, lihat [StopDbInstanceAutomatedBackupsReplication](#) di Referensi AWS CLI Perintah.

## stop-db-instance

Contoh kode berikut menunjukkan cara menggunakan `stop-db-instance`.

### AWS CLI

Untuk menghentikan instans DB

`stop-db-instance` Contoh berikut menghentikan instance DB yang ditentukan.

```
aws rds stop-db-instance \
  --db-instance-identifier test-instance
```

Output:

```
{
  "DBInstance": {
    "DBInstanceStatus": "stopping",
    ...some output truncated...
  }
}
```

```
}
```

- Untuk detail API, lihat [StopDbInstance](#) di Referensi AWS CLI Perintah.

## switchover-blue-green-deployment

Contoh kode berikut menunjukkan cara menggunakan `switchover-blue-green-deployment`.

### AWS CLI

Contoh 1: Untuk mengganti penerapan biru/hijau untuk instance RDS DB

`switchover-blue-green-deployment` Contoh berikut mempromosikan lingkungan hijau yang ditentukan sebagai lingkungan produksi baru.

```
aws rds switchover-blue-green-deployment \  
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake \  
  --switchover-timeout 300
```

Output:

```
{  
  "BlueGreenDeployment": {  
    "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",  
    "BlueGreenDeploymentName": "bgd-cli-test-instance",  
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",  
    "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-  
blhile",  
    "SwitchoverDetails": [  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance-green-blhile",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance-replica-1",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance-replica-1-green-k5fv7u",  
        "Status": "AVAILABLE"  
      }  
    ]  
  }  
}
```

```

    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ggsh8m",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-o2vwm0",
      "Status": "AVAILABLE"
    }
  ],
  "Tasks": [
    {
      "Name": "CREATING_READ_REPLICA_OF_SOURCE",
      "Status": "COMPLETED"
    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "COMPLETED"
    },
    {
      "Name": "CONFIGURE_BACKUPS",
      "Status": "COMPLETED"
    },
    {
      "Name": "CREATING_TOPOLOGY_OF_SOURCE",
      "Status": "COMPLETED"
    }
  ],
  "Status": "SWITCHOVER_IN_PROGRESS",
  "CreateTime": "2022-02-25T22:33:22.225000+00:00"
}

```

Untuk informasi selengkapnya, lihat [Mengganti penerapan biru/hijau di Panduan Pengguna Amazon RDS](#).

Contoh 2: Untuk mempromosikan penyebaran biru/hijau untuk cluster DB MySQL Aurora

`switchover-blue-green-deployment` Contoh berikut mempromosikan lingkungan hijau yang ditentukan sebagai lingkungan produksi baru.

```
aws rds switchover-blue-green-deployment \  
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake \  
  --switchover-timeout 300
```

Output:

```
{  
  "BlueGreenDeployment": {  
    "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",  
    "BlueGreenDeploymentName": "my-blue-green-deployment",  
    "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-  
cluster",  
    "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-  
cluster-green-3ud8z6",  
    "SwitchoverDetails": [  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-  
aurora-mysql-cluster",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-  
aurora-mysql-cluster-green-3ud8z6",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-  
mysql-cluster-1",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-  
mysql-cluster-1-green-bvxc73",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-  
mysql-cluster-2",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-  
mysql-cluster-2-green-7wc4ie",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-  
mysql-cluster-3",
```

```
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3-green-p4xxkz",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-np1ik1",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-miszlf",
        "Status": "AVAILABLE"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "COMPLETED"
    }
],
"Status": "SWITCHOVER_IN_PROGRESS",
"CreateTime": "2022-02-25T22:38:49.522000+00:00"
}
}
```

Untuk informasi selengkapnya, lihat [Mengganti penerapan biru/hijau di Panduan Pengguna Amazon Aurora](#).

- Untuk detail API, lihat [SwitchoverBlueGreenDeployment](#) di Referensi AWS CLI Perintah.

## Contoh Layanan Data Amazon RDS menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan Layanan Data Amazon RDS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-execute-statement**

Contoh kode berikut menunjukkan cara menggunakan `batch-execute-statement`.

AWS CLI

Untuk mengeksekusi pernyataan SQL batch

`batch-execute-statement` Contoh berikut mengeksekusi pernyataan SQL batch atas array data dengan set parameter.

```
aws rds-data batch-execute-statement \  
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \  
  --database "mydb" \  
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \  
  --statement "SELECT * FROM mytable"
```

```
--sql "insert into mytable values (:id, :val)" \
--parameter-sets "[[{"name": \"id\", \"value\": {\"longValue\": 1}}, {\"name\": \"val\", \"value\": {\"stringValue\": \"ValueOne\"}}, [ {\"name\": \"id\", \"value\": {\"longValue\": 2}}, {\"name\": \"val\", \"value\": {\"stringValue\": \"ValueTwo\"}}], [ {\"name\": \"id\", \"value\": {\"longValue\": 3}}, {\"name\": \"val\", \"value\": {\"stringValue\": \"ValueThree\"}}]]]"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan API Data untuk Aurora Tanpa Server di Panduan Pengguna](#) Amazon RDS.

- Untuk detail API, lihat [BatchExecuteStatement](#) di Referensi AWS CLI Perintah.

## begin-transaction

Contoh kode berikut menunjukkan cara menggunakan `begin-transaction`.

### AWS CLI

Untuk memulai transaksi SQL

`begin-transaction` Contoh berikut memulai transaksi SQL.

```
aws rds-data begin-transaction \
--resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
--database "mydb" \
--secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret"
```

Output:

```
{
  "transactionId": "ABC1234567890xyz"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan API Data untuk Aurora Tanpa Server di Panduan Pengguna](#) Amazon RDS.

- Untuk detail API, lihat [BeginTransaction](#) di Referensi AWS CLI Perintah.



## commit-transaction

Contoh kode berikut menunjukkan cara menggunakan `commit-transaction`.

### AWS CLI

Untuk melakukan transaksi SQL

`commit-transaction` Contoh berikut mengakhiri transaksi SQL yang ditentukan dan melakukan perubahan yang Anda buat sebagai bagian dari itu.

```
aws rds-data commit-transaction \  
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \  
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \  
  --transaction-id "ABC1234567890xyz"
```

Output:

```
{  
  "transactionStatus": "Transaction Committed"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan API Data untuk Aurora Tanpa Server di Panduan Pengguna Amazon RDS](#).

- Untuk detail API, lihat [CommitTransaction](#) di Referensi AWS CLI Perintah.

## execute-statement

Contoh kode berikut menunjukkan cara menggunakan `execute-statement`.

### AWS CLI

Contoh 1: Untuk mengeksekusi pernyataan SQL yang merupakan bagian dari transaksi

`execute-statement` Contoh berikut menjalankan pernyataan SQL yang merupakan bagian dari transaksi.

```
aws rds-data execute-statement \  
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \  
  --database "mydb" \  
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \  
  --statement "SELECT * FROM mytable"
```

```
--sql "update mytable set quantity=5 where id=201" \
--transaction-id "ABC1234567890xyz"
```

Output:

```
{
  "numberOfRecordsUpdated": 1
}
```

Contoh 2: Untuk mengeksekusi pernyataan SQL dengan parameter

`execute-statement` Contoh berikut menjalankan pernyataan SQL dengan parameter.

```
aws rds-data execute-statement \
  --resource-arn "arn:aws:rds:us-east-1:123456789012:cluster:mydbcluster" \
  --database "mydb" \
  --secret-arn "arn:aws:secretsmanager:us-east-1:123456789012:secret:mysecret" \
  --sql "insert into mytable values (:id, :val)" \
  --parameters "[{\"name\": \"id\", \"value\": {\"longValue\": 1}}, {\"name\": \"val\", \"value\": {\"stringValue\": \"value1\"}}]"
```

Output:

```
{
  "numberOfRecordsUpdated": 1
}
```

Untuk informasi selengkapnya, lihat [Menggunakan API Data untuk Aurora Tanpa Server di Panduan Pengguna](#) Amazon RDS.

- Untuk detail API, lihat [ExecuteStatement](#) di Referensi AWS CLI Perintah.

## **rollback-transaction**

Contoh kode berikut menunjukkan cara menggunakan `rollback-transaction`.

AWS CLI

Untuk memutar kembali transaksi SQL

`rollback-transaction` Contoh berikut menggulung kembali transaksi SQL yang ditentukan.

```
aws rds-data rollback-transaction \  
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \  
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \  
  --transaction-id "ABC1234567890xyz"
```

Output:

```
{  
  "transactionStatus": "Rollback Complete"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan API Data untuk Aurora Tanpa Server di Panduan Pengguna](#) Amazon RDS.

- Untuk detail API, lihat [RollbackTransaction](#) di Referensi AWS CLI Perintah.

## Contoh Performance Insights Amazon RDS menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum AWS Command Line Interface dengan menggunakan Performance Insights Amazon RDS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-dimension-keys**

Contoh kode berikut menunjukkan cara menggunakan `describe-dimension-keys`.

## AWS CLI

Untuk menggambarkan kunci dimensi

Contoh ini meminta nama semua acara tunggu. Data dirangkum berdasarkan nama acara, dan nilai agregat dari peristiwa tersebut selama periode waktu yang ditentukan.

Perintah:

```
aws pi describe-dimension-keys --service-type RDS --identifier db-
LKCG0BK26374TPTDFX0IWVCPMM --start-time 1527026400 --end-time 1527080400 --metric
db.load.avg --group-by '{"Group":"db.wait_event"}
```

Output:

```
{
  "AlignedEndTime": 1.5270804E9,
  "AlignedStartTime": 1.5270264E9,
  "Keys": [
    {
      "Dimensions": {"db.wait_event.name": "wait/synch/mutex/innodb/
aurora_lock_thread_slot_futex"},
      "Total": 0.05906906851195666
    },
    {
      "Dimensions": {"db.wait_event.name": "wait/io/aurora_redo_log_flush"},
      "Total": 0.015824722186149193
    },
    {
      "Dimensions": {"db.wait_event.name": "CPU"},
      "Total": 0.008014396230265477
    },
    {
      "Dimensions": {"db.wait_event.name": "wait/io/
aurora_respond_to_client"},
      "Total": 0.0036361612526204477
    },
    {
      "Dimensions": {"db.wait_event.name": "wait/io/table/sql/handler"},
      "Total": 0.0019108398419382965
    },
    {
```

```

    "Dimensions": {"db.wait_event.name": "wait/synch/cond/mysys/
my_thread_var::suspend"},
    "Total": 8.533847837782684E-4
  },
  {
    "Dimensions": {"db.wait_event.name": "wait/io/file/csv/data"},
    "Total": 6.864181956477376E-4
  },
  {
    "Dimensions": {"db.wait_event.name": "Unknown"},
    "Total": 3.895887056379051E-4
  },
  {
    "Dimensions": {"db.wait_event.name": "wait/synch/mutex/sql/
FILE_AS_TABLE::LOCK_shim_lists"},
    "Total": 3.710368625122906E-5
  },
  {
    "Dimensions": {"db.wait_event.name": "wait/lock/table/sql/handler"},
    "Total": 0
  }
]
}

```

- Untuk detail API, lihat [DescribeDimensionKeys](#) di Referensi AWS CLI Perintah.

## get-resource-metrics

Contoh kode berikut menunjukkan cara menggunakan `get-resource-metrics`.

### AWS CLI

Untuk mendapatkan metrik sumber daya

Contoh ini meminta titik data untuk grup dimensi `db.wait_event`, dan untuk dimensi `db.wait_event.name` dalam grup itu. Sebagai tanggapan, titik data yang relevan dikelompokkan berdasarkan dimensi yang diminta (`db.wait_event.name`).

Perintah:

```
aws pi get-resource-metrics --service-type RDS --identifier db-
LKCG0BK26374TPTDFX0IWVCPM --start-time 1527026400 --end-time 1527080400 --period-
in-seconds 300 --metric db.load.avg --metric-queries file://metric-queries.json
```

Argumen untuk `--metric-queries` disimpan dalam file JSON, `metric-queries.json`. Berikut adalah isi dari file tersebut:

```
[
  {
    "Metric": "db.load.avg",
    "GroupBy": {
      "Group": "db.wait_event"
    }
  }
]
```

Output:

```
{
  "AlignedEndTime": 1.5270804E9,
  "AlignedStartTime": 1.5270264E9,
  "Identifier": "db-LKCG0BK26374TPTDFX0IWVCPM",
  "MetricList": [
    {
      "Key": {
        "Metric": "db.load.avg"
      },
      "DataPoints": [
        {
          "Timestamp": 1527026700.0,
          "Value": 1.3533333333333333
        },
        {
          "Timestamp": 1527027000.0,
          "Value": 0.88
        },
        <...remaining output omitted...>
      ]
    },
    {
      "Key": {
        "Metric": "db.load.avg",
```

```

        "Dimensions": {
            "db.wait_event.name": "wait/synch/mutex/innodb/
aurora_lock_thread_slot_futex"
        }
    },
    "DataPoints": [
        {
            "Timestamp": 1527026700.0,
            "Value": 0.8566666666666667
        },
        {
            "Timestamp": 1527027000.0,
            "Value": 0.8633333333333333
        },
        <...remaining output omitted...>
    ],
    <...remaining output omitted...>
]
}

```

- Untuk detail API, lihat [GetResourceMetrics](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Redshift menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Amazon Redshift.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **accept-reserved-node-exchange**

Contoh kode berikut menunjukkan cara menggunakan `accept-reserved-node-exchange`.

#### AWS CLI

Untuk menerima pertukaran node cadangan

`accept-reserved-node-exchange` Contoh berikut menerima pertukaran node cadangan DC1 untuk node cadangan DC2.

```
aws redshift accept-reserved-node-exchange /
  --reserved-node-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE /
  --target-reserved-node-offering-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE
```

Output:

```
{
  "ExchangedReservedNode": {
    "ReservedNodeId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
    "ReservedNodeOfferingId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
    "NodeType": "dc2.large",
    "StartTime": "2019-12-06T21:17:26Z",
    "Duration": 31536000,
    "FixedPrice": 0.0,
    "UsagePrice": 0.0,
    "CurrencyCode": "USD",
    "NodeCount": 1,
    "State": "exchanging",
    "OfferingType": "All Upfront",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.0,
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "ReservedNodeOfferingType": "Regular"
  }
}
```



Untuk informasi selengkapnya, lihat [Memutakhirkan Node Cadangan Dengan AWS CLI](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [AcceptReservedNodeExchanged](#) di Referensi AWS CLI Perintah.

## **authorize-cluster-security-group-ingress**

Contoh kode berikut menunjukkan cara menggunakan `authorize-cluster-security-group-ingress`.

### AWS CLI

Mengotorisasi Akses ke Group This contoh Keamanan EC2 mengotorisasi akses ke grup keamanan Amazon EC2 bernama. Perintah:

```
aws redshift authorize-cluster-security-group-ingress --cluster-security-group-name mysecuritygroup --ec2-security-group-name myec2securitygroup --ec2-security-group-owner-id 123445677890
```

Mengotorisasi Akses ke rentang CIDR Contoh ini mengotorisasi akses ke rentang CIDR. Command:

```
aws redshift authorize-cluster-security-group-ingress --cluster-security-group-name mysecuritygroup --cidrip 192.168.100.100/32
```

- Untuk detail API, lihat [AuthorizeClusterSecurityGroupIngress](#) di Referensi AWS CLI Perintah.

## **authorize-snapshot-access**

Contoh kode berikut menunjukkan cara menggunakan `authorize-snapshot-access`.

### AWS CLI

Otorisasi AWS Akun untuk Memulihkan Snapshot This contoh mengotorisasi AWS akun 444455556666 untuk memulihkan snapshot. `my-snapshot-id` Secara default, output dalam format JSON. Command:

```
aws redshift authorize-snapshot-access --snapshot-id my-snapshot-id --account-with-restore-access 444455556666
```

**Hasil:**

```
{
  "Snapshot": {
    "Status": "available",
    "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
    "EstimatedSecondsToCompletion": 0,
    "AvailabilityZone": "us-east-1a",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "Encrypted": false,
    "OwnerAccount": "111122223333",
    "BackupProgressInMegabytes": 11.0,
    "ElapsedTimeInSeconds": 0,
    "DBName": "dev",
    "CurrentBackupRateInMegabytesPerSecond": 0.1534,
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "ActualIncrementalBackupSizeInMegabytes": 11.0,
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "TotalBackupSizeInMegabytes": 20.0,
    "Port": 5439,
    "NumberOfNodes": 2,
    "SnapshotIdentifier": "my-snapshot-id"
  }
}
```

- Untuk detail API, lihat [AuthorizeSnapshotAccess](#) di Referensi AWS CLI Perintah.

**batch-delete-cluster-snapshots**

Contoh kode berikut menunjukkan cara menggunakan `batch-delete-cluster-snapshots`.

**AWS CLI**

Untuk menghapus satu set snapshot cluster

`batch-delete-cluster-snapshots` Contoh berikut menghapus satu set snapshot cluster manual.

```
aws redshift batch-delete-cluster-snapshots \
```

```
--identifiers SnapshotIdentifier=mycluster-2019-11-06-14-12
SnapshotIdentifier=mycluster-2019-11-06-14-20
```

Output:

```
{
  "Resources": [
    "mycluster-2019-11-06-14-12",
    "mycluster-2019-11-06-14-20"
  ]
}
```

Untuk informasi selengkapnya, lihat [Snapshot Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [BatchDeleteClusterSnapshots](#) di Referensi AWS CLI Perintah.

## batch-modify-cluster-snapshots

Contoh kode berikut menunjukkan cara menggunakan `batch-modify-cluster-snapshots`.

### AWS CLI

Untuk memodifikasi satu set snapshot cluster

`batch-modify-cluster-snapshots` Contoh berikut memodifikasi pengaturan untuk satu set snapshot cluster.

```
aws redshift batch-modify-cluster-snapshots \
  --snapshot-identifier-list mycluster-2019-11-06-16-31 mycluster-2019-11-06-16-32 \
  --manual-snapshot-retention-period 30
```

Output:

```
{
  "Resources": [
    "mycluster-2019-11-06-16-31",
    "mycluster-2019-11-06-16-32"
  ],
  "Errors": [],
  "ResponseMetadata": {
```

```

    "RequestId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
    "HTTPStatusCode": 200,
    "HTTPHeaders": {
        "x-amzn-requestid": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
        "content-type": "text/xml",
        "content-length": "480",
        "date": "Sat, 07 Dec 2019 00:36:09 GMT",
        "connection": "keep-alive"
    },
    "RetryAttempts": 0
}

```

Untuk informasi selengkapnya, lihat [Snapshot Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [BatchModifyClusterSnapshots](#) di Referensi AWS CLI Perintah.

## cancel-resize

Contoh kode berikut menunjukkan cara menggunakan `cancel-resize`.

### AWS CLI

Untuk membatalkan pengubahan ukuran klaster

`cancel-resize` Contoh berikut membatalkan operasi pengubahan ukuran klasik untuk sebuah cluster.

```

aws redshift cancel-resize \
  --cluster-identifier mycluster

```

Output:

```

{
  "TargetNodeType": "dc2.large",
  "TargetNumberOfNodes": 2,
  "TargetClusterType": "multi-node",
  "Status": "CANCELLING",
  "ResizeType": "ClassicResize",
  "TargetEncryptionType": "NONE"
}

```

Untuk informasi selengkapnya, lihat [Mengubah Ukuran Cluster di Amazon Redshift di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [CancelResized](#) di Referensi AWS CLI Perintah.

## copy-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `copy-cluster-snapshot`.

### AWS CLI

Dapatkan Deskripsi Semua Cluster Versions This contoh mengembalikan deskripsi semua versi cluster. Secara default, output dalam format JSON. Command:

```
aws redshift copy-cluster-snapshot --source-snapshot-identifier
cm:examplecluster-2013-01-22-19-27-58 --target-snapshot-identifier my-saved-
snapshot-copy
```

Hasil:

```
{
  "Snapshot": {
    "Status": "available",
    "SnapshotCreateTime": "2013-01-22T19:27:58.931Z",
    "AvailabilityZone": "us-east-1c",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "ClusterCreateTime": "2013-01-22T19:23:59.368Z",
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "examplecluster",
    "Port": 5439,
    "NumberOfNodes": "2",
    "SnapshotIdentifier": "my-saved-snapshot-copy"
  },
  "ResponseMetadata": {
    "RequestId": "3b279691-64e3-11e2-bec0-17624ad140dd"
  }
}
```

- Untuk detail API, lihat [CopyClusterSnapshot](#) di Referensi AWS CLI Perintah.

## create-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `create-cluster-parameter-group`.

### AWS CLI

Buat GroupThis contoh Parameter Cluster membuat parameter cluster baru Group.Command:

```
aws redshift create-cluster-parameter-group --parameter-group-name
myclusterparametergroup --parameter-group-family redshift-1.0 --description "My
first cluster parameter group"
```

Hasil:

```
{
  "ClusterParameterGroup": {
    "ParameterGroupFamily": "redshift-1.0",
    "Description": "My first cluster parameter group",
    "ParameterGroupName": "myclusterparametergroup"
  },
  "ResponseMetadata": {
    "RequestId": "739448f0-64cc-11e2-8f7d-3b939af52818"
  }
}
```

- Untuk detail API, lihat [CreateClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## create-cluster-security-group

Contoh kode berikut menunjukkan cara menggunakan `create-cluster-security-group`.

### AWS CLI

Membuat GroupThis contoh Keamanan Cluster membuat grup keamanan klaster baru. Secara default, output dalam format JSON. Command:

```
aws redshift create-cluster-security-group --cluster-security-group-name
mysecuritygroup --description "This is my cluster security group"
```

Hasil:

```
{
```

```

"create_cluster_security_group_response": {
  "create_cluster_security_group_result": {
    "cluster_security_group": {
      "description": "This is my cluster security group",
      "owner_id": "300454760768",
      "cluster_security_group_name": "mysecuritygroup",
      "ec2_security_groups": \[],
      "ip_ranges": \[]
    }
  },
  "response_metadata": {
    "request_id": "5df486a0-343a-11e2-b0d8-d15d0ef48549"
  }
}
}

```

Anda juga dapat memperoleh informasi yang sama dalam format teks menggunakan `--output text` option.Command:

`--output text`Option.command:

Option.command:

```

aws redshift create-cluster-security-group --cluster-security-group-name
mysecuritygroup --description "This is my cluster security group" --output text

```

Hasil:

```

This is my cluster security group 300454760768 mysecuritygroup
a0c0bfab-343a-11e2-95d2-c3dc9fe8ab57

```

- Untuk detail API, lihat [CreateClusterSecurityGroup](#) di Referensi AWS CLI Perintah.

## create-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `create-cluster-snapshot`.

### AWS CLI

Buat SnapshotThis contoh Cluster membuat snapshot cluster baru. Secara default, output dalam format JSON. Command:

```
aws redshift create-cluster-snapshot --cluster-identifier mycluster --snapshot-identifier my-snapshot-id
```

Hasil:

```
{
  "Snapshot": {
    "Status": "creating",
    "SnapshotCreateTime": "2013-01-22T22:20:33.548Z",
    "AvailabilityZone": "us-east-1a",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "Port": 5439,
    "NumberOfNodes": "2",
    "SnapshotIdentifier": "my-snapshot-id"
  },
  "ResponseMetadata": {
    "RequestId": "f024d1a5-64e1-11e2-88c5-53eb05787dfb"
  }
}
```

- Untuk detail API, lihat [CreateClusterSnapshot](#) di Referensi AWS CLI Perintah.

## create-cluster-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `create-cluster-subnet-group`.

AWS CLI

Buat Group This contoh Cluster Subnet membuat cluster subnet group. Command baru:

```
aws redshift create-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup --description "My subnet group" --subnet-ids subnet-763fdd1c
```

Hasil:



```
{
  "ClusterSubnetGroup": {
    "Subnets": [
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-763fdd1c",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1a"
        }
      }
    ],
    "VpcId": "vpc-7e3fdd14",
    "SubnetGroupStatus": "Complete",
    "Description": "My subnet group",
    "ClusterSubnetGroupName": "mysubnetgroup"
  },
  "ResponseMetadata": {
    "RequestId": "500b8ce2-698f-11e2-9790-fd67517fb6fd"
  }
}
```

- Untuk detail API, lihat [CreateClusterSubnetGroup](#) di Referensi AWS CLI Perintah.

## create-cluster

Contoh kode berikut menunjukkan cara menggunakan `create-cluster`.

### AWS CLI

Buat Cluster dengan ParametersThis contoh Minimal membuat cluster dengan set minimal parameter. Secara default, output dalam format JSON. Command:

```
aws redshift create-cluster --node-type dw.hs1.xlarge --number-of-nodes 2 --master-username adminuser --master-user-password TopSecret1 --cluster-identifier mycluster
```

Hasil:

```
{
  "Cluster": {
    "NodeType": "dw.hs1.xlarge",
    "ClusterVersion": "1.0",
    "PubliclyAccessible": "true",
```

```

"MasterUsername": "adminuser",
"ClusterParameterGroups": [
  {
    "ParameterApplyStatus": "in-sync",
    "ParameterGroupName": "default.redshift-1.0"
  } ],
"ClusterSecurityGroups": [
  {
    "Status": "active",
    "ClusterSecurityGroupName": "default"
  } ],
"AllowVersionUpgrade": true,
"VpcSecurityGroups": \[],
"PreferredMaintenanceWindow": "sat:03:30-sat:04:00",
"AutomatedSnapshotRetentionPeriod": 1,
"ClusterStatus": "creating",
"ClusterIdentifier": "mycluster",
"DBName": "dev",
"NumberOfNodes": 2,
"PendingModifiedValues": {
  "MasterUserPassword": "\*****"
}
},
"ResponseMetadata": {
  "RequestId": "7cf4bcfc-64dd-11e2-bea9-49e0ce183f07"
}
}

```

- Untuk detail API, lihat [CreateCluster](#) di Referensi AWS CLI Perintah.

## create-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `create-event-subscription`.

### AWS CLI

Untuk membuat langganan notifikasi untuk suatu acara

`create-event-subscription` Contoh berikut membuat langganan pemberitahuan acara.

```

aws redshift create-event-subscription \
  --subscription-name mysubscription \
  --sns-topic-arn arn:aws:sns:us-west-2:123456789012:MySNSTopic \

```

```
--source-type cluster \  
--source-ids mycluster
```

### Output:

```
{  
  "EventSubscription": {  
    "CustomerAwsId": "123456789012",  
    "CustSubscriptionId": "mysubscription",  
    "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNSStopic",  
    "Status": "active",  
    "SubscriptionCreationTime": "2019-12-09T20:05:19.365Z",  
    "SourceType": "cluster",  
    "SourceIdsList": [  
      "mycluster"  
    ],  
    "EventCategoriesList": [],  
    "Severity": "INFO",  
    "Enabled": true,  
    "Tags": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan Peristiwa Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [CreateEventSubscription](#) di Referensi AWS CLI Perintah.

## create-hsm-client-certificate

Contoh kode berikut menunjukkan cara menggunakan `create-hsm-client-certificate`.

### AWS CLI

Untuk membuat sertifikat klien HSM

`create-hsm-client-certificate` Contoh berikut menghasilkan sertifikat klien HSM yang dapat digunakan cluster untuk terhubung ke HSM.

```
aws redshift create-hsm-client-certificate \  
--hsm-client-certificate-identifier myhsmclientcert
```

**Output:**

```
{
  "HsmClientCertificate": {
    "HsmClientCertificateIdentifier": "myhsmclientcert",
    "HsmClientCertificatePublicKey": "-----BEGIN CERTIFICATE-----
MIICiEXAMPLECQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTEXAMPLEEwDgYDVQHEwdTZWF0dGx1MQ8wDQYDVQKQEWZBbWF6
b24xFDASBgNVBAStC01BTSBDb25EXAMPLEIwEAYDVQQDEw1UZXR0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb2EXAMPLETEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBEXAMPLEMRAwDgYD
EXAMPLETZWF0dGx1MQ8wDQYDVQKQEWZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
b2x1MRIwEAEXAMPLEEw1UZXR0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKEXAMPLEAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk6EXAMPLE3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugEXAMPLEzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEEEXAMPLEEAtCu4
nUhVVxYUEXAMPLEEh8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GEXAMPLE10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rEXAMPLE=-----END CERTIFICATE-----\n",
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Referensi Izin API Amazon Redshift di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [CreateHsmClientCertificate](#) di Referensi AWS CLI Perintah.

**create-hsm-configuration**

Contoh kode berikut menunjukkan cara menggunakan `create-hsm-configuration`.

**AWS CLI**

Untuk membuat konfigurasi HSM

`create-hsm-configuration` Contoh berikut membuat konfigurasi HSM tertentu yang berisi informasi yang diperlukan oleh cluster untuk menyimpan dan menggunakan kunci enkripsi database dalam modul keamanan perangkat keras (HSM).

```
aws redshift create-hsm-configuration /
```

```
--hsm-configuration-identifier myhsmconnection
--description "My HSM connection"
--hsm-ip-address 192.0.2.09
--hsm-partition-name myhsmpartition /
--hsm-partition-password A1b2c3d4 /
--hsm-server-public-certificate myhsmclientcert
```

Output:

```
{
  "HsmConfiguration": {
    "HsmConfigurationIdentifier": "myhsmconnection",
    "Description": "My HSM connection",
    "HsmIpAddress": "192.0.2.09",
    "HsmPartitionName": "myhsmpartition",
    "Tags": []
  }
}
```

- Untuk detail API, lihat [CreateHsmConfiguration](#) di Referensi AWS CLI Perintah.

## create-snapshot-copy-grant

Contoh kode berikut menunjukkan cara menggunakan `create-snapshot-copy-grant`.

AWS CLI

Untuk membuat hibah salinan snapshot

`create-snapshot-copy-grant` Contoh berikut membuat hibah salinan snapshot dan mengenkripsi snapshot yang disalin di Wilayah tujuan. AWS

```
aws redshift create-snapshot-copy-grant \
  --snapshot-copy-grant-name mysnapshotcopygrantname
```

Output:

```
{
  "SnapshotCopyGrant": {
    "SnapshotCopyGrantName": "mysnapshotcopygrantname",
```

```
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvbEXAMPLEKEY",
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Enkripsi Database Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [CreateSnapshotCopyGrant](#) di Referensi AWS CLI Perintah.

## create-snapshot-schedule

Contoh kode berikut menunjukkan cara menggunakan `create-snapshot-schedule`.

### AWS CLI

Untuk membuat jadwal snapshot

`create-snapshot-schedule` Contoh berikut membuat jadwal snapshot dengan deskripsi yang ditentukan dan tingkat setiap 12 jam.

```
aws redshift create-snapshot-schedule \
  --schedule-definitions "rate(12 hours)" \
  --schedule-identifier mysnapshotschedule \
  --schedule-description "My schedule description"
```

Output:

```
{
  "ScheduleDefinitions": [
    "rate(12 hours)"
  ],
  "ScheduleIdentifier": "mysnapshotschedule",
  "ScheduleDescription": "My schedule description",
  "Tags": []
}
```

Untuk informasi selengkapnya, lihat [Jadwal Snapshot Otomatis di Panduan](#) Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [CreateSnapshotSchedule](#) di Referensi AWS CLI Perintah.

## create-tags

Contoh kode berikut menunjukkan cara menggunakan `create-tags`.

### AWS CLI

Untuk membuat tag untuk sebuah cluster

`create-tags` Contoh berikut menambahkan pasangan kunci tag/nilai yang ditentukan ke cluster tertentu.

```
aws redshift create-tags \  
  --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \  
  --tags "Key"="mytags","Value"="tag1"
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya di Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [CreateTags](#) di Referensi AWS CLI Perintah.

## delete-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster-parameter-group`.

### AWS CLI

Hapus Group This contoh Parameter Cluster menghapus parameter cluster Group. Command:

```
aws redshift delete-cluster-parameter-group --parameter-group-name  
myclusterparametergroup
```

- Untuk detail API, lihat [DeleteClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## delete-cluster-security-group

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster-security-group`.

### AWS CLI

Menghapus Group This contoh Keamanan Cluster menghapus grup keamanan klaster. Command:

```
aws redshift delete-cluster-security-group --cluster-security-group-name
mysecuritygroup
```

- Untuk detail API, lihat [DeleteClusterSecurityGroup](#) di Referensi AWS CLI Perintah.

## **delete-cluster-snapshot**

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster-snapshot`.

### AWS CLI

Menghapus Snapshot This contoh Cluster menghapus cluster Snapshot. Command:

```
aws redshift delete-cluster-snapshot --snapshot-identifier my-snapshot-id
```

- Untuk detail API, lihat [DeleteClusterSnapshot](#) di Referensi AWS CLI Perintah.

## **delete-cluster-subnet-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster-subnet-group`.

### AWS CLI

Menghapus Group This contoh subnet Cluster menghapus cluster subnet group. Command:

```
aws redshift delete-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
```

Hasil:

```
{
  "ResponseMetadata": {
    "RequestId": "253fbffd-6993-11e2-bc3a-47431073908a"
  }
}
```

- Untuk detail API, lihat [DeleteClusterSubnetGroup](#) di Referensi AWS CLI Perintah.

## **delete-cluster**

Contoh kode berikut menunjukkan cara menggunakan `delete-cluster`.



## AWS CLI

Hapus Cluster tanpa SnapshotThis contoh Final Cluster menghapus cluster, memaksa penghapusan data sehingga tidak ada snapshot cluster akhir yang dibuat.Command:

```
aws redshift delete-cluster --cluster-identifier mycluster --skip-final-cluster-snapshot
```

Menghapus Cluster, Mengizinkan SnapshotThis contoh Final Cluster menghapus klaster, tetapi menentukan snapshot.Command cluster akhir:

```
aws redshift delete-cluster --cluster-identifier mycluster --final-cluster-snapshot-identifier myfinalsnapshot
```

- Untuk detail API, lihat [DeleteCluster](#) di Referensi AWS CLI Perintah.

## delete-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `delete-event-subscription`.

### AWS CLI

Untuk menghapus langganan acara

`delete-event-subscription`Contoh berikut menghapus langganan pemberitahuan acara yang ditentukan.

```
aws redshift delete-event-subscription \
  --subscription-name mysubscription
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan Peristiwa Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DeleteEventSubscription](#) di Referensi AWS CLI Perintah.

## delete-hsm-client-certificate

Contoh kode berikut menunjukkan cara menggunakan `delete-hsm-client-certificate`.

## AWS CLI

Untuk menghapus sertifikat klien HSM

`delete-hsm-client-certificate` Contoh berikut menghapus sertifikat klien HSM.

```
aws redshift delete-hsm-client-certificate \  
  --hsm-client-certificate-identifier myhsmclientcert
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Referensi Izin API Amazon Redshift di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [DeleteHsmClientCertificate](#) di Referensi AWS CLI Perintah.

## **delete-hsm-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-hsm-configuration`.

## AWS CLI

Untuk menghapus konfigurasi HSM

`delete-hsm-configuration` Contoh berikut menghapus konfigurasi HSM yang ditentukan dari akun saat ini AWS .

```
aws redshift delete-hsm-configuration /  
  --hsm-configuration-identifier myhsmconnection
```

Perintah ini tidak menghasilkan output apa pun.

- Untuk detail API, lihat [DeleteHsmConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-scheduled-action**

Contoh kode berikut menunjukkan cara menggunakan `delete-scheduled-action`.

## AWS CLI

Untuk menghapus tindakan terjadwal

`delete-scheduled-action` Contoh berikut menghapus tindakan terjadwal yang ditentukan.

```
aws redshift delete-scheduled-action \  
  --scheduled-action-name myscheduledaction
```

Perintah ini tidak menghasilkan output apa pun.

- Untuk detail API, lihat [DeleteScheduledAction](#) di Referensi AWS CLI Perintah.

## `delete-snapshot-copy-grant`

Contoh kode berikut menunjukkan cara menggunakan `delete-snapshot-copy-grant`.

### AWS CLI

Untuk menghapus hibah salinan snapshot

`delete-snapshot-copy-grant` Contoh berikut menghapus hibah salinan snapshot yang ditentukan.

```
aws redshift delete-snapshot-copy-grant \  
  --snapshot-copy-grant-name mysnapshotcopygrantname
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Enkripsi Database Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DeleteSnapshotCopyGrant](#) di Referensi AWS CLI Perintah.

## `delete-snapshot-schedule`

Contoh kode berikut menunjukkan cara menggunakan `delete-snapshot-schedule`.

### AWS CLI

Untuk menghapus jadwal snapshot

`delete-snapshot-schedule` Contoh berikut menghapus jadwal snapshot yang ditentukan. Anda harus memisahkan cluster sebelum menghapus jadwal.

```
aws redshift delete-snapshot-schedule \  
  --cluster-id myclusterid
```

```
--schedule-identifier mysnapshotschedule
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Jadwal Snapshot Otomatis di Panduan](#) Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DeleteSnapshotScheduled](#) di Referensi AWS CLI Perintah.

## **delete-tags**

Contoh kode berikut menunjukkan cara menggunakan `delete-tags`.

### AWS CLI

Untuk menghapus tag dari klaster

`delete-tags` Contoh berikut menghapus tag dengan nama kunci yang ditentukan dari cluster yang ditentukan.

```
aws redshift delete-tags \  
  --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \  
  --tag-keys "clustertagkey" "clustertagvalue"
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya di Amazon](#) Redshift di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DeleteTags](#) di Referensi AWS CLI Perintah.

## **describe-account-attributes**

Contoh kode berikut menunjukkan cara menggunakan `describe-account-attributes`.

### AWS CLI

Untuk mendeskripsikan atribut AWS akun

`describe-account-attributes` Contoh berikut menampilkan atribut yang dilampirkan ke AWS akun panggilan.

```
aws redshift describe-account-attributes
```

Output:

```
{
  "AccountAttributes": [
    {
      "AttributeName": "max-defer-maintenance-duration",
      "AttributeValues": [
        {
          "AttributeValue": "45"
        }
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeAccountAttributes](#) di Referensi AWS CLI Perintah.

## describe-cluster-db-revisions

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-db-revisions`.

### AWS CLI

Untuk menggambarkan revisi DB untuk sebuah cluster

`describe-cluster-db-revisions` Contoh berikut menampilkan rincian array `ClusterDbRevision` objek untuk cluster tertentu.

```
aws redshift describe-cluster-db-revisions \
  --cluster-identifier mycluster
```

Output:

```
{
  "ClusterDbRevisions": [
    {
      "ClusterIdentifier": "mycluster",
      "CurrentDatabaseRevision": "11420",
      "DatabaseRevisionReleaseDate": "2019-11-22T16:43:49.597Z",

```

```

    "RevisionTargets": []
  }
]
}

```

- Untuk detail API, lihat [DescribeClusterDbRevisions](#) di Referensi AWS CLI Perintah.

## describe-cluster-parameter-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-parameter-groups`.

### AWS CLI

Dapatkan Deskripsi All Cluster Parameter Groups. Contoh ini mengembalikan deskripsi semua kelompok parameter cluster untuk akun, dengan header kolom. Secara default, output dalam format JSON. Command:

```
aws redshift describe-cluster-parameter-groups
```

Hasil:

```

{
  "ParameterGroups": [
    {
      "ParameterGroupFamily": "redshift-1.0",
      "Description": "My first cluster parameter group",
      "ParameterGroupName": "myclusterparametergroup"
    }
  ],
  "ResponseMetadata": {
    "RequestId": "8ceb8f6f-64cc-11e2-bea9-49e0ce183f07"
  }
}

```

Anda juga dapat memperoleh informasi yang sama dalam format teks menggunakan `--output text` option. Command:

```
--output text
```

Option command:

```
aws redshift describe-cluster-parameter-groups --output text
```

**Hasil:**

```
redshift-1.0      My first cluster parameter group      myclusterparametergroup
RESPONSEMETADATA 9e665a36-64cc-11e2-8f7d-3b939af52818
```

- Untuk detail API, lihat [DescribeClusterParameterGroups](#) di Referensi AWS CLI Perintah.

**describe-cluster-parameters**

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-parameters`.

**AWS CLI**

Ambil Parameter untuk Group. Ini contoh Parameter Cluster Tertentu mengambil parameter untuk grup parameter bernama. Secara default, output dalam format JSON. Command:

```
aws redshift describe-cluster-parameters --parameter-group-name
myclusterparametergroup
```

**Hasil:**

```
{
  "Parameters": [
    {
      "Description": "Sets the display format for date and time values.",
      "DataType": "string",
      "IsModifiable": true,
      "Source": "engine-default",
      "ParameterValue": "ISO, MDY",
      "ParameterName": "datestyle"
    },
    {
      "Description": "Sets the number of digits displayed for floating-point
values",
      "DataType": "integer",
      "IsModifiable": true,
      "AllowedValues": "-15-2",
      "Source": "engine-default",
      "ParameterValue": "0",
      "ParameterName": "extra_float_digits"
    },
    (...remaining output omitted...)
  ]
}
```

```
]
}
```

Anda juga dapat memperoleh informasi yang sama dalam format teks menggunakan `--output text` option.Command:

`--output text`Option.command:

Option.command:

```
aws redshift describe-cluster-parameters --parameter-group-name
myclusterparametergroup --output text
```

Hasil:

```
RESPONSEMETADATA    cdac40aa-64cc-11e2-9e70-918437dd236d
Sets the display format for date and time values.    string True    engine-default
ISO, MDY    datestyle
Sets the number of digits displayed for floating-point values    integer True
-15-2    engine-default 0    extra_float_digits
This parameter applies a user-defined label to a group of queries that are run
during the same session..    string True    engine-default default query_group
require ssl for all databaseconnections    boolean True    true,false    engine-
default false    require_ssl
Sets the schema search order for names that are not schema-qualified.    string
True    engine-default $user, public    search_path
Aborts any statement that takes over the specified number of milliseconds.    integer
True    engine-default 0    statement_timeout
wlm json configuration    string True    engine-default
\["query_concurrency":5}]    wlm_json_configuration
```

- Untuk detail API, lihat [DescribeClusterParameters](#) di Referensi AWS CLI Perintah.

## describe-cluster-security-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-security-groups`.

AWS CLI

Dapatkan Deskripsi All Cluster Security GroupsThis contoh mengembalikan deskripsi semua grup keamanan klaster untuk akun tersebut. Secara default, output dalam format JSON. Command:



```
aws redshift describe-cluster-security-groups
```

Hasil:

```
{
  "ClusterSecurityGroups": [
    {
      "OwnerId": "100447751468",
      "Description": "default",
      "ClusterSecurityGroupName": "default",
      "EC2SecurityGroups": \[],
      "IPRanges": [
        {
          "Status": "authorized",
          "CIDRIP": "0.0.0.0/0"
        }
      ]
    },
    {
      "OwnerId": "100447751468",
      "Description": "This is my cluster security group",
      "ClusterSecurityGroupName": "mysecuritygroup",
      "EC2SecurityGroups": \[],
      "IPRanges": \[]
    },
    (...remaining output omitted...)
  ]
}
```

- Untuk detail API, lihat [DescribeClusterSecurityGroups](#) di Referensi AWS CLI Perintah.

## describe-cluster-snapshots

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-snapshots`.

### AWS CLI

Dapatkan Deskripsi Semua Cluster Snapshots. Contoh ini mengembalikan deskripsi semua snapshot cluster untuk akun. Secara default, output dalam format JSON. Command:

```
aws redshift describe-cluster-snapshots
```

**Hasil:**

```
{
  "Snapshots": [
    {
      "Status": "available",
      "SnapshotCreateTime": "2013-07-17T22:02:22.852Z",
      "EstimatedSecondsToCompletion": -1,
      "AvailabilityZone": "us-east-1a",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "Encrypted": false,
      "OwnerAccount": "111122223333",
      "BackupProgressInMegabytes": 20.0,
      "ElapsedTimeInSeconds": 0,
      "DBName": "dev",
      "CurrentBackupRateInMegabytesPerSecond": 0.0,
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
      "ActualIncrementalBackupSizeInMegabytes"; 20.0
      "SnapshotType": "automated",
      "NodeType": "dw.hs1.xlarge",
      "ClusterIdentifier": "mycluster",
      "Port": 5439,
      "TotalBackupSizeInMegabytes": 20.0,
      "NumberOfNodes": "2",
      "SnapshotIdentifier": "cm:mycluster-2013-01-22-22-04-18"
    },
    {
      "EstimatedSecondsToCompletion": 0,
      "OwnerAccount": "111122223333",
      "CurrentBackupRateInMegabytesPerSecond": 0.1534,
      "ActualIncrementalBackupSizeInMegabytes"; 11.0,
      "NumberOfNodes": "2",
      "Status": "available",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "AccountsWithRestoreAccess": [
        {
          "AccountID": "444455556666"
        }
      ],
      "TotalBackupSizeInMegabytes": 20.0,
      "DBName": "dev",
      "BackupProgressInMegabytes": 11.0,
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",

```

```

    "ElapsedTimeInSeconds": 0,
    "ClusterIdentifier": "mycluster",
    "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
    "AvailabilityZone": "us-east-1a",
    "NodeType": "dw.hs1.xlarge",
    "Encrypted": false,
    "SnapshotType": "manual",
    "Port": 5439,
    "SnapshotIdentifier": "my-snapshot-id"
  } ]
}
(...remaining output omitted...)

```

- Untuk detail API, lihat [DescribeClusterSnapshots](#) di Referensi AWS CLI Perintah.

## describe-cluster-subnet-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-subnet-groups`.

### AWS CLI

Dapatkan Deskripsi Semua Cluster Subnet Groups. Contoh ini mengembalikan deskripsi semua kelompok subnet cluster. Secara default, output dalam format JSON. Command:

```
aws redshift describe-cluster-subnet-groups
```

Hasil:

```

{
  "ClusterSubnetGroups": [
    {
      "Subnets": [
        {
          "SubnetStatus": "Active",
          "SubnetIdentifier": "subnet-763fdd1c",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
          }
        }
      ],
      "VpcId": "vpc-7e3fdd14",
      "SubnetGroupStatus": "Complete",
    }
  ]
}

```

```

        "Description": "My subnet group",
        "ClusterSubnetGroupName": "mysubnetgroup"
    }
],
"ResponseMetadata": {
    "RequestId": "37fa8c89-6990-11e2-8f75-ab4018764c77"
}
}

```

- Untuk detail API, lihat [DescribeClusterSubnetGroups](#) di Referensi AWS CLI Perintah.

## describe-cluster-tracks

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-tracks`.

### AWS CLI

Untuk menggambarkan trek cluster

`describe-cluster-tracks` Contoh berikut menampilkan rincian trek pemeliharaan yang tersedia.

```

aws redshift describe-cluster-tracks \
  --maintenance-track-name current

```

Output:

```

{
  "MaintenanceTracks": [
    {
      "MaintenanceTrackName": "current",
      "DatabaseVersion": "1.0.11420",
      "UpdateTargets": [
        {
          "MaintenanceTrackName": "preview_features",
          "DatabaseVersion": "1.0.11746",
          "SupportedOperations": [
            {
              "OperationName": "restore-from-cluster-snapshot"
            }
          ]
        }
      ]
    }
  ],
}

```

```

        {
            "MaintenanceTrackName": "trailing",
            "DatabaseVersion": "1.0.11116",
            "SupportedOperations": [
                {
                    "OperationName": "restore-from-cluster-snapshot"
                },
                {
                    "OperationName": "modify-cluster"
                }
            ]
        }
    ]
}

```

Untuk informasi selengkapnya, lihat [Memilih Trek Pemeliharaan Cluster](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeClusterTracks](#) di Referensi AWS CLI Perintah.

## describe-cluster-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-cluster-versions`.

### AWS CLI

Dapatkan Deskripsi Semua Cluster Versions. Contoh ini mengembalikan deskripsi semua versi cluster. Secara default, output dalam format JSON. Command:

```
aws redshift describe-cluster-versions
```

Hasil:

```

{
  "ClusterVersions": [
    {
      "ClusterVersion": "1.0",
      "Description": "Initial release",
      "ClusterParameterGroupFamily": "redshift-1.0"
    }
  ],
}

```

```
"ResponseMetadata": {  
  "RequestId": "16a53de3-64cc-11e2-bec0-17624ad140dd"  
}
```

- Untuk detail API, lihat [DescribeClusterVersions](#) di Referensi AWS CLI Perintah.

## describe-clusters

Contoh kode berikut menunjukkan cara menggunakan `describe-clusters`.

### AWS CLI

Dapatkan Deskripsi Semua Clusters This contoh mengembalikan deskripsi semua cluster untuk akun. Secara default, output dalam format JSON. Command:

```
aws redshift describe-clusters
```

### Hasil:

```
{  
  "Clusters": [  
    {  
      "NodeType": "dw.hs1.xlarge",  
      "Endpoint": {  
        "Port": 5439,  
        "Address": "mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com"  
      },  
      "ClusterVersion": "1.0",  
      "PubliclyAccessible": "true",  
      "MasterUsername": "adminuser",  
      "ClusterParameterGroups": [  
        {  
          "ParameterApplyStatus": "in-sync",  
          "ParameterGroupName": "default.redshift-1.0"  
        } ],  
      "ClusterSecurityGroups": [  
        {  
          "Status": "active",  
          "ClusterSecurityGroupName": "default"  
        } ],  
      "AllowVersionUpgrade": true,  
    }  
  ]  
}
```

```

    "VpcSecurityGroups": \[],
    "AvailabilityZone": "us-east-1a",
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "PreferredMaintenanceWindow": "sat:03:30-sat:04:00",
    "AutomatedSnapshotRetentionPeriod": 1,
    "ClusterStatus": "available",
    "ClusterIdentifier": "mycluster",
    "DBName": "dev",
    "NumberOfNodes": 2,
    "PendingModifiedValues": {}
  } ],
  "ResponseMetadata": {
    "RequestId": "65b71cac-64df-11e2-8f5b-e90bd6c77476"
  }
}

```

Anda juga dapat memperoleh informasi yang sama dalam format teks menggunakan `--output text` option.Command:

`--output textOption.command:`

Option.command:

```
aws redshift describe-clusters --output text
```

Hasil:

```

dw.hs1.xlarge      1.0      true      adminuser      True      us-east-1a
2013-01-22T21:59:29.559Z      sat:03:30-sat:04:00      1      available
mycluster      dev      2
ENDPOINT      5439      mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com
in-sync      default.redshift-1.0
active      default
PENDINGMODIFIEDVALUES
RESPONSEMETADATA      934281a8-64df-11e2-b07c-f7fbdd006c67

```

- Untuk detail API, lihat [DescribeClusters](#) di Referensi AWS CLI Perintah.

## describe-default-cluster-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-default-cluster-parameters`.

## AWS CLI

Dapatkan Deskripsi ParametersThis contoh Cluster Default mengembalikan deskripsi parameter cluster default untuk `redshift-1.0` keluarga. Secara default, output dalam format JSON.Command:

```
aws redshift describe-default-cluster-parameters --parameter-group-family
redshift-1.0
```

Hasil:

```
{
  "DefaultClusterParameters": {
    "ParameterGroupFamily": "redshift-1.0",
    "Parameters": [
      {
        "Description": "Sets the display format for date and time values.",
        "DataType": "string",
        "IsModifiable": true,
        "Source": "engine-default",
        "ParameterValue": "ISO, MDY",
        "ParameterName": "datestyle"
      },
      {
        "Description": "Sets the number of digits displayed for floating-point
values",
        "DataType": "integer",
        "IsModifiable": true,
        "AllowedValues": "-15-2",
        "Source": "engine-default",
        "ParameterValue": "0",
        "ParameterName": "extra_float_digits"
      },
      (...remaining output omitted...)
    ]
  }
}
```

Untuk melihat daftar keluarga grup parameter yang valid, gunakan `describe-cluster-parameter-groups` perintah.

`describe-cluster-parameter-groups`perintah.



perintah.

- Untuk detail API, lihat [DescribeDefaultClusterParameters](#) di Referensi AWS CLI Perintah.

## describe-event-categories

Contoh kode berikut menunjukkan cara menggunakan `describe-event-categories`.

### AWS CLI

Untuk mendeskripsikan kategori acara untuk sebuah kluster

`describe-event-categories` Contoh berikut menampilkan rincian untuk kategori acara untuk cluster.

```
aws redshift describe-event-categories \
  --source-type cluster
```

Output:

```
{
  "EventCategoriesMapList": [
    {
      "SourceType": "cluster",
      "Events": [
        {
          "EventId": "REDSHIFT-EVENT-2000",
          "EventCategories": [
            "management"
          ],
          "EventDescription": "Cluster <cluster name> created at <time in
UTC>.",
          "Severity": "INFO"
        },
        {
          "EventId": "REDSHIFT-EVENT-2001",
          "EventCategories": [
            "management"
          ],
          "EventDescription": "Cluster <cluster name> deleted at <time in
UTC>.",
          "Severity": "INFO"
        }
      ]
    }
  ]
}
```

```

        {
            "EventId": "REDSHIFT-EVENT-3625",
            "EventCategories": [
                "monitoring"
            ],
            "EventDescription": "The cluster <cluster name> can't be resumed
with its previous elastic network interface <ENI id>. We will allocate a new
elastic network interface and associate it with the cluster node.",
            "Severity": "INFO"
        }
    ]
}

```

- Untuk detail API, lihat [DescribeEventCategories](#) di Referensi AWS CLI Perintah.

## describe-event-subscriptions

Contoh kode berikut menunjukkan cara menggunakan `describe-event-subscriptions`.

### AWS CLI

Untuk menggambarkan langganan acara

`describe-event-subscriptions` Contoh berikut menampilkan langganan pemberitahuan acara untuk langganan yang ditentukan.

```
aws redshift describe-event-subscriptions \
  --subscription-name mysubscription
```

Output:

```

{
  "EventSubscriptionsList": [
    {
      "CustomerAwsId": "123456789012",
      "CustSubscriptionId": "mysubscription",
      "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNStopic",
      "Status": "active",
      "SubscriptionCreationTime": "2019-12-09T21:50:21.332Z",
      "SourceIdsList": [],
    }
  ]
}

```

```
    "EventCategoriesList": [
      "management"
    ],
    "Severity": "ERROR",
    "Enabled": true,
    "Tags": []
  }
]
```

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan Peristiwa Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeEventSubscriptions](#) di Referensi AWS CLI Perintah.

## describe-events

Contoh kode berikut menunjukkan cara menggunakan `describe-events`.

### AWS CLI

Jelaskan Semua Peristiwa Contoh ini mengembalikan semua peristiwa. Secara default, output dalam format JSON. Command:

```
aws redshift describe-events
```

Hasil:

```
{
  "Events": [
    {
      "Date": "2013-01-22T19:17:03.640Z",
      "SourceIdentifier": "myclusterparametergroup",
      "Message": "Cluster parameter group myclusterparametergroup has been
created.",
      "SourceType": "cluster-parameter-group"
    }
  ],
  "ResponseMetadata": {
    "RequestId": "9f056111-64c9-11e2-9390-ff04f2c1e638"
  }
}
```

Anda juga dapat memperoleh informasi yang sama dalam format teks menggunakan `--output text` option.Command:

`--output text`Option.command:

Option.command:

```
aws redshift describe-events --output text
```

Hasil:

```
2013-01-22T19:17:03.640Z    myclusterparametergroup Cluster parameter group
myclusterparametergroup has been created.    cluster-parameter-group
RESPONSEMETADATA    8e5fe765-64c9-11e2-bce3-e56f52c50e17
```

- Untuk detail API, lihat [DescribeEvents](#) di Referensi AWS CLI Perintah.

## describe-hsm-client-certificates

Contoh kode berikut menunjukkan cara menggunakan `describe-hsm-client-certificates`.

AWS CLI

Untuk menggambarkan sertifikat klien HSM

`describe-hsm-client-certificates` Contoh berikut menampilkan rincian untuk sertifikat klien HSM yang ditentukan.

```
aws redshift describe-hsm-client-certificates \
--hsm-client-certificate-identifier myhsmclientcert
```

Output:

```
{
  "HsmClientCertificates": [
    {
      "HsmClientCertificateIdentifier": "myhsmclientcert",
      "HsmClientCertificatePublicKey": "-----BEGIN CERTIFICATE-----\
EXAMPLECAfICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
```

```
VVMxCzAJBgNVBAsTC01BTSBDb25zEXAMPLEwEAYDVQQDEwLUZXN0Q21sYWMxHzAd
b24xFDASBgNVBAsTC01BTSBDb25zEXAMPLEwEAYDVQQDEwLUZXN0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhEXAMPLEDI1MjA0EXAMPLEN
EXAMPLE0MjA0NTIxWjCBiDELMAkGA1UEBhMCMVVMxCzAJBgNVBAsTC01BTSBDb25z
VQQHEwdTZWF0dGEXAMPLEQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDb25z
b2x1MRIwEAYDVQQDEwLUZXN0Q21sEXAMPLEdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIEXAMPLEMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY67EXAMPLEE
EXAMPLEZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9EXAMPLE6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEEXAMPLEBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rEXAMPLE=-----END CERTIFICATE-----\n",
"Tags": []
}
]
}
```

Untuk informasi selengkapnya, lihat [Referensi Izin API Amazon Redshift di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [DescribeHsmClientCertificates](#) di Referensi AWS CLI Perintah.

## describe-hsm-configurations

Contoh kode berikut menunjukkan cara menggunakan `describe-hsm-configurations`.

### AWS CLI

Untuk menggambarkan konfigurasi HSM

`describe-hsm-configurations` Contoh berikut menampilkan detail untuk konfigurasi HSM yang tersedia untuk akun panggilan AWS .

```
aws redshift describe-hsm-configurations /
--hsm-configuration-identifier myhsmconnection
```

Output:

```
{
  "HsmConfigurations": [
    {
```

```
        "HsmConfigurationIdentifier": "myhsmconnection",
        "Description": "My HSM connection",
        "HsmIpAddress": "192.0.2.09",
        "HsmPartitionName": "myhsmpartition",
        "Tags": []
    }
]
```

- Untuk detail API, lihat [DescribeHsmConfigurations](#) di Referensi AWS CLI Perintah.

## describe-logging-status

Contoh kode berikut menunjukkan cara menggunakan `describe-logging-status`.

### AWS CLI

Untuk mendeskripsikan status logging untuk sebuah klaster

`describe-logging-status` Contoh berikut menampilkan apakah informasi, seperti query dan upaya koneksi, sedang dicatat untuk sebuah cluster.

```
aws redshift describe-logging-status \
  --cluster-identifier mycluster
```

Output:

```
{
  "LoggingEnabled": false
}
```

Untuk informasi selengkapnya, lihat [Logging Audit Database](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeLoggingStatus](#) di Referensi AWS CLI Perintah.

## describe-node-configuration-options

Contoh kode berikut menunjukkan cara menggunakan `describe-node-configuration-options`.

## AWS CLI

Untuk menggambarkan opsi konfigurasi simpul

`describe-node-configuration-options` Contoh berikut menampilkan properti konfigurasi node yang mungkin seperti tipe node, jumlah node, dan penggunaan disk untuk snapshot cluster yang ditentukan.

```
aws redshift describe-node-configuration-options \  
  --action-type restore-cluster \  
  --snapshot-identifier rs:mycluster-2019-12-09-16-42-43
```

Output:

```
{  
  "NodeConfigurationOptionList": [  
    {  
      "NodeType": "dc2.large",  
      "NumberOfNodes": 2,  
      "EstimatedDiskUtilizationPercent": 19.61  
    },  
    {  
      "NodeType": "dc2.large",  
      "NumberOfNodes": 4,  
      "EstimatedDiskUtilizationPercent": 9.96  
    },  
    {  
      "NodeType": "ds2.xlarge",  
      "NumberOfNodes": 2,  
      "EstimatedDiskUtilizationPercent": 1.53  
    },  
    {  
      "NodeType": "ds2.xlarge",  
      "NumberOfNodes": 4,  
      "EstimatedDiskUtilizationPercent": 0.78  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membeli Node Cadangan Amazon Redshift di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [DescribeNodeConfigurationOptions](#) di Referensi AWS CLI Perintah.

## describe-orderable-cluster-options

Contoh kode berikut menunjukkan cara menggunakan `describe-orderable-cluster-options`.

### AWS CLI

Menjelaskan All Orderable Cluster Options. This contoh mengembalikan deskripsi dari semua opsi cluster yang dapat dipesan. Secara default, output dalam format JSON. Command:

```
aws redshift describe-orderable-cluster-options
```

### Hasil:

```
{
  "OrderableClusterOptions": [
    {
      "NodeType": "dw.hs1.8xlarge",
      "AvailabilityZones": [
        { "Name": "us-east-1a" },
        { "Name": "us-east-1b" },
        { "Name": "us-east-1c" } ],
      "ClusterVersion": "1.0",
      "ClusterType": "multi-node"
    },
    {
      "NodeType": "dw.hs1.xlarge",
      "AvailabilityZones": [
        { "Name": "us-east-1a" },
        { "Name": "us-east-1b" },
        { "Name": "us-east-1c" } ],
      "ClusterVersion": "1.0",
      "ClusterType": "multi-node"
    },
    {
      "NodeType": "dw.hs1.xlarge",
      "AvailabilityZones": [
        { "Name": "us-east-1a" },
        { "Name": "us-east-1b" },
        { "Name": "us-east-1c" } ],
      "ClusterVersion": "1.0",
      "ClusterType": "single-node"
    } ],
  "ResponseMetadata": {
```



```

    "RequestId": "f6000035-64cb-11e2-9135-ff82df53a51a"
  }
}

```

Anda juga dapat memperoleh informasi yang sama dalam format teks menggunakan `--output text` option. Command:

`--output text` option command:

Option command:

```
aws redshift describe-orderable-cluster-options --output text
```

Hasil:

```

dw.hs1.8xlarge      1.0      multi-node
us-east-1a
us-east-1b
us-east-1c
dw.hs1.xlarge      1.0      multi-node
us-east-1a
us-east-1b
us-east-1c
dw.hs1.xlarge      1.0      single-node
us-east-1a
us-east-1b
us-east-1c
RESPONSEMETADATA   e648696b-64cb-11e2-bec0-17624ad140dd

```

- Untuk detail API, lihat [DescribeOrderableClusterOptions](#) di Referensi AWS CLI Perintah.

## describe-reserved-node-offerings

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-node-offerings`.

AWS CLI

Jelaskan Offerings This contoh Node Cadangan menunjukkan semua penawaran node cadangan yang tersedia untuk Purchase. command:

```
aws redshift describe-reserved-node-offerings
```

**Hasil:**

```
{
  "ReservedNodeOfferings": [
    {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.xlarge",
      "UsagePrice": "",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": "",
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "Duration": 31536000,
      "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
    },
    {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.8xlarge",
      "UsagePrice": "",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": "",
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "Duration": 31536000,
      "ReservedNodeOfferingId": "e5a2ff3b-352d-4a9c-ad7d-373c4cab5dd2"
    },
    ...remaining output omitted...
  ],
  "ResponseMetadata": {
    "RequestId": "8b1a1a43-75ff-11e2-9666-e142fe91ddd1"
  }
}
```

Jika Anda ingin membeli penawaran node yang dicadangkan, Anda dapat menelepon `purchase-reserved-node-offering` menggunakan valid `ReservedNodeOfferingId`.

`purchase-reserved-node-offering` menggunakan yang valid `ReservedNodeOfferingId`.

menggunakan valid `ReservedNodeOfferingId`.

ReservedNodeOfferingId.

- Untuk detail API, lihat [DescribeReservedNodeOfferings](#) di Referensi AWS CLI Perintah.

## describe-reserved-nodes

Contoh kode berikut menunjukkan cara menggunakan `describe-reserved-nodes`.

### AWS CLI

NodesThis Contoh Description Reserved menunjukkan penawaran node reserved yang telah dibeli. Command:

```
aws redshift describe-reserved-nodes
```

Hasil:

```
{
  "ResponseMetadata": {
    "RequestId": "bc29ce2e-7600-11e2-9949-4b361e7420b7"
  },
  "ReservedNodes": [
    {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.xlarge",
      "ReservedNodeId": "1ba8e2e3-bc01-4d65-b35d-a4a3e931547e",
      "UsagePrice": "",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": "",
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "NodeCount": 1,
      "State": "payment-pending",
      "StartTime": "2013-02-13T17:08:39.051Z",
      "Duration": 31536000,
      "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
    }
  ]
}
```

```
}
```

- Untuk detail API, lihat [DescribeReservedNodes](#) di Referensi AWS CLI Perintah.

## describe-resize

Contoh kode berikut menunjukkan cara menggunakan `describe-resize`.

### AWS CLI

Jelaskan `ResizeThis` contoh menjelaskan ukuran terbaru dari sebuah cluster. Permintaan itu untuk 3 node tipe `dw.hs1.8xlarge`. Command:

```
aws redshift describe-resize --cluster-identifier mycluster
```

Hasil:

```
{
  "Status": "NONE",
  "TargetClusterType": "multi-node",
  "TargetNodeType": "dw.hs1.8xlarge",
  "ResponseMetadata": {
    "RequestId": "9f52b0b4-7733-11e2-aa9b-318b2909bd27"
  },
  "TargetNumberOfNodes": "3"
}
```

- Untuk detail API, lihat [DescribeResized](#) di Referensi AWS CLI Perintah.

## describe-scheduled-actions

Contoh kode berikut menunjukkan cara menggunakan `describe-scheduled-actions`.

### AWS CLI

Untuk menggambarkan tindakan terjadwal

`describe-scheduled-actions` Contoh berikut menampilkan detail untuk setiap tindakan yang dijadwalkan saat ini.

```
aws redshift describe-scheduled-actions
```

**Output:**

```
{
  "ScheduledActions": [
    {
      "ScheduledActionName": "resizecluster",
      "TargetAction": {
        "ResizeCluster": {
          "ClusterIdentifier": "mycluster",
          "NumberOfNodes": 4,
          "Classic": false
        }
      },
      "Schedule": "at(2019-12-10T00:07:00)",
      "IamRole": "arn:aws:iam::123456789012:role/myRedshiftRole",
      "State": "ACTIVE",
      "NextInvocations": [
        "2019-12-10T00:07:00Z"
      ]
    }
  ]
}
```

- Untuk detail API, lihat [DescribeScheduledActions](#) di Referensi AWS CLI Perintah.

**describe-snapshot-copy-grants**

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshot-copy-grants`.

**AWS CLI**

Untuk menggambarkan hibah salinan snapshot

`describe-snapshot-copy-grants` Contoh berikut menampilkan rincian untuk hibah salinan snapshot cluster yang ditentukan.

```
aws redshift describe-snapshot-copy-grants \
  --snapshot-copy-grant-name mysnapshotcopygrantname
```

**Output:**

```
{
```

```

    "SnapshotCopyGrants": [
      {
        "SnapshotCopyGrantName": "mysnapshotcopygrantname",
        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvbEXAMPLEKEY",
        "Tags": []
      }
    ]
  }

```

Untuk informasi selengkapnya, lihat [Enkripsi Database Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeSnapshotCopyGrants](#) di Referensi AWS CLI Perintah.

## describe-snapshot-schedules

Contoh kode berikut menunjukkan cara menggunakan `describe-snapshot-schedules`.

### AWS CLI

Untuk menggambarkan jadwal snapshot

`describe-snapshot-schedules` Contoh berikut menampilkan rincian untuk jadwal snapshot cluster yang ditentukan.

```

aws redshift describe-snapshot-schedules \
  --cluster-identifier mycluster \
  --schedule-identifier mysnapshotschedule

```

Output:

```

{
  "SnapshotSchedules": [
    {
      "ScheduleDefinitions": [
        "rate(12 hours)"
      ],
      "ScheduleIdentifier": "mysnapshotschedule",
      "ScheduleDescription": "My schedule description",
      "Tags": [],
      "AssociatedClusterCount": 1,
      "AssociatedClusters": [

```

```
    {
      "ClusterIdentifier": "mycluster",
      "ScheduleAssociationState": "ACTIVE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Jadwal Snapshot Otomatis di Panduan](#) Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeSnapshotSchedules](#) di Referensi AWS CLI Perintah.

## describe-storage

Contoh kode berikut menunjukkan cara menggunakan `describe-storage`.

### AWS CLI

Untuk menggambarkan penyimpanan

`describe-storage` Contoh berikut menampilkan detail tentang penyimpanan cadangan dan ukuran penyimpanan sementara untuk akun.

```
aws redshift describe-storage
```

Output:

```
{
  "TotalBackupSizeInMegaBytes": 193149.0,
  "TotalProvisionedStorageInMegaBytes": 655360.0
}
```

Untuk informasi selengkapnya, lihat [Mengelola Penyimpanan Snapshot](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeStorage](#) di Referensi AWS CLI Perintah.

## describe-table-restore-status

Contoh kode berikut menunjukkan cara menggunakan `describe-table-restore-status`.

## AWS CLI

Untuk menjelaskan status permintaan pemulihan tabel dari snapshot cluster

`describe-table-restore-status` Contoh berikut menampilkan rincian untuk permintaan pemulihan tabel yang dibuat untuk cluster tertentu.

```
aws redshift describe-table-restore-status /
--cluster-identifier mycluster
```

Output:

```
{
  "TableRestoreStatusDetails": [
    {
      "TableRestoreRequestId": "z1116630-0e80-46f4-ba86-bd9670411ebd",
      "Status": "IN_PROGRESS",
      "RequestTime": "2019-12-27T18:22:12.257Z",
      "ClusterIdentifier": "mycluster",
      "SnapshotIdentifier": "mysnapshotid",
      "SourceDatabaseName": "dev",
      "SourceSchemaName": "public",
      "SourceTableName": "mytable",
      "TargetDatabaseName": "dev",
      "TargetSchemaName": "public",
      "NewTableName": "mytable-clone"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulihkan Tabel dari Snapshot di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [DescribeTableRestoreStatus](#) di Referensi AWS CLI Perintah.

## **describe-tags**

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

## AWS CLI

Untuk menggambarkan tag



`describe-tags` Contoh berikut menampilkan sumber daya cluster tertentu terkait dengan nama tag tertentu dan nilai-nilai.

```
aws redshift describe-tags \  
  --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \  
  --tag-keys clustertagkey \  
  --tag-values clustertagvalue
```

Output:

```
{  
  "TaggedResources": [  
    {  
      "Tag": {  
        "Key": "clustertagkey",  
        "Value": "clustertagvalue"  
      },  
      "ResourceName": "arn:aws:redshift:us-  
west-2:123456789012:cluster:mycluster",  
      "ResourceType": "cluster"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya di Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## **disable-snapshot-copy**

Contoh kode berikut menunjukkan cara menggunakan `disable-snapshot-copy`.

### AWS CLI

Untuk menonaktifkan salinan snapshot untuk kluster

`disable-snapshot-copy` Contoh berikut menonaktifkan salinan otomatis snapshot untuk cluster yang ditentukan.

```
aws redshift disable-snapshot-copy \  
  --cluster-identifier mycluster
```

## Output:

```
{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "available",
    "ClusterAvailabilityStatus": "Available",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
      "Port": 5439
    },
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
    "AutomatedSnapshotRetentionPeriod": 3,
    "ManualSnapshotRetentionPeriod": -1,
    "ClusterSecurityGroups": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sh-i9b431cd",
        "Status": "active"
      }
    ],
    "ClusterParameterGroups": [
      {
        "ParameterGroupName": "default.redshift-1.0",
        "ParameterApplyStatus": "in-sync"
      }
    ],
    "ClusterSubnetGroupName": "default",
    "VpcId": "vpc-b1fel7t9",
    "AvailabilityZone": "us-west-2f",
    "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
    "PendingModifiedValues": {
      "NodeType": "dc2.large",
      "NumberOfNodes": 2,
      "ClusterType": "multi-node"
    },
    "ClusterVersion": "1.0",
    "AllowVersionUpgrade": true,
    "NumberOfNodes": 4,
    "PubliclyAccessible": false,
    "Encrypted": false,
  }
}
```

```
    "Tags": [
      {
        "Key": "mytags",
        "Value": "tag1"
      }
    ],
    "EnhancedVpcRouting": false,
    "IamRoles": [
      {
        "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
        "ApplyStatus": "in-sync"
      }
    ],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [],
    "ExpectedNextSnapshotScheduleTime": "2019-12-10T04:42:43.390Z",
    "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
    "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Menyalin Snapshot ke AWS Wilayah Lain di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [DisableSnapshotCopy](#) di Referensi AWS CLI Perintah.

## enable-snapshot-copy

Contoh kode berikut menunjukkan cara menggunakan `enable-snapshot-copy`.

### AWS CLI

Untuk mengaktifkan salinan snapshot untuk kluster

`enable-snapshot-copy` Contoh berikut memungkinkan salinan otomatis snapshot untuk cluster yang ditentukan.

```
aws redshift enable-snapshot-copy \
  --cluster-identifier mycluster \
  --destination-region us-west-1
```

Output:

```
{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "available",
    "ClusterAvailabilityStatus": "Available",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
      "Port": 5439
    },
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
    "AutomatedSnapshotRetentionPeriod": 3,
    "ManualSnapshotRetentionPeriod": -1,
    "ClusterSecurityGroups": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sh-f4c731cd",
        "Status": "active"
      }
    ],
    "ClusterParameterGroups": [
      {
        "ParameterGroupName": "default.redshift-1.0",
        "ParameterApplyStatus": "in-sync"
      }
    ],
    "ClusterSubnetGroupName": "default",
    "VpcId": "vpc-b1ael7t9",
    "AvailabilityZone": "us-west-2f",
    "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
    "PendingModifiedValues": {
      "NodeType": "dc2.large",
      "NumberOfNodes": 2,
      "ClusterType": "multi-node"
    },
    "ClusterVersion": "1.0",
    "AllowVersionUpgrade": true,
    "NumberOfNodes": 4,
    "PubliclyAccessible": false,
    "Encrypted": false,
    "ClusterSnapshotCopyStatus": {
```

```

        "DestinationRegion": "us-west-1",
        "RetentionPeriod": 7,
        "ManualSnapshotRetentionPeriod": -1
    },
    "Tags": [
        {
            "Key": "mytags",
            "Value": "tag1"
        }
    ],
    "EnhancedVpcRouting": false,
    "IamRoles": [
        {
            "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
            "ApplyStatus": "in-sync"
        }
    ],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [],
    "ExpectedNextSnapshotScheduleTime": "2019-12-10T04:42:43.390Z",
    "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
    "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
}
}

```

Untuk informasi selengkapnya, lihat [Menyalin Snapshot ke AWS Wilayah Lain di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [EnableSnapshotCopy](#) di Referensi AWS CLI Perintah.

## get-cluster-credentials

Contoh kode berikut menunjukkan cara menggunakan `get-cluster-credentials`.

### AWS CLI

Untuk mendapatkan kredensi klaster untuk akun AWS

`get-cluster-credentials` Contoh berikut mengambil kredensial sementara yang memungkinkan akses ke database Amazon Redshift.

```
aws redshift get-cluster-credentials \
  --db-user adminuser --db-name dev \
```

```
--cluster-identifier mycluster
```

Output:

```
{
  "DbUser": "IAM:adminuser",
  "DbPassword": "AMAFUyyuros/QjxPTtgzcsuQsqzIasdzJEN04aCtWDzXx109d6UmpkBtvEqFly/
EXAMPLE==",
  "Expiration": "2019-12-10T17:25:05.770Z"
}
```

Untuk informasi selengkapnya, lihat [Membuat Kredensial Database IAM Menggunakan Amazon Redshift CLI atau API di Panduan Manajemen Cluster](#) Amazon Redshift.

- Untuk detail API, lihat [GetClusterCredentials](#) di Referensi AWS CLI Perintah.

## get-reserved-node-exchange-offerings

Contoh kode berikut menunjukkan cara menggunakan `get-reserved-node-exchange-offerings`.

AWS CLI

Untuk mendapatkan penawaran pertukaran node yang dipesan

`get-reserved-node-exchange-offerings` Contoh berikut mengambil array yang cocok dengan node DC1 reserved yang ditentukan. DC2 ReservedNodeOfferings

```
aws redshift get-reserved-node-exchange-offerings \
  --reserved-node-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE
```

Output:

```
{
  "ReservedNodeOfferings": [
    {
      "ReservedNodeOfferingId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
      "NodeType": "dc2.large",
      "Duration": 31536000,
      "FixedPrice": 0.0,
      "UsagePrice": 0.0,
      "CurrencyCode": "USD",

```

```

        "OfferingType": "All Upfront",
        "RecurringCharges": [
            {
                "RecurringChargeAmount": 0.0,
                "RecurringChargeFrequency": "Hourly"
            }
        ],
        "ReservedNodeOfferingType": "Regular"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Memutakhirkan Node Cadangan Dengan AWS CLI](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [GetReservedNodeExchangeOfferings](#) di Referensi AWS CLI Perintah.

## modify-cluster-iam-roles

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-iam-roles`.

### AWS CLI

Untuk memodifikasi peran IAM untuk sebuah cluster

`modify-cluster-iam-roles` Contoh berikut menghapus peran AWS IAM tertentu dari cluster tertentu.

```

aws redshift modify-cluster-iam-roles \
  --cluster-identifier mycluster \
  --remove-iam-roles arn:aws:iam::123456789012:role/myRedshiftRole

```

Output:

```

{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "available",
    "ClusterAvailabilityStatus": "Available",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {

```

```
    "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
    "Port": 5439
  },
  "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
  "AutomatedSnapshotRetentionPeriod": 3,
  "ManualSnapshotRetentionPeriod": -1,
  "ClusterSecurityGroups": [],
  "VpcSecurityGroups": [
    {
      "VpcSecurityGroupId": "sh-f9b731sd",
      "Status": "active"
    }
  ],
  "ClusterParameterGroups": [
    {
      "ParameterGroupName": "default.redshift-1.0",
      "ParameterApplyStatus": "in-sync"
    }
  ],
  "ClusterSubnetGroupName": "default",
  "VpcId": "vpc-b2fal7t9",
  "AvailabilityZone": "us-west-2f",
  "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
  "PendingModifiedValues": {
    "NodeType": "dc2.large",
    "NumberOfNodes": 2,
    "ClusterType": "multi-node"
  },
  "ClusterVersion": "1.0",
  "AllowVersionUpgrade": true,
  "NumberOfNodes": 4,
  "PubliclyAccessible": false,
  "Encrypted": false,
  "ClusterSnapshotCopyStatus": {
    "DestinationRegion": "us-west-1",
    "RetentionPeriod": 7,
    "ManualSnapshotRetentionPeriod": -1
  },
  "Tags": [
    {
      "Key": "mytags",
      "Value": "tag1"
    }
  ],
],
```



```

    "EnhancedVpcRouting": false,
    "IamRoles": [],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [],
    "ExpectedNextSnapshotScheduleTime": "2019-12-11T04:42:55.631Z",
    "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
    "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Menggunakan Kebijakan Berbasis Identitas \(Kebijakan IAM\) untuk Amazon Redshift di Panduan Manajemen Cluster Amazon Redshift](#).

- Untuk detail API, lihat [ModifyClusterIamRoles](#) di Referensi AWS CLI Perintah.

## modify-cluster-maintenance

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-maintenance`.

### AWS CLI

Untuk memodifikasi pemeliharaan cluster

`modify-cluster-maintenance` Contoh berikut menunda pemeliharaan cluster yang ditentukan selama 30 hari.

```

aws redshift modify-cluster-maintenance \
  --cluster-identifier mycluster \
  --defer-maintenance \
  --defer-maintenance-duration 30

```

Output:

```

{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "available",
    "ClusterAvailabilityStatus": "Available",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",

```

```
    "Port": 5439
  },
  "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
  "AutomatedSnapshotRetentionPeriod": 3,
  "ManualSnapshotRetentionPeriod": -1,
  "ClusterSecurityGroups": [],
  "VpcSecurityGroups": [
    {
      "VpcSecurityGroupId": "sh-a1a123ab",
      "Status": "active"
    }
  ],
  "ClusterParameterGroups": [
    {
      "ParameterGroupName": "default.redshift-1.0",
      "ParameterApplyStatus": "in-sync"
    }
  ],
  "ClusterSubnetGroupName": "default",
  "VpcId": "vpc-b1ael7t9",
  "AvailabilityZone": "us-west-2f",
  "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
  "PendingModifiedValues": {
    "NodeType": "dc2.large",
    "NumberOfNodes": 2,
    "ClusterType": "multi-node"
  },
  "ClusterVersion": "1.0",
  "AllowVersionUpgrade": true,
  "NumberOfNodes": 4,
  "PubliclyAccessible": false,
  "Encrypted": false,
  "ClusterSnapshotCopyStatus": {
    "DestinationRegion": "us-west-1",
    "RetentionPeriod": 7,
    "ManualSnapshotRetentionPeriod": -1
  },
  "Tags": [
    {
      "Key": "mytags",
      "Value": "tag1"
    }
  ],
  "EnhancedVpcRouting": false,
```

```

    "IamRoles": [],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [
      {
        "DeferMaintenanceIdentifier": "dfm-mUdVIffFcT1B4SGhw6fyF",
        "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
        "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
      }
    ],
    "ExpectedNextSnapshotScheduleTime": "2019-12-11T04:42:55.631Z",
    "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
    "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Pemeliharaan Cluster](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ModifyClusterMaintenance](#) di Referensi AWS CLI Perintah.

## modify-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-parameter-group`.

### AWS CLI

Memodifikasi parameter dalam grup parameter

`modify-cluster-parameter-group` Contoh berikut memodifikasi parameter `wlm_json_configuration` untuk manajemen beban kerja. Ini menerima parameter dari file yang berisi konten JSON yang ditunjukkan di bawah ini.

```

aws redshift modify-cluster-parameter-group \
  --parameter-group-name myclusterparametergroup \
  --parameters file://modify_pg.json

```

Isi dari `modify_pg.json`:

```

[
  {
    "ParameterName": "wlm_json_configuration",
    "ParameterValue": "[{\"user_group\": \"example_user_group1\", \"query_group\": \"example_query_group1\", \"query_concurrency\": 7}, {\"query_concurrency\": 5}]"
  }
]

```

```
    }
  ]
}
```

Output:

```
{
  "ParameterGroupStatus": "Your parameter group has been updated but changes won't
  get applied until you reboot the associated Clusters.",
  "ParameterGroupName": "myclusterparametergroup",
  "ResponseMetadata": {
    "RequestId": "09974cc0-64cd-11e2-bea9-49e0ce183f07"
  }
}
```

- Untuk detail API, lihat [ModifyClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## modify-cluster-snapshot-schedule

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-snapshot-schedule`.

### AWS CLI

Untuk mengubah jadwal snapshot cluster

`modify-cluster-snapshot-schedule` Contoh berikut menghapus jadwal snapshot yang ditentukan dari cluster yang ditentukan.

```
aws redshift modify-cluster-snapshot-schedule \
  --cluster-identifier mycluster \
  --schedule-identifier mysnapshotschedule \
  --disassociate-schedule
```

Perintah ini tidak menghasilkan output apa pun.

Untuk informasi selengkapnya, lihat [Jadwal Snapshot Otomatis di Panduan](#) Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ModifyClusterSnapshotSchedule](#) di Referensi AWS CLI Perintah.

## modify-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-snapshot`.

## AWS CLI

Untuk memodifikasi snapshot cluster

`modify-cluster-snapshot` Contoh berikut menetapkan pengaturan periode retensi manual untuk snapshot cluster yang ditentukan ke nilai 10 hari.

```
aws redshift modify-cluster-snapshot \  
--snapshot-identifier mycluster-2019-11-06-16-32 \  
--manual-snapshot-retention-period 10
```

Output:

```
{  
  "Snapshot": {  
    "SnapshotIdentifier": "mycluster-2019-11-06-16-32",  
    "ClusterIdentifier": "mycluster",  
    "SnapshotCreateTime": "2019-12-07T00:34:05.633Z",  
    "Status": "available",  
    "Port": 5439,  
    "AvailabilityZone": "us-west-2f",  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "MasterUsername": "adminuser",  
    "ClusterVersion": "1.0",  
    "SnapshotType": "manual",  
    "NodeType": "dc2.large",  
    "NumberOfNodes": 2,  
    "DBName": "dev",  
    "VpcId": "vpc-b1cel7t9",  
    "Encrypted": false,  
    "EncryptedWithHSM": false,  
    "OwnerAccount": "123456789012",  
    "TotalBackupSizeInMegaBytes": 64384.0,  
    "ActualIncrementalBackupSizeInMegaBytes": 24.0,  
    "BackupProgressInMegaBytes": 24.0,  
    "CurrentBackupRateInMegaBytesPerSecond": 13.0011,  
    "EstimatedSecondsToCompletion": 0,  
    "ElapsedTimeInSeconds": 1,  
    "Tags": [  
      {  
        "Key": "mytagkey",  
        "Value": "mytagvalue"  
      }  
    ]  
  }  
}
```

```
    ],  
    "EnhancedVpcRouting": false,  
    "MaintenanceTrackName": "current",  
    "ManualSnapshotRetentionPeriod": 10,  
    "ManualSnapshotRemainingDays": 6,  
    "SnapshotRetentionStartTime": "2019-12-07T00:34:07.479Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Snapshot Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ModifyClusterSnapshot](#) di Referensi AWS CLI Perintah.

## modify-cluster-subnet-group

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster-subnet-group`.

### AWS CLI

Memodifikasi Subnet dalam Group This contoh Cluster Subnet menunjukkan cara memodifikasi daftar subnet dalam grup subnet cache. Secara default, output dalam format JSON. Command:

```
aws redshift modify-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup  
--subnet-ids subnet-763fdd1 subnet-ac830e9
```

### Hasil:

```
{  
  "ClusterSubnetGroup":  
  {  
    "Subnets": [  
      {  
        "SubnetStatus": "Active",  
        "SubnetIdentifier": "subnet-763fdd1c",  
        "SubnetAvailabilityZone":  
          { "Name": "us-east-1a" }  
      },  
      {  
        "SubnetStatus": "Active",  
        "SubnetIdentifier": "subnet-ac830e9",  
        "SubnetAvailabilityZone":
```

```

        { "Name": "us-east-1b" }
      } ],
      "VpcId": "vpc-7e3fdd14",
      "SubnetGroupStatus": "Complete",
      "Description": "My subnet group",
      "ClusterSubnetGroupName": "mysubnetgroup"
    },
    "ResponseMetadata": {
      "RequestId": "8da93e89-8372-f936-93a8-873918938197a"
    }
  }
}

```

- Untuk detail API, lihat [ModifyClusterSubnetGroup](#) di Referensi AWS CLI Perintah.

## modify-cluster

Contoh kode berikut menunjukkan cara menggunakan `modify-cluster`.

### AWS CLI

Mengaitkan Grup Keamanan dengan ClusterThis contoh menunjukkan cara mengaitkan grup keamanan klaster dengan `Cluster.command` yang ditentukan:

```
aws redshift modify-cluster --cluster-identifier mycluster --cluster-security-groups mysecuritygroup
```

Ubah Jendela Pemeliharaan untuk ClusterThis menunjukkan cara mengubah jendela pemeliharaan pilihan mingguan untuk cluster menjadi jendela minimum empat jam mulai hari Minggu pukul 11:15, dan berakhir Senin pukul 3:15 pagi. Perintah:

```
aws redshift modify-cluster --cluster-identifier mycluster --preferred-maintenance-window Sun:23:15-Mon:03:15
```

Ubah Kata Sandi Master untuk ClusterThis contoh menunjukkan cara mengubah kata sandi master untuk cluster. Command:

```
aws redshift modify-cluster --cluster-identifier mycluster --master-user-password A1b2c3d4
```

- Untuk detail API, lihat [ModifyCluster](#) di Referensi AWS CLI Perintah.

## modify-event-subscription

Contoh kode berikut menunjukkan cara menggunakan `modify-event-subscription`.

### AWS CLI

Untuk mengubah langganan acara

`modify-event-subscription` Contoh berikut menonaktifkan langganan pemberitahuan acara yang ditentukan.

```
aws redshift modify-event-subscription \  
  --subscription-name mysubscription \  
  --no-enabled
```

Output:

```
{  
  "EventSubscription": {  
    "CustomerAwsId": "123456789012",  
    "CustSubscriptionId": "mysubscription",  
    "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNSStopic",  
    "Status": "active",  
    "SubscriptionCreationTime": "2019-12-09T21:50:21.332Z",  
    "SourceIdsList": [],  
    "EventCategoriesList": [  
      "management"  
    ],  
    "Severity": "ERROR",  
    "Enabled": false,  
    "Tags": []  
  }  
}
```

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan Peristiwa Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ModifyEventSubscription](#) di Referensi AWS CLI Perintah.

## modify-scheduled-action

Contoh kode berikut menunjukkan cara menggunakan `modify-scheduled-action`.



## AWS CLI

Untuk mengubah tindakan terjadwal

`modify-scheduled-action` Contoh berikut menambahkan deskripsi untuk tindakan terjadwal tertentu yang ada.

```
aws redshift modify-scheduled-action \  
  --scheduled-action-name myscheduledaction \  
  --scheduled-action-description "My scheduled action"
```

Output:

```
{  
  "ScheduledActionName": "myscheduledaction",  
  "TargetAction": {  
    "ResizeCluster": {  
      "ClusterIdentifier": "mycluster",  
      "NumberOfNodes": 2,  
      "Classic": false  
    }  
  },  
  "Schedule": "at(2019-12-25T00:00:00)",  
  "IamRole": "arn:aws:iam::123456789012:role/myRedshiftRole",  
  "ScheduledActionDescription": "My scheduled action",  
  "State": "ACTIVE",  
  "NextInvocations": [  
    "2019-12-25T00:00:00Z"  
  ]  
}
```

- Untuk detail API, lihat [ModifyScheduledAction](#) di Referensi AWS CLI Perintah.

## `modify-snapshot-copy-retention-period`

Contoh kode berikut menunjukkan cara menggunakan `modify-snapshot-copy-retention-period`.

## AWS CLI

Untuk mengubah periode penyimpanan salinan snapshot

`modify-snapshot-copy-retention-period` Contoh berikut memodifikasi jumlah hari untuk menyimpan snapshot untuk cluster yang ditentukan di AWS Wilayah tujuan setelah disalin dari Wilayah sumber. AWS

```
aws redshift modify-snapshot-copy-retention-period \  
  --cluster-identifier mycluster \  
  --retention-period 15
```

Output:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "available",  
    "ClusterAvailabilityStatus": "Available",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    },  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "AutomatedSnapshotRetentionPeriod": 3,  
    "ManualSnapshotRetentionPeriod": -1,  
    "ClusterSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sh-a1a123ab",  
        "Status": "active"  
      }  
    ],  
    "ClusterParameterGroups": [  
      {  
        "ParameterGroupName": "default.redshift-1.0",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ],  
    "ClusterSubnetGroupName": "default",  
    "VpcId": "vpc-b1fet7t9",  
    "AvailabilityZone": "us-west-2f",  
    "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",  
    "PendingModifiedValues": {
```

```

        "NodeType": "dc2.large",
        "NumberOfNodes": 2,
        "ClusterType": "multi-node"
    },
    "ClusterVersion": "1.0",
    "AllowVersionUpgrade": true,
    "NumberOfNodes": 4,
    "PubliclyAccessible": false,
    "Encrypted": false,
    "ClusterSnapshotCopyStatus": {
        "DestinationRegion": "us-west-1",
        "RetentionPeriod": 15,
        "ManualSnapshotRetentionPeriod": -1
    },
    "Tags": [
        {
            "Key": "mytags",
            "Value": "tag1"
        }
    ],
    "EnhancedVpcRouting": false,
    "IamRoles": [],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [
        {
            "DeferMaintenanceIdentifier": "dfm-mUdVSfDcT1F4SGhw6fyF",
            "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
            "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
        }
    ],
    "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z"
}
}

```

Untuk informasi selengkapnya, lihat [Format Jadwal Snapshot](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ModifySnapshotCopyRetentionPeriod](#) di Referensi AWS CLI Perintah.

## **modify-snapshot-schedule**

Contoh kode berikut menunjukkan cara menggunakan `modify-snapshot-schedule`.

## AWS CLI

Untuk mengubah jadwal snapshot

`modify-snapshot-schedule` Contoh berikut memodifikasi tingkat jadwal snapshot yang ditentukan untuk setiap 10 jam.

```
aws redshift modify-snapshot-schedule \  
  --schedule-identifier mysnapshotschedule \  
  --schedule-definitions "rate(10 hours)"
```

Output:

```
{  
  "ScheduleDefinitions": [  
    "rate(10 hours)"  
  ],  
  "ScheduleIdentifier": "mysnapshotschedule",  
  "ScheduleDescription": "My schedule description",  
  "Tags": []  
}
```

Untuk informasi selengkapnya, lihat [Format Jadwal Snapshot](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ModifySnapshotSchedule](#) di Referensi AWS CLI Perintah.

## **purchase-reserved-node-offering**

Contoh kode berikut menunjukkan cara menggunakan `purchase-reserved-node-offering`.

### AWS CLI

Membeli Node This contoh Reserved menunjukkan cara membeli penawaran node reserved. `reserved-node-offering-id` ini diperoleh dengan `describe-reserved-node-offerings` memanggil. Command:

```
aws redshift purchase-reserved-node-offering --reserved-node-offering-id ceb6a579-  
cf4c-4343-be8b-d832c45ab51c
```

Hasil:

```
{
  "ReservedNode": {
    "OfferingType": "Heavy Utilization",
    "FixedPrice": "",
    "NodeType": "dw.hs1.xlarge",
    "ReservedNodeId": "1ba8e2e3-bc01-4d65-b35d-a4a3e931547e",
    "UsagePrice": "",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": "",
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "NodeCount": 1,
    "State": "payment-pending",
    "StartTime": "2013-02-13T17:08:39.051Z",
    "Duration": 31536000,
    "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
  },
  "ResponseMetadata": {
    "RequestId": "01bda7bf-7600-11e2-b605-2568d7396e7f"
  }
}
```

- Untuk detail API, lihat [PurchaseReservedNodeOffering](#) di Referensi AWS CLI Perintah.

## reboot-cluster

Contoh kode berikut menunjukkan cara menggunakan `reboot-cluster`.

### AWS CLI

Reboot Cluster This contoh me-reboot sebuah cluster. Secara default, output dalam format JSON. Command:

```
aws redshift reboot-cluster --cluster-identifier mycluster
```

Hasil:

```
{
  "Cluster": {
```

```
"NodeType": "dw.hs1.xlarge",
"Endpoint": {
  "Port": 5439,
  "Address": "mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com"
},
"ClusterVersion": "1.0",
"PubliclyAccessible": "true",
"MasterUsername": "adminuser",
"ClusterParameterGroups": [
  {
    "ParameterApplyStatus": "in-sync",
    "ParameterGroupName": "default.redshift-1.0"
  }
],
"ClusterSecurityGroups": [
  {
    "Status": "active",
    "ClusterSecurityGroupName": "default"
  }
],
"AllowVersionUpgrade": true,
"VpcSecurityGroups": [],
"AvailabilityZone": "us-east-1a",
"ClusterCreateTime": "2013-01-22T21:59:29.559Z",
"PreferredMaintenanceWindow": "sun:23:15-mon:03:15",
"AutomatedSnapshotRetentionPeriod": 1,
"ClusterStatus": "rebooting",
"ClusterIdentifier": "mycluster",
"DBName": "dev",
"NumberOfNodes": 2,
"PendingModifiedValues": {}
},
"ResponseMetadata": {
  "RequestId": "61c8b564-64e8-11e2-8f7d-3b939af52818"
}
}
```

- Untuk detail API, lihat [RebootCluster](#) di Referensi AWS CLI Perintah.

## reset-cluster-parameter-group

Contoh kode berikut menunjukkan cara menggunakan `reset-cluster-parameter-group`.

## AWS CLI

Reset Parameter dalam GroupThis contoh Parameter menunjukkan cara mengatur ulang semua parameter dalam parameter Group.Command:

```
aws redshift reset-cluster-parameter-group --parameter-group-name
myclusterparametergroup --reset-all-parameters
```

- Untuk detail API, lihat [ResetClusterParameterGroup](#) di Referensi AWS CLI Perintah.

## resize-cluster

Contoh kode berikut menunjukkan cara menggunakan `resize-cluster`.

### AWS CLI

Untuk mengubah ukuran cluster

`resize-cluster` Contoh berikut mengubah ukuran cluster tertentu.

```
aws redshift resize-cluster \
  --cluster-identifier mycluster \
  --cluster-type multi-node \
  --node-type dc2.large \
  --number-of-nodes 6 \
  --classic
```

Output:

```
{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "resizing",
    "ClusterAvailabilityStatus": "Modifying",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
      "Port": 5439
    },
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
```

```
"AutomatedSnapshotRetentionPeriod": 3,
"ManualSnapshotRetentionPeriod": -1,
"ClusterSecurityGroups": [],
"VpcSecurityGroups": [
  {
    "VpcSecurityGroupId": "sh-a1a123ab",
    "Status": "active"
  }
],
"ClusterParameterGroups": [
  {
    "ParameterGroupName": "default.redshift-1.0",
    "ParameterApplyStatus": "in-sync"
  }
],
"ClusterSubnetGroupName": "default",
"VpcId": "vpc-a1abc1a1",
"AvailabilityZone": "us-west-2f",
"PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
"PendingModifiedValues": {
  "NodeType": "dc2.large",
  "NumberOfNodes": 6,
  "ClusterType": "multi-node"
},
"ClusterVersion": "1.0",
"AllowVersionUpgrade": true,
"NumberOfNodes": 4,
"PubliclyAccessible": false,
"Encrypted": false,
"ClusterSnapshotCopyStatus": {
  "DestinationRegion": "us-west-1",
  "RetentionPeriod": 15,
  "ManualSnapshotRetentionPeriod": -1
},
"Tags": [
  {
    "Key": "mytags",
    "Value": "tag1"
  }
],
"EnhancedVpcRouting": false,
"IamRoles": [],
"MaintenanceTrackName": "current",
"DeferredMaintenanceWindows": [
```



```

    {
      "DeferMaintenanceIdentifier": "dfm-mUdVCfDcT1B4SGhw6fyF",
      "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
      "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
    }
  ],
  "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z",
  "ResizeInfo": {
    "ResizeType": "ClassicResize",
    "AllowCancelResize": true
  }
}

```

Untuk informasi selengkapnya, lihat [Mengubah Ukuran Cluster](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [ResizeCluster](#) di Referensi AWS CLI Perintah.

## restore-from-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `restore-from-cluster-snapshot`.

### AWS CLI

Memulihkan Cluster Dari Snapshot This contoh mengembalikan cluster dari Snapshot. Command:

```
aws redshift restore-from-cluster-snapshot --cluster-identifier mycluster-clone --
snapshot-identifier my-snapshot-id
```

Hasil:

```

{
  "Cluster": {
    "NodeType": "dw.hs1.xlarge",
    "ClusterVersion": "1.0",
    "PubliclyAccessible": "true",
    "MasterUsername": "adminuser",
    "ClusterParameterGroups": [
      {
        "ParameterApplyStatus": "in-sync",
        "ParameterGroupName": "default.redshift-1.0"
      }
    ]
  }
}

```

```

    }
  ],
  "ClusterSecurityGroups": [
    {
      "Status": "active",
      "ClusterSecurityGroupName": "default"
    }
  ],
  "AllowVersionUpgrade": true,
  "VpcSecurityGroups": \[],
  "PreferredMaintenanceWindow": "sun:23:15-mon:03:15",
  "AutomatedSnapshotRetentionPeriod": 1,
  "ClusterStatus": "creating",
  "ClusterIdentifier": "mycluster-clone",
  "DBName": "dev",
  "NumberOfNodes": 2,
  "PendingModifiedValues": {}
},
"ResponseMetadata": {
  "RequestId": "77fd512b-64e3-11e2-8f5b-e90bd6c77476"
}
}

```

- Untuk detail API, lihat [RestoreFromClusterSnapshot](#) di Referensi AWS CLI Perintah.

## restore-table-from-cluster-snapshot

Contoh kode berikut menunjukkan cara menggunakan `restore-table-from-cluster-snapshot`.

### AWS CLI

Untuk mengembalikan tabel dari snapshot cluster

`restore-table-from-cluster-snapshot` Contoh berikut membuat tabel baru dari tabel yang ditentukan dalam snapshot cluster tertentu.

```

aws redshift restore-table-from-cluster-snapshot /
--cluster-identifier mycluster /
--snapshot-identifier mycluster-2019-11-19-16-17 /
--source-database-name dev /
--source-schema-name public /
--source-table-name mytable /

```

```
--target-database-name dev /  
--target-schema-name public /  
--new-table-name mytable-clone
```

### Output:

```
{  
  "TableRestoreStatus": {  
    "TableRestoreRequestId": "a123a12b-abc1-1a1a-a123-a1234ab12345",  
    "Status": "PENDING",  
    "RequestTime": "2019-12-20T00:20:16.402Z",  
    "ClusterIdentifier": "mycluster",  
    "SnapshotIdentifier": "mycluster-2019-11-19-16-17",  
    "SourceDatabaseName": "dev",  
    "SourceSchemaName": "public",  
    "SourceTableName": "mytable",  
    "TargetDatabaseName": "dev",  
    "TargetSchemaName": "public",  
    "NewTableName": "mytable-clone"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memulihkan Tabel dari Snapshot di Panduan](#) Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [RestoreTableFromClusterSnapshot](#) di Referensi AWS CLI Perintah.

## revoke-cluster-security-group-ingress

Contoh kode berikut menunjukkan cara menggunakan `revoke-cluster-security-group-ingress`.

### AWS CLI

Mencabut Akses dari GroupThis contoh Keamanan EC2 mencabut akses ke grup keamanan Amazon EC2 bernama. Perintah:

```
aws redshift revoke-cluster-security-group-ingress --cluster-security-group-name  
mysecuritygroup --ec2-security-group-name myec2securitygroup --ec2-security-group-  
owner-id 123445677890
```

Mencabut Akses ke rentang CIDR Contoh ini mencabut akses ke rentang CIDR. Command:

```
aws redshift revoke-cluster-security-group-ingress --cluster-security-group-name
mysecuritygroup --cidrip 192.168.100.100/32
```

- Untuk detail API, lihat [RevokeClusterSecurityGroupIngress](#) di Referensi AWS CLI Perintah.

## revoke-snapshot-access

Contoh kode berikut menunjukkan cara menggunakan `revoke-snapshot-access`.

### AWS CLI

Mencabut otorisasi AWS akun untuk memulihkan Snapshot. Contoh mencabut otorisasi AWS akun 444455556666 untuk mengembalikan snapshot. `my-snapshot-id` Secara default, output dalam format JSON. Command:

```
aws redshift revoke-snapshot-access --snapshot-id my-snapshot-id --account-with-
restore-access 444455556666
```

Hasil:

```
{
  "Snapshot": {
    "Status": "available",
    "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
    "EstimatedSecondsToCompletion": 0,
    "AvailabilityZone": "us-east-1a",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "Encrypted": false,
    "OwnerAccount": "111122223333",
    "BackupProgressInMegabytes": 11.0,
    "ElapsedTimeInSeconds": 0,
    "DBName": "dev",
    "CurrentBackupRateInMegabytesPerSecond": 0.1534,
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "ActualIncrementalBackupSizeInMegabytes": 11.0,
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "TotalBackupSizeInMegabytes": 20.0,
    "Port": 5439,
```

```

    "NumberOfNodes": 2,
    "SnapshotIdentifier": "my-snapshot-id"
  }
}

```

- Untuk detail API, lihat [RevokeSnapshotAccess](#) di Referensi AWS CLI Perintah.

## rotate-encryption-key

Contoh kode berikut menunjukkan cara menggunakan `rotate-encryption-key`.

### AWS CLI

Untuk memutar kunci enkripsi untuk klaster

`rotate-encryption-key` Contoh berikut memutar kunci enkripsi untuk cluster tertentu.

```

aws redshift rotate-encryption-key \
  --cluster-identifier mycluster

```

Output:

```

{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "rotating-keys",
    "ClusterAvailabilityStatus": "Modifying",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
      "Port": 5439
    },
    "ClusterCreateTime": "2019-12-10T19:25:45.886Z",
    "AutomatedSnapshotRetentionPeriod": 30,
    "ManualSnapshotRetentionPeriod": -1,
    "ClusterSecurityGroups": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sh-a1a123ab",
        "Status": "active"
      }
    ]
  }
}

```

```

    }
  ],
  "ClusterParameterGroups": [
    {
      "ParameterGroupName": "default.redshift-1.0",
      "ParameterApplyStatus": "in-sync"
    }
  ],
  "ClusterSubnetGroupName": "default",
  "VpcId": "vpc-a1abc1a1",
  "AvailabilityZone": "us-west-2a",
  "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
  "PendingModifiedValues": {},
  "ClusterVersion": "1.0",
  "AllowVersionUpgrade": true,
  "NumberOfNodes": 2,
  "PubliclyAccessible": false,
  "Encrypted": true,
  "Tags": [],
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvbEXAMPLEKEY",
  "EnhancedVpcRouting": false,
  "IamRoles": [
    {
      "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
      "ApplyStatus": "in-sync"
    }
  ],
  "MaintenanceTrackName": "current",
  "DeferredMaintenanceWindows": [],
  "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
}
}

```

Untuk informasi selengkapnya, lihat [Enkripsi Database Amazon Redshift](#) di Panduan Manajemen Cluster Amazon Redshift.

- Untuk detail API, lihat [RotateEncryptionKey](#) di Referensi AWS CLI Perintah.

## Contoh Rekognition Amazon menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon Rekognition.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **compare-faces**

Contoh kode berikut menunjukkan cara menggunakan `compare-faces`.

Untuk informasi selengkapnya, lihat [Membandingkan wajah dalam gambar](#).

AWS CLI

Untuk membandingkan wajah dalam dua gambar

`compare-faces` Perintah berikut membandingkan wajah dalam dua gambar yang disimpan dalam bucket Amazon S3.

```
aws rekognition compare-faces \  
  --source-image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"source.jpg"}}' \  
  --target-image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"target.jpg"}}'
```

Output:

```
{  
  "UnmatchedFaces": [],  
  "FaceMatches": [  
    {  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.12368916720151901,  
          "Top": 0.16007372736930847,
```

```
        "Left": 0.5901257991790771,  
        "Height": 0.25140416622161865  
    },  
    "Confidence": 100.0,  
    "Pose": {  
        "Yaw": -3.7351467609405518,  
        "Roll": -0.10309021919965744,  
        "Pitch": 0.8637830018997192  
    },  
    "Quality": {  
        "Sharpness": 95.51618957519531,  
        "Brightness": 65.29893493652344  
    },  
    "Landmarks": [  
        {  
            "Y": 0.26721030473709106,  
            "X": 0.6204193830490112,  
            "Type": "eyeLeft"  
        },  
        {  
            "Y": 0.26831310987472534,  
            "X": 0.6776827573776245,  
            "Type": "eyeRight"  
        },  
        {  
            "Y": 0.3514654338359833,  
            "X": 0.6241428852081299,  
            "Type": "mouthLeft"  
        },  
        {  
            "Y": 0.35258132219314575,  
            "X": 0.6713621020317078,  
            "Type": "mouthRight"  
        },  
        {  
            "Y": 0.3140771687030792,  
            "X": 0.6428444981575012,  
            "Type": "nose"  
        }  
    ]  
},  
"Similarity": 100.0  
}  
],
```



```
"SourceImageFace": {
  "BoundingBox": {
    "Width": 0.12368916720151901,
    "Top": 0.16007372736930847,
    "Left": 0.5901257991790771,
    "Height": 0.25140416622161865
  },
  "Confidence": 100.0
}
}
```

Untuk informasi selengkapnya, lihat [Membandingkan Wajah dalam Gambar](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [CompareFaces](#) di Referensi AWS CLI Perintah.

## create-collection

Contoh kode berikut menunjukkan cara menggunakan `create-collection`.

Untuk informasi selengkapnya, lihat [Membuat koleksi](#).

### AWS CLI

Untuk membuat koleksi

`create-collection` Perintah berikut membuat koleksi dengan nama yang ditentukan.

```
aws rekognition create-collection \
  --collection-id "MyCollection"
```

Output:

```
{
  "CollectionArn": "aws:rekognition:us-west-2:123456789012:collection/
MyCollection",
  "FaceModelVersion": "4.0",
  "StatusCode": 200
}
```

Untuk informasi selengkapnya, lihat [Membuat Koleksi](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [CreateCollection](#) di Referensi AWS CLI Perintah.

## create-stream-processor

Contoh kode berikut menunjukkan cara menggunakan `create-stream-processor`.

### AWS CLI

Untuk membuat prosesor aliran baru

`create-stream-processor` Contoh berikut membuat prosesor aliran baru dengan konfigurasi yang ditentukan.

```
aws rekognition create-stream-processor --name my-stream-processor \
  --input '{"KinesisVideoStream":{"Arn":"arn:aws:kinesisvideo:us-
west-2:123456789012:stream/macwebcam/1530559711205"}}' \
  --stream-processor-output '{"KinesisDataStream":{"Arn":"arn:aws:kinesis:us-
west-2:123456789012:stream/AmazonRekognitionRekStream"}}' \
  --role-arn arn:aws:iam::123456789012:role/AmazonRekognitionDetect \
  --settings '{"FaceSearch":
{"CollectionId":"MyCollection","FaceMatchThreshold":85.5}}'
```

Output:

```
{
  "StreamProcessorArn": "arn:aws:rekognition:us-
west-2:123456789012:streamprocessor/my-stream-processor"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Video Streaming](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [CreateStreamProcessor](#) di Referensi AWS CLI Perintah.

## delete-collection

Contoh kode berikut menunjukkan cara menggunakan `delete-collection`.

Untuk informasi selengkapnya, lihat [Menghapus koleksi](#).

## AWS CLI

Untuk menghapus koleksi

`delete-collection` Perintah berikut menghapus koleksi yang ditentukan.

```
aws rekognition delete-collection \  
  --collection-id MyCollection
```

Output:

```
{  
  "StatusCode": 200  
}
```

Untuk informasi selengkapnya, lihat [Menghapus Koleksi di Panduan](#) Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DeleteCollection](#) di Referensi AWS CLI Perintah.

## `delete-faces`

Contoh kode berikut menunjukkan cara menggunakan `delete-faces`.

Untuk informasi selengkapnya, lihat [Menghapus wajah dari koleksi](#).

## AWS CLI

Untuk menghapus wajah dari koleksi

`delete-faces` Perintah berikut menghapus wajah yang ditentukan dari koleksi.

```
aws rekognition delete-faces \  
  --collection-id MyCollection  
  --face-ids '["0040279c-0178-436e-b70a-e61b074e96b0"]'
```

Output:

```
{  
  "DeletedFaces": [  
    "0040279c-0178-436e-b70a-e61b074e96b0"  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Menghapus Wajah dari Koleksi](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DeleteFaces](#) di Referensi AWS CLI Perintah.

## **delete-stream-processor**

Contoh kode berikut menunjukkan cara menggunakan `delete-stream-processor`.

### AWS CLI

Untuk menghapus prosesor aliran

`delete-stream-processor` Perintah berikut menghapus prosesor aliran yang ditentukan.

```
aws rekognition delete-stream-processor \
  --name my-stream-processor
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Video Streaming](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DeleteStreamProcessor](#) di Referensi AWS CLI Perintah.

## **describe-collection**

Contoh kode berikut menunjukkan cara menggunakan `describe-collection`.

Untuk informasi selengkapnya, lihat [Menjelaskan koleksi](#).

### AWS CLI

Untuk menggambarkan koleksi

`describe-collection` Contoh berikut menampilkan rincian tentang koleksi yang ditentukan.

```
aws rekognition describe-collection \
```

```
--collection-id MyCollection
```

#### Output:

```
{
  "FaceCount": 200,
  "CreationTimestamp": 1569444828.274,
  "CollectionARN": "arn:aws:rekognition:us-west-2:123456789012:collection/
MyCollection",
  "FaceModelVersion": "4.0"
}
```

Untuk informasi selengkapnya, lihat [Menjelaskan Koleksi di Panduan](#) Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DescribeCollection](#) di Referensi AWS CLI Perintah.

## describe-stream-processor

Contoh kode berikut menunjukkan cara menggunakan `describe-stream-processor`.

### AWS CLI

Untuk mendapatkan informasi tentang prosesor aliran

`describe-stream-processor` Perintah berikut menampilkan rincian tentang prosesor aliran yang ditentukan.

```
aws rekognition describe-stream-processor \
  --name my-stream-processor
```

#### Output:

```
{
  "Status": "STOPPED",
  "Name": "my-stream-processor",
  "LastUpdateTimestamp": 1532449292.712,
  "Settings": {
    "FaceSearch": {
      "FaceMatchThreshold": 80.0,

```

```

        "CollectionId": "my-collection"
    }
},
"RoleArn": "arn:aws:iam::123456789012:role/AmazonRekognitionDetectStream",
"StreamProcessorArn": "arn:aws:rekognition:us-
west-2:123456789012:streamprocessor/my-stream-processpr",
"Output": {
    "KinesisDataStream": {
        "Arn": "arn:aws:kinesis:us-west-2:123456789012:stream/
AmazonRekognitionRekStream"
    }
},
"Input": {
    "KinesisVideoStream": {
        "Arn": "arn:aws:kinesisvideo:us-west-2:123456789012:stream/
macwebcam/123456789012"
    }
},
"CreationTimestamp": 1532449292.712
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Video Streaming](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DescribeStreamProcessor](#) di Referensi AWS CLI Perintah.

## detect-faces

Contoh kode berikut menunjukkan cara menggunakan `detect-faces`.

Untuk informasi selengkapnya, lihat [Mendeteksi wajah dalam gambar](#).

### AWS CLI

Untuk mendeteksi wajah dalam gambar

`detect-faces` Perintah berikut mendeteksi wajah dalam gambar tertentu yang disimpan dalam bucket Amazon S3.

```

aws rekognition detect-faces \
  --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"MyFriend.jpg"}}' \
  --attributes "ALL"

```

## Output:

```
{
  "FaceDetails": [
    {
      "Confidence": 100.0,
      "Eyeglasses": {
        "Confidence": 98.91107940673828,
        "Value": false
      },
      "Sunglasses": {
        "Confidence": 99.7966537475586,
        "Value": false
      },
      "Gender": {
        "Confidence": 99.56611633300781,
        "Value": "Male"
      },
      "Landmarks": [
        {
          "Y": 0.26721030473709106,
          "X": 0.6204193830490112,
          "Type": "eyeLeft"
        },
        {
          "Y": 0.26831310987472534,
          "X": 0.6776827573776245,
          "Type": "eyeRight"
        },
        {
          "Y": 0.3514654338359833,
          "X": 0.6241428852081299,
          "Type": "mouthLeft"
        },
        {
          "Y": 0.35258132219314575,
          "X": 0.6713621020317078,
          "Type": "mouthRight"
        },
        {
          "Y": 0.3140771687030792,
          "X": 0.6428444981575012,
          "Type": "nose"
        }
      ]
    }
  ]
}
```

```
{
  "Y": 0.24662546813488007,
  "X": 0.6001564860343933,
  "Type": "leftEyeBrowLeft"
},
{
  "Y": 0.24326619505882263,
  "X": 0.6303644776344299,
  "Type": "leftEyeBrowRight"
},
{
  "Y": 0.23818562924861908,
  "X": 0.6146903038024902,
  "Type": "leftEyeBrowUp"
},
{
  "Y": 0.24373626708984375,
  "X": 0.6640064716339111,
  "Type": "rightEyeBrowLeft"
},
{
  "Y": 0.24877218902111053,
  "X": 0.7025929093360901,
  "Type": "rightEyeBrowRight"
},
{
  "Y": 0.23938551545143127,
  "X": 0.6823262572288513,
  "Type": "rightEyeBrowUp"
},
{
  "Y": 0.265746533870697,
  "X": 0.6112898588180542,
  "Type": "leftEyeLeft"
},
{
  "Y": 0.2676128149032593,
  "X": 0.6317071914672852,
  "Type": "leftEyeRight"
},
{
  "Y": 0.262735515832901,
  "X": 0.6201658248901367,
  "Type": "leftEyeUp"
}
```



```
    },
    {
      "Y": 0.27025148272514343,
      "X": 0.6206279993057251,
      "Type": "leftEyeDown"
    },
    {
      "Y": 0.268223375082016,
      "X": 0.6658390760421753,
      "Type": "rightEyeLeft"
    },
    {
      "Y": 0.2672517001628876,
      "X": 0.687832236289978,
      "Type": "rightEyeRight"
    },
    {
      "Y": 0.26383838057518005,
      "X": 0.6769183874130249,
      "Type": "rightEyeUp"
    },
    {
      "Y": 0.27138751745224,
      "X": 0.676596462726593,
      "Type": "rightEyeDown"
    },
    {
      "Y": 0.32283174991607666,
      "X": 0.6350004076957703,
      "Type": "noseLeft"
    },
    {
      "Y": 0.3219289481639862,
      "X": 0.6567046642303467,
      "Type": "noseRight"
    },
    {
      "Y": 0.3420318365097046,
      "X": 0.6450609564781189,
      "Type": "mouthUp"
    },
    {
      "Y": 0.3664324879646301,
      "X": 0.6455618143081665,
```

```
        "Type": "mouthDown"
    },
    {
        "Y": 0.26721030473709106,
        "X": 0.6204193830490112,
        "Type": "leftPupil"
    },
    {
        "Y": 0.26831310987472534,
        "X": 0.6776827573776245,
        "Type": "rightPupil"
    },
    {
        "Y": 0.26343393325805664,
        "X": 0.5946047306060791,
        "Type": "upperJawlineLeft"
    },
    {
        "Y": 0.3543180525302887,
        "X": 0.6044883728027344,
        "Type": "midJawlineLeft"
    },
    {
        "Y": 0.4084877669811249,
        "X": 0.6477024555206299,
        "Type": "chinBottom"
    },
    {
        "Y": 0.3562754988670349,
        "X": 0.707981526851654,
        "Type": "midJawlineRight"
    },
    {
        "Y": 0.26580461859703064,
        "X": 0.7234612107276917,
        "Type": "upperJawlineRight"
    }
],
"Pose": {
    "Yaw": -3.7351467609405518,
    "Roll": -0.10309021919965744,
    "Pitch": 0.8637830018997192
},
"Emotions": [
```

```
{
  "Confidence": 8.74203109741211,
  "Type": "SURPRISED"
},
{
  "Confidence": 2.501944065093994,
  "Type": "ANGRY"
},
{
  "Confidence": 0.7378743290901184,
  "Type": "DISGUSTED"
},
{
  "Confidence": 3.5296201705932617,
  "Type": "HAPPY"
},
{
  "Confidence": 1.7162904739379883,
  "Type": "SAD"
},
{
  "Confidence": 9.518536567687988,
  "Type": "CONFUSED"
},
{
  "Confidence": 0.45474427938461304,
  "Type": "FEAR"
},
{
  "Confidence": 72.79895782470703,
  "Type": "CALM"
}
],
"AgeRange": {
  "High": 48,
  "Low": 32
},
"EyesOpen": {
  "Confidence": 98.93987274169922,
  "Value": true
},
"BoundingBox": {
  "Width": 0.12368916720151901,
  "Top": 0.16007372736930847,
```

```
        "Left": 0.5901257991790771,  
        "Height": 0.25140416622161865  
    },  
    "Smile": {  
        "Confidence": 93.4493179321289,  
        "Value": false  
    },  
    "MouthOpen": {  
        "Confidence": 90.53053283691406,  
        "Value": false  
    },  
    "Quality": {  
        "Sharpness": 95.51618957519531,  
        "Brightness": 65.29893493652344  
    },  
    "Mustache": {  
        "Confidence": 89.85221099853516,  
        "Value": false  
    },  
    "Beard": {  
        "Confidence": 86.1991195678711,  
        "Value": true  
    }  
  }  
]  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Wajah dalam Gambar](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DetectFaces](#) di Referensi AWS CLI Perintah.

## **detect-labels**

Contoh kode berikut menunjukkan cara menggunakan `detect-labels`.

Untuk informasi selengkapnya, lihat [Mendeteksi label dalam gambar](#).

### AWS CLI

Untuk mendeteksi label dalam gambar

`detect-labels` Contoh berikut mendeteksi adegan dan objek dalam gambar yang disimpan dalam bucket Amazon S3.

```
aws rekognition detect-labels \  
  --image '{"S3Object":{"Bucket":"bucket","Name":"image"}}'
```

Output:

```
{  
  "Labels": [  
    {  
      "Instances": [],  
      "Confidence": 99.15271759033203,  
      "Parents": [  
        {  
          "Name": "Vehicle"  
        },  
        {  
          "Name": "Transportation"  
        }  
      ],  
      "Name": "Automobile"  
    },  
    {  
      "Instances": [],  
      "Confidence": 99.15271759033203,  
      "Parents": [  
        {  
          "Name": "Transportation"  
        }  
      ],  
      "Name": "Vehicle"  
    },  
    {  
      "Instances": [],  
      "Confidence": 99.15271759033203,  
      "Parents": [],  
      "Name": "Transportation"  
    },  
    {  
      "Instances": [  
        {  
          "BoundingBox": {
```

```
        "Width": 0.10616336017847061,  
        "Top": 0.5039216876029968,  
        "Left": 0.0037978808395564556,  
        "Height": 0.18528179824352264  
    },  
    "Confidence": 99.15271759033203  
},  
{  
    "BoundingBox": {  
        "Width": 0.2429988533258438,  
        "Top": 0.5251884460449219,  
        "Left": 0.7309805154800415,  
        "Height": 0.21577216684818268  
    },  
    "Confidence": 99.1286392211914  
},  
{  
    "BoundingBox": {  
        "Width": 0.14233611524105072,  
        "Top": 0.5333095788955688,  
        "Left": 0.6494812965393066,  
        "Height": 0.15528248250484467  
    },  
    "Confidence": 98.48368072509766  
},  
{  
    "BoundingBox": {  
        "Width": 0.11086395382881165,  
        "Top": 0.5354844927787781,  
        "Left": 0.10355594009160995,  
        "Height": 0.10271988064050674  
    },  
    "Confidence": 96.45606231689453  
},  
{  
    "BoundingBox": {  
        "Width": 0.06254628300666809,  
        "Top": 0.5573825240135193,  
        "Left": 0.46083059906959534,  
        "Height": 0.053911514580249786  
    },  
    "Confidence": 93.65448760986328  
},  
{
```

```
    "BoundingBox": {
      "Width": 0.10105438530445099,
      "Top": 0.534368634223938,
      "Left": 0.5743985772132874,
      "Height": 0.12226245552301407
    },
    "Confidence": 93.06217193603516
  },
  {
    "BoundingBox": {
      "Width": 0.056389667093753815,
      "Top": 0.5235804319381714,
      "Left": 0.9427769780158997,
      "Height": 0.17163699865341187
    },
    "Confidence": 92.6864013671875
  },
  {
    "BoundingBox": {
      "Width": 0.06003860384225845,
      "Top": 0.5441341400146484,
      "Left": 0.22409997880458832,
      "Height": 0.06737709045410156
    },
    "Confidence": 90.4227066040039
  },
  {
    "BoundingBox": {
      "Width": 0.02848697081208229,
      "Top": 0.5107086896896362,
      "Left": 0,
      "Height": 0.19150497019290924
    },
    "Confidence": 86.65286254882812
  },
  {
    "BoundingBox": {
      "Width": 0.04067881405353546,
      "Top": 0.5566273927688599,
      "Left": 0.316415935754776,
      "Height": 0.03428703173995018
    },
    "Confidence": 85.36471557617188
  },
}
```

```
    {
      "BoundingBox": {
        "Width": 0.043411049991846085,
        "Top": 0.5394920110702515,
        "Left": 0.18293385207653046,
        "Height": 0.0893595889210701
      },
      "Confidence": 82.21705627441406
    },
    {
      "BoundingBox": {
        "Width": 0.031183116137981415,
        "Top": 0.5579366683959961,
        "Left": 0.2853088080883026,
        "Height": 0.03989990055561066
      },
      "Confidence": 81.0157470703125
    },
    {
      "BoundingBox": {
        "Width": 0.031113790348172188,
        "Top": 0.5504819750785828,
        "Left": 0.2580395042896271,
        "Height": 0.056484755128622055
      },
      "Confidence": 56.13441467285156
    },
    {
      "BoundingBox": {
        "Width": 0.08586374670267105,
        "Top": 0.5438792705535889,
        "Left": 0.5128012895584106,
        "Height": 0.08550430089235306
      },
      "Confidence": 52.37760925292969
    }
  ],
  "Confidence": 99.15271759033203,
  "Parents": [
    {
      "Name": "Vehicle"
    },
    {
      "Name": "Transportation"
    }
  ]
}
```



```
    }
  ],
  "Name": "Car"
},
{
  "Instances": [],
  "Confidence": 98.9914321899414,
  "Parents": [],
  "Name": "Human"
},
{
  "Instances": [
    {
      "BoundingBox": {
        "Width": 0.19360728561878204,
        "Top": 0.35072067379951477,
        "Left": 0.43734854459762573,
        "Height": 0.2742200493812561
      },
      "Confidence": 98.9914321899414
    },
    {
      "BoundingBox": {
        "Width": 0.03801717236638069,
        "Top": 0.5010883808135986,
        "Left": 0.9155802130699158,
        "Height": 0.06597328186035156
      },
      "Confidence": 85.02790832519531
    }
  ],
  "Confidence": 98.9914321899414,
  "Parents": [],
  "Name": "Person"
},
{
  "Instances": [],
  "Confidence": 93.24951934814453,
  "Parents": [],
  "Name": "Machine"
},
{
  "Instances": [
    {
```

```
    "BoundingBox": {
      "Width": 0.03561960905790329,
      "Top": 0.6468243598937988,
      "Left": 0.7850857377052307,
      "Height": 0.08878646790981293
    },
    "Confidence": 93.24951934814453
  },
  {
    "BoundingBox": {
      "Width": 0.02217046171426773,
      "Top": 0.6149078607559204,
      "Left": 0.04757237061858177,
      "Height": 0.07136218994855881
    },
    "Confidence": 91.5025863647461
  },
  {
    "BoundingBox": {
      "Width": 0.016197510063648224,
      "Top": 0.6274210214614868,
      "Left": 0.6472989320755005,
      "Height": 0.04955997318029404
    },
    "Confidence": 85.14686584472656
  },
  {
    "BoundingBox": {
      "Width": 0.020207518711686134,
      "Top": 0.6348286867141724,
      "Left": 0.7295016646385193,
      "Height": 0.07059963047504425
    },
    "Confidence": 83.34547424316406
  },
  {
    "BoundingBox": {
      "Width": 0.020280985161662102,
      "Top": 0.6171894669532776,
      "Left": 0.08744934946298599,
      "Height": 0.05297485366463661
    },
    "Confidence": 79.9981460571289
  },
}
```

```
{
  "BoundingBox": {
    "Width": 0.018318990245461464,
    "Top": 0.623889148235321,
    "Left": 0.6836880445480347,
    "Height": 0.06730121374130249
  },
  "Confidence": 78.87144470214844
},
{
  "BoundingBox": {
    "Width": 0.021310249343514442,
    "Top": 0.6167286038398743,
    "Left": 0.004064912907779217,
    "Height": 0.08317798376083374
  },
  "Confidence": 75.89361572265625
},
{
  "BoundingBox": {
    "Width": 0.03604431077837944,
    "Top": 0.7030032277107239,
    "Left": 0.9254803657531738,
    "Height": 0.04569442570209503
  },
  "Confidence": 64.402587890625
},
{
  "BoundingBox": {
    "Width": 0.009834849275648594,
    "Top": 0.5821820497512817,
    "Left": 0.28094568848609924,
    "Height": 0.01964157074689865
  },
  "Confidence": 62.79907989501953
},
{
  "BoundingBox": {
    "Width": 0.01475677452981472,
    "Top": 0.6137543320655823,
    "Left": 0.5950819253921509,
    "Height": 0.039063986390829086
  },
  "Confidence": 59.40483474731445
}
```

```
    }
  ],
  "Confidence": 93.24951934814453,
  "Parents": [
    {
      "Name": "Machine"
    }
  ],
  "Name": "Wheel"
},
{
  "Instances": [],
  "Confidence": 92.61514282226562,
  "Parents": [],
  "Name": "Road"
},
{
  "Instances": [],
  "Confidence": 92.37877655029297,
  "Parents": [
    {
      "Name": "Person"
    }
  ],
  "Name": "Sport"
},
{
  "Instances": [],
  "Confidence": 92.37877655029297,
  "Parents": [
    {
      "Name": "Person"
    }
  ],
  "Name": "Sports"
},
{
  "Instances": [
    {
      "BoundingBox": {
        "Width": 0.12326609343290329,
        "Top": 0.6332163214683533,
        "Left": 0.44815489649772644,
        "Height": 0.058117982000112534
```

```
        },
        "Confidence": 92.37877655029297
    }
],
"Confidence": 92.37877655029297,
"Parents": [
    {
        "Name": "Person"
    },
    {
        "Name": "Sport"
    }
],
"Name": "Skateboard"
},
{
    "Instances": [],
    "Confidence": 90.62931060791016,
    "Parents": [
        {
            "Name": "Person"
        }
    ],
    "Name": "Pedestrian"
},
{
    "Instances": [],
    "Confidence": 88.81334686279297,
    "Parents": [],
    "Name": "Asphalt"
},
{
    "Instances": [],
    "Confidence": 88.81334686279297,
    "Parents": [],
    "Name": "Tarmac"
},
{
    "Instances": [],
    "Confidence": 88.23201751708984,
    "Parents": [],
    "Name": "Path"
},
{
```

```
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [],
    "Name": "Urban"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [
      {
        "Name": "Building"
      },
      {
        "Name": "Urban"
      }
    ],
    "Name": "Town"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [],
    "Name": "Building"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [
      {
        "Name": "Building"
      },
      {
        "Name": "Urban"
      }
    ],
    "Name": "City"
  },
  {
    "Instances": [],
    "Confidence": 78.37934875488281,
    "Parents": [
      {
        "Name": "Car"
      }
    ],
```

```
        {
            "Name": "Vehicle"
        },
        {
            "Name": "Transportation"
        }
    ],
    "Name": "Parking Lot"
},
{
    "Instances": [],
    "Confidence": 78.37934875488281,
    "Parents": [
        {
            "Name": "Car"
        },
        {
            "Name": "Vehicle"
        },
        {
            "Name": "Transportation"
        }
    ],
    "Name": "Parking"
},
{
    "Instances": [],
    "Confidence": 74.37590026855469,
    "Parents": [
        {
            "Name": "Building"
        },
        {
            "Name": "Urban"
        },
        {
            "Name": "City"
        }
    ],
    "Name": "Downtown"
},
{
    "Instances": [],
    "Confidence": 69.84622955322266,
```

```
    "Parents": [
      {
        "Name": "Road"
      }
    ],
    "Name": "Intersection"
  },
  {
    "Instances": [],
    "Confidence": 57.68518829345703,
    "Parents": [
      {
        "Name": "Sports Car"
      },
      {
        "Name": "Car"
      },
      {
        "Name": "Vehicle"
      },
      {
        "Name": "Transportation"
      }
    ],
    "Name": "Coupe"
  },
  {
    "Instances": [],
    "Confidence": 57.68518829345703,
    "Parents": [
      {
        "Name": "Car"
      },
      {
        "Name": "Vehicle"
      },
      {
        "Name": "Transportation"
      }
    ],
    "Name": "Sports Car"
  },
  {
    "Instances": [],
```



```
    "Confidence": 56.59492111206055,
    "Parents": [
      {
        "Name": "Path"
      }
    ],
    "Name": "Sidewalk"
  },
  {
    "Instances": [],
    "Confidence": 56.59492111206055,
    "Parents": [
      {
        "Name": "Path"
      }
    ],
    "Name": "Pavement"
  },
  {
    "Instances": [],
    "Confidence": 55.58770751953125,
    "Parents": [
      {
        "Name": "Building"
      },
      {
        "Name": "Urban"
      }
    ],
    "Name": "Neighborhood"
  }
],
"LabelModelVersion": "2.0"
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Label dalam Gambar](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [DetectLabels](#) di Referensi AWS CLI Perintah.

## **detect-moderation-labels**

Contoh kode berikut menunjukkan cara menggunakan `detect-moderation-labels`.

Untuk informasi selengkapnya, lihat [Mendeteksi gambar yang tidak pantas](#).

## AWS CLI

Untuk mendeteksi konten yang tidak aman dalam gambar

`detect-moderation-labels` Perintah berikut mendeteksi konten yang tidak aman dalam gambar tertentu yang disimpan dalam bucket Amazon S3.

```
aws rekognition detect-moderation-labels \  
  --image "S3object={Bucket=MyImageS3Bucket,Name=gun.jpg}"
```

Output:

```
{  
  "ModerationModelVersion": "3.0",  
  "ModerationLabels": [  
    {  
      "Confidence": 97.29618072509766,  
      "ParentName": "Violence",  
      "Name": "Weapon Violence"  
    },  
    {  
      "Confidence": 97.29618072509766,  
      "ParentName": "",  
      "Name": "Violence"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Gambar Tidak Aman di Panduan Pengembang Rekognition Amazon](#).

- Untuk detail API, lihat [DetectModerationLabels](#) di Referensi AWS CLI Perintah.

## **detect-text**

Contoh kode berikut menunjukkan cara menggunakan `detect-text`.

Untuk informasi selengkapnya, lihat [Mendeteksi teks dalam gambar](#).

## AWS CLI

Untuk mendeteksi teks dalam gambar

`detect-text` Perintah berikut mendeteksi teks dalam gambar yang ditentukan.

```
aws rekognition detect-text \  
  --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"ExamplePicture.jpg"}}'
```

Output:

```
{  
  "TextDetections": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 0.24624845385551453,  
          "Top": 0.28288066387176514,  
          "Left": 0.391388863325119,  
          "Height": 0.022687450051307678  
        },  
        "Polygon": [  
          {  
            "Y": 0.28288066387176514,  
            "X": 0.391388863325119  
          },  
          {  
            "Y": 0.2826388478279114,  
            "X": 0.6376373171806335  
          },  
          {  
            "Y": 0.30532628297805786,  
            "X": 0.637677013874054  
          },  
          {  
            "Y": 0.305568128824234,  
            "X": 0.39142853021621704  
          }  
        ]  
      },  
      "Confidence": 94.35709381103516,  
      "DetectedText": "ESTD 1882",  
      "Type": "LINE",  
    }  
  ]  
}
```

```
    "Id": 0
  },
  {
    "Geometry": {
      "BoundingBox": {
        "Width": 0.33933889865875244,
        "Top": 0.32603850960731506,
        "Left": 0.34534579515457153,
        "Height": 0.07126858830451965
      },
      "Polygon": [
        {
          "Y": 0.32603850960731506,
          "X": 0.34534579515457153
        },
        {
          "Y": 0.32633158564567566,
          "X": 0.684684693813324
        },
        {
          "Y": 0.3976001739501953,
          "X": 0.684575080871582
        },
        {
          "Y": 0.3973070979118347,
          "X": 0.345236212015152
        }
      ]
    },
    "Confidence": 99.95779418945312,
    "DetectedText": "BRAINS",
    "Type": "LINE",
    "Id": 1
  },
  {
    "Confidence": 97.22098541259766,
    "Geometry": {
      "BoundingBox": {
        "Width": 0.061079490929841995,
        "Top": 0.2843210697174072,
        "Left": 0.391391396522522,
        "Height": 0.021029088646173477
      },
      "Polygon": [
```

```
        {
            "Y": 0.2843210697174072,
            "X": 0.391391396522522
        },
        {
            "Y": 0.2828207015991211,
            "X": 0.4524524509906769
        },
        {
            "Y": 0.3038259446620941,
            "X": 0.4534534513950348
        },
        {
            "Y": 0.30532634258270264,
            "X": 0.3923923969268799
        }
    ]
},
"DetectedText": "ESTD",
"ParentId": 0,
"Type": "WORD",
"Id": 2
},
{
    "Confidence": 91.49320983886719,
    "Geometry": {
        "BoundingBox": {
            "Width": 0.07007007300853729,
            "Top": 0.2828207015991211,
            "Left": 0.5675675868988037,
            "Height": 0.02250562608242035
        },
        "Polygon": [
            {
                "Y": 0.2828207015991211,
                "X": 0.5675675868988037
            },
            {
                "Y": 0.2828207015991211,
                "X": 0.6376376152038574
            },
            {
                "Y": 0.30532634258270264,
                "X": 0.6376376152038574
            }
        ]
    }
}
```

```
    },
    {
      "Y": 0.30532634258270264,
      "X": 0.5675675868988037
    }
  ]
},
"DetectedText": "1882",
"ParentId": 0,
"Type": "WORD",
"Id": 3
},
{
  "Confidence": 99.95779418945312,
  "Geometry": {
    "BoundingBox": {
      "Width": 0.33933934569358826,
      "Top": 0.32633158564567566,
      "Left": 0.3453453481197357,
      "Height": 0.07127484679222107
    },
    "Polygon": [
      {
        "Y": 0.32633158564567566,
        "X": 0.3453453481197357
      },
      {
        "Y": 0.32633158564567566,
        "X": 0.684684693813324
      },
      {
        "Y": 0.39759939908981323,
        "X": 0.6836836934089661
      },
      {
        "Y": 0.39684921503067017,
        "X": 0.3453453481197357
      }
    ]
  },
  "DetectedText": "BRAINS",
  "ParentId": 1,
  "Type": "WORD",
  "Id": 4
}
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [DetectText](#) di Referensi AWS CLI Perintah.

## disassociate-faces

Contoh kode berikut menunjukkan cara menggunakan `disassociate-faces`.

### AWS CLI

```
aws rekognition disassociate-faces --face-ids list-of-face-ids  
  --user-id user-id --collection-id collection-name --region region-name
```

- Untuk detail API, lihat [DisassociateFaces](#) di Referensi AWS CLI Perintah.

## get-celebrity-info

Contoh kode berikut menunjukkan cara menggunakan `get-celebrity-info`.

### AWS CLI

Untuk mendapatkan informasi tentang selebriti

`get-celebrity-info` Perintah berikut menampilkan informasi tentang selebriti yang ditentukan. `id` Parameter berasal dari panggilan sebelumnya `recognize-celebrities`.

```
aws rekognition get-celebrity-info --id nnnnnnn
```

Output:

```
{  
  "Name": "Celeb A",  
  "Urls": [  
    "www.imdb.com/name/aaaaaaaaa"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan Informasi Tentang Selebriti](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [GetCelebrityInfo](#) di Referensi AWS CLI Perintah.

## get-celebrity-recognition

Contoh kode berikut menunjukkan cara menggunakan `get-celebrity-recognition`.

### AWS CLI

Untuk mendapatkan hasil operasi pengakuan selebriti

`get-celebrity-recognition` Perintah berikut menampilkan hasil operasi pengenalan selebriti yang Anda mulai sebelumnya dengan menelepon `start-celebrity-recognition`

```
aws rekognition get-celebrity-recognition \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef
```

### Output:

```
{
  "NextToken": "3D01Clx1CiT31VsRDkA03IybLb/h5AtDWSGuhYi
+N1FIJwwPtAkuKzDhL2rV3GcwmNt77+12",
  "Celebrities": [
    {
      "Timestamp": 0,
      "Celebrity": {
        "Confidence": 96.0,
        "Face": {
          "BoundingBox": {
            "Width": 0.70333331823349,
            "Top": 0.16750000417232513,
            "Left": 0.19555555284023285,
            "Height": 0.3956249952316284
          },
          "Landmarks": [
            {
              "Y": 0.31031012535095215,
              "X": 0.441436767578125,
              "Type": "eyeLeft"
            },
            {
```



```

        "Y": 0.3081788718700409,
        "X": 0.6437258720397949,
        "Type": "eyeRight"
    },
    {
        "Y": 0.39542075991630554,
        "X": 0.5572493076324463,
        "Type": "nose"
    },
    {
        "Y": 0.4597957134246826,
        "X": 0.4579732120037079,
        "Type": "mouthLeft"
    },
    {
        "Y": 0.45688048005104065,
        "X": 0.6349081993103027,
        "Type": "mouthRight"
    }
],
"Pose": {
    "Yaw": 8.943398475646973,
    "Roll": -2.0309247970581055,
    "Pitch": -0.5674862861633301
},
"Quality": {
    "Sharpness": 99.40211486816406,
    "Brightness": 89.47132110595703
},
"Confidence": 99.99861145019531
},
"Name": "CelebrityA",
"Urls": [
    "www.imdb.com/name/111111111"
],
"Id": "nnnnnn"
}
},
{
    "Timestamp": 467,
    "Celebrity": {
        "Confidence": 99.0,
        "Face": {
            "BoundingBox": {

```

```
    "Width": 0.6877777576446533,  
    "Top": 0.18437500298023224,  
    "Left": 0.20555555820465088,  
    "Height": 0.3868750035762787  
  },  
  "Landmarks": [  
    {  
      "Y": 0.31895750761032104,  
      "X": 0.4411413371562958,  
      "Type": "eyeLeft"  
    },  
    {  
      "Y": 0.3140959143638611,  
      "X": 0.6523157954216003,  
      "Type": "eyeRight"  
    },  
    {  
      "Y": 0.4016456604003906,  
      "X": 0.5682755708694458,  
      "Type": "nose"  
    },  
    {  
      "Y": 0.46894142031669617,  
      "X": 0.4597797095775604,  
      "Type": "mouthLeft"  
    },  
    {  
      "Y": 0.46971091628074646,  
      "X": 0.6286435127258301,  
      "Type": "mouthRight"  
    }  
  ],  
  "Pose": {  
    "Yaw": 10.433465957641602,  
    "Roll": -3.347442388534546,  
    "Pitch": 1.3709543943405151  
  },  
  "Quality": {  
    "Sharpness": 99.5531005859375,  
    "Brightness": 88.5764389038086  
  },  
  "Confidence": 99.99148559570312  
},  
"Name": "Jane Celebrity",
```

```

        "Urls": [
            "www.imdb.com/name/111111111"
        ],
        "Id": "nnnnnn"
    }
}
],
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
    "Format": "QuickTime / MOV",
    "FrameRate": 29.978118896484375,
    "Codec": "h264",
    "DurationMillis": 4570,
    "FrameHeight": 1920,
    "FrameWidth": 1080
}
}

```

Untuk informasi selengkapnya, lihat [Mengenali Selebriti dalam Video Tersimpan](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [GetCelebrityRecognition](#) di Referensi AWS CLI Perintah.

## get-content-moderation

Contoh kode berikut menunjukkan cara menggunakan `get-content-moderation`.

### AWS CLI

Untuk mendapatkan hasil operasi konten yang tidak aman

`get-content-moderation` Perintah berikut menampilkan hasil operasi konten tidak aman yang Anda mulai sebelumnya dengan menelepon `start-content-moderation`.

```

aws rekognition get-content-moderation \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef

```

### Output:

```

{
  "NextToken": "d1hcKMHMzpCBGFukz6I03JMcWiJAamCVhXHt3r6b4b5Tfbyw3q7o+Jeezt
+Zpgf0nW9FCCgQ",
  "ModerationLabels": [

```

```
{
  "Timestamp": 0,
  "ModerationLabel": {
    "Confidence": 97.39583587646484,
    "ParentName": "",
    "Name": "Violence"
  }
},
{
  "Timestamp": 0,
  "ModerationLabel": {
    "Confidence": 97.39583587646484,
    "ParentName": "Violence",
    "Name": "Weapon Violence"
  }
}
],
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.97515869140625,
  "Codec": "h264",
  "DurationMillis": 6039,
  "FrameHeight": 1920,
  "FrameWidth": 1080
}
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Video Tersimpan Tidak Aman di Panduan Pengembang Rekognition Amazon](#).

- Untuk detail API, lihat [GetContentModeration](#) di Referensi AWS CLI Perintah.

## get-face-detection

Contoh kode berikut menunjukkan cara menggunakan `get-face-detection`.

### AWS CLI

Untuk mendapatkan hasil operasi deteksi wajah

`get-face-detection` Perintah berikut menampilkan hasil operasi deteksi wajah yang Anda mulai sebelumnya dengan menelepon `start-face-detection`.

```
aws rekognition get-face-detection \  
--job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

**Output:**

```
{  
  "Faces": [  
    {  
      "Timestamp": 467,  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.1560753583908081,  
          "Top": 0.13555361330509186,  
          "Left": -0.0952017530798912,  
          "Height": 0.6934483051300049  
        },  
        "Landmarks": [  
          {  
            "Y": 0.4013825058937073,  
            "X": -0.041750285774469376,  
            "Type": "eyeLeft"  
          },  
          {  
            "Y": 0.41695496439933777,  
            "X": 0.027979329228401184,  
            "Type": "eyeRight"  
          },  
          {  
            "Y": 0.6375303268432617,  
            "X": -0.04034662991762161,  
            "Type": "mouthLeft"  
          },  
          {  
            "Y": 0.6497718691825867,  
            "X": 0.013960429467260838,  
            "Type": "mouthRight"  
          },  
          {  
            "Y": 0.5238034129142761,  
            "X": 0.008022055961191654,  
            "Type": "nose"  
          }  
        ]  
      }  
    ],  
  ]  
}
```

```
    "Pose": {
      "Yaw": -58.07863998413086,
      "Roll": 1.9384294748306274,
      "Pitch": -24.66305160522461
    },
    "Quality": {
      "Sharpness": 83.14741516113281,
      "Brightness": 25.75942611694336
    },
    "Confidence": 87.7622299194336
  }
},
{
  "Timestamp": 967,
  "Face": {
    "BoundingBox": {
      "Width": 0.28559377789497375,
      "Top": 0.19436298310756683,
      "Left": 0.024553587660193443,
      "Height": 0.7216082215309143
    },
    "Landmarks": [
      {
        "Y": 0.4650231599807739,
        "X": 0.16269078850746155,
        "Type": "eyeLeft"
      },
      {
        "Y": 0.4843238294124603,
        "X": 0.2782580852508545,
        "Type": "eyeRight"
      },
      {
        "Y": 0.71530681848526,
        "X": 0.1741468608379364,
        "Type": "mouthLeft"
      },
      {
        "Y": 0.7310671210289001,
        "X": 0.26857468485832214,
        "Type": "mouthRight"
      },
      {
        "Y": 0.582602322101593,
```

```
        "X": 0.2566150426864624,
        "Type": "nose"
      }
    ],
    "Pose": {
      "Yaw": 11.487052917480469,
      "Roll": 5.074230670928955,
      "Pitch": 15.396159172058105
    },
    "Quality": {
      "Sharpness": 73.32209777832031,
      "Brightness": 54.96497344970703
    },
    "Confidence": 99.99998474121094
  }
}
],
"NextToken":
"OzL223pDKy91160/02KXRqFIEAwxy4PkgYcm3hSo0rdysbXg5Ex0eFgTGEj0ADEac6S037U",
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.970617294311523,
  "Codec": "h264",
  "DurationMillis": 6806,
  "FrameHeight": 1080,
  "FrameWidth": 1920
}
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Wajah dalam Video Tersimpan](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [GetFaceDetection](#) di Referensi AWS CLI Perintah.

## get-face-search

Contoh kode berikut menunjukkan cara menggunakan `get-face-search`.

### AWS CLI

Untuk mendapatkan hasil operasi pencarian wajah

`get-face-search`Perintah berikut menampilkan hasil operasi pencarian wajah yang Anda mulai sebelumnya dengan menelepon `start-face-search`.

```
aws rekognition get-face-search \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
  "Persons": [
    {
      "Timestamp": 467,
      "FaceMatches": [],
      "Person": {
        "Index": 0,
        "Face": {
          "BoundingBox": {
            "Width": 0.1560753583908081,
            "Top": 0.13555361330509186,
            "Left": -0.0952017530798912,
            "Height": 0.6934483051300049
          },
          "Landmarks": [
            {
              "Y": 0.4013825058937073,
              "X": -0.041750285774469376,
              "Type": "eyeLeft"
            },
            {
              "Y": 0.41695496439933777,
              "X": 0.027979329228401184,
              "Type": "eyeRight"
            },
            {
              "Y": 0.6375303268432617,
              "X": -0.04034662991762161,
              "Type": "mouthLeft"
            },
            {
              "Y": 0.6497718691825867,
              "X": 0.013960429467260838,
              "Type": "mouthRight"
            }
          ]
        }
      }
    }
  ]
}
```



```
        {
            "Y": 0.5238034129142761,
            "X": 0.008022055961191654,
            "Type": "nose"
        }
    ],
    "Pose": {
        "Yaw": -58.07863998413086,
        "Roll": 1.9384294748306274,
        "Pitch": -24.66305160522461
    },
    "Quality": {
        "Sharpness": 83.14741516113281,
        "Brightness": 25.75942611694336
    },
    "Confidence": 87.7622299194336
}
},
{
    "Timestamp": 967,
    "FaceMatches": [
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.12368900328874588,
                    "Top": 0.16007399559020996,
                    "Left": 0.5901259779930115,
                    "Height": 0.2514039874076843
                },
                "FaceId": "056a95fa-2060-4159-9cab-7ed4daa030fa",
                "ExternalImageId": "image3.jpg",
                "Confidence": 100.0,
                "ImageId": "08f8a078-8929-37fd-8e8f-aadf690e8232"
            },
            "Similarity": 98.44476318359375
        }
    ],
    "Person": {
        "Index": 1,
        "Face": {
            "BoundingBox": {
                "Width": 0.28559377789497375,
                "Top": 0.19436298310756683,
```

```
        "Left": 0.024553587660193443,  
        "Height": 0.7216082215309143  
    },  
    "Landmarks": [  
        {  
            "Y": 0.4650231599807739,  
            "X": 0.16269078850746155,  
            "Type": "eyeLeft"  
        },  
        {  
            "Y": 0.4843238294124603,  
            "X": 0.2782580852508545,  
            "Type": "eyeRight"  
        },  
        {  
            "Y": 0.71530681848526,  
            "X": 0.1741468608379364,  
            "Type": "mouthLeft"  
        },  
        {  
            "Y": 0.7310671210289001,  
            "X": 0.26857468485832214,  
            "Type": "mouthRight"  
        },  
        {  
            "Y": 0.582602322101593,  
            "X": 0.2566150426864624,  
            "Type": "nose"  
        }  
    ],  
    "Pose": {  
        "Yaw": 11.487052917480469,  
        "Roll": 5.074230670928955,  
        "Pitch": 15.396159172058105  
    },  
    "Quality": {  
        "Sharpness": 73.32209777832031,  
        "Brightness": 54.96497344970703  
    },  
    "Confidence": 99.99998474121094  
    }  
} ]
```

```
"NextToken": "5bkgcezyuaqhtWk3C80TW6cjRghrwV9XDMivm5B3MXm+Lv6G+L+GejyFHPHoNa/ldXIC4c/d",
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.970617294311523,
  "Codec": "h264",
  "DurationMillis": 6806,
  "FrameHeight": 1080,
  "FrameWidth": 1920
}
}
```

Untuk informasi selengkapnya, lihat [Mencari Video Tersimpan untuk Wajah](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [GetFaceSearch](#) di Referensi AWS CLI Perintah.

## get-label-detection

Contoh kode berikut menunjukkan cara menggunakan `get-label-detection`.

### AWS CLI

Untuk mendapatkan hasil operasi deteksi objek dan adegan

`get-label-detection` Perintah berikut menampilkan hasil operasi deteksi objek dan adegan yang Anda mulai sebelumnya dengan menelepon `start-label-detection`.

```
aws rekognition get-label-detection \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
  "Labels": [
    {
      "Timestamp": 0,
      "Label": {
        "Instances": [],
        "Confidence": 50.19071578979492,
        "Parents": [
          {
```

```
        "Name": "Person"
      },
      {
        "Name": "Crowd"
      }
    ],
    "Name": "Audience"
  }
},
{
  "Timestamp": 0,
  "Label": {
    "Instances": [],
    "Confidence": 55.74115753173828,
    "Parents": [
      {
        "Name": "Room"
      },
      {
        "Name": "Indoors"
      },
      {
        "Name": "School"
      }
    ],
    "Name": "Classroom"
  }
}
],
"JobStatus": "SUCCEEDED",
"LabelModelVersion": "2.0",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.970617294311523,
  "Codec": "h264",
  "DurationMillis": 6806,
  "FrameHeight": 1080,
  "FrameWidth": 1920
},
"NextToken": "BMugzAi4L72IERzQdbpyMQuEFBsJlo5W0Yx3mfG+sR9mm98E1/
Cp0benspRfs/5FBQFs4X7G"
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Label dalam Video](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [GetLabelDetection](#) di Referensi AWS CLI Perintah.

## get-person-tracking

Contoh kode berikut menunjukkan cara menggunakan `get-person-tracking`.

### AWS CLI

Untuk mendapatkan hasil dari operasi pathing orang

`get-person-tracking` Perintah berikut menampilkan hasil operasi pathing orang yang Anda mulai sebelumnya dengan menelepon `start-person-tracking`.

```
aws rekognition get-person-tracking \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
  "Persons": [
    {
      "Timestamp": 500,
      "Person": {
        "BoundingBox": {
          "Width": 0.4151041805744171,
          "Top": 0.07870370149612427,
          "Left": 0.0,
          "Height": 0.9212962985038757
        },
        "Index": 0
      }
    },
    {
      "Timestamp": 567,
      "Person": {
        "BoundingBox": {
          "Width": 0.4755208194255829,
          "Top": 0.07777778059244156,
          "Left": 0.0,
          "Height": 0.91944444417953491
        }
      }
    }
  ]
}
```

```

        },
        "Index": 0
    }
}
],
"NextToken": "D/vRIYNyhG79ugdta3f+8cRg9oSRo
+HigG0uxRiYpTn0ExnqTi1CJektVAc4HrAXDv25eHYk",
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
    "Format": "QuickTime / MOV",
    "FrameRate": 29.970617294311523,
    "Codec": "h264",
    "DurationMillis": 6806,
    "FrameHeight": 1080,
    "FrameWidth": 1920
}
}
}

```

Untuk informasi selengkapnya, lihat [People Pathing](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [GetPersonTracking](#) di Referensi AWS CLI Perintah.

## index-faces

Contoh kode berikut menunjukkan cara menggunakan `index-faces`.

Untuk informasi selengkapnya, lihat [Menambahkan wajah ke koleksi](#).

### AWS CLI

Untuk menambahkan wajah ke koleksi

`index-faces` Perintah berikut menambahkan wajah yang ditemukan dalam gambar ke koleksi yang ditentukan.

```

aws rekognition index-faces \
  --image '{"S3Object":{"Bucket":"MyVideoS3Bucket","Name":"MyPicture.jpg"}}' \
  --collection-id MyCollection \
  --max-faces 1 \
  --quality-filter "AUTO" \
  --detection-attributes "ALL" \
  --external-image-id "MyPicture.jpg"

```

## Output:

```
{
  "FaceRecords": [
    {
      "FaceDetail": {
        "Confidence": 99.993408203125,
        "Eyeglasses": {
          "Confidence": 99.11750030517578,
          "Value": false
        },
        "Sunglasses": {
          "Confidence": 99.98249053955078,
          "Value": false
        },
        "Gender": {
          "Confidence": 99.92769622802734,
          "Value": "Male"
        },
        "Landmarks": [
          {
            "Y": 0.26750367879867554,
            "X": 0.6202793717384338,
            "Type": "eyeLeft"
          },
          {
            "Y": 0.26642778515815735,
            "X": 0.6787431836128235,
            "Type": "eyeRight"
          },
          {
            "Y": 0.31361380219459534,
            "X": 0.6421601176261902,
            "Type": "nose"
          },
          {
            "Y": 0.3495299220085144,
            "X": 0.6216195225715637,
            "Type": "mouthLeft"
          },
          {
            "Y": 0.35194727778434753,
            "X": 0.669899046421051,
            "Type": "mouthRight"
          }
        ]
      }
    }
  ]
}
```

```
    },  
    {  
      "Y": 0.26844894886016846,  
      "X": 0.6210268139839172,  
      "Type": "leftPupil"  
    },  
    {  
      "Y": 0.26707562804222107,  
      "X": 0.6817160844802856,  
      "Type": "rightPupil"  
    },  
    {  
      "Y": 0.24834522604942322,  
      "X": 0.6018546223640442,  
      "Type": "leftEyeBrowLeft"  
    },  
    {  
      "Y": 0.24397172033786774,  
      "X": 0.6172008514404297,  
      "Type": "leftEyeBrowUp"  
    },  
    {  
      "Y": 0.24677404761314392,  
      "X": 0.6339119076728821,  
      "Type": "leftEyeBrowRight"  
    },  
    {  
      "Y": 0.24582654237747192,  
      "X": 0.6619398593902588,  
      "Type": "rightEyeBrowLeft"  
    },  
    {  
      "Y": 0.23973053693771362,  
      "X": 0.6804757118225098,  
      "Type": "rightEyeBrowUp"  
    },  
    {  
      "Y": 0.24441994726657867,  
      "X": 0.6978968977928162,  
      "Type": "rightEyeBrowRight"  
    },  
    {  
      "Y": 0.2695908546447754,  
      "X": 0.6085202693939209,
```



```
    "Type": "leftEyeLeft"
  },
  {
    "Y": 0.26716896891593933,
    "X": 0.6315826177597046,
    "Type": "leftEyeRight"
  },
  {
    "Y": 0.26289820671081543,
    "X": 0.6202316880226135,
    "Type": "leftEyeUp"
  },
  {
    "Y": 0.27123287320137024,
    "X": 0.6205548048019409,
    "Type": "leftEyeDown"
  },
  {
    "Y": 0.2668408751487732,
    "X": 0.6663622260093689,
    "Type": "rightEyeLeft"
  },
  {
    "Y": 0.26741549372673035,
    "X": 0.6910083889961243,
    "Type": "rightEyeRight"
  },
  {
    "Y": 0.2614026665687561,
    "X": 0.6785826086997986,
    "Type": "rightEyeUp"
  },
  {
    "Y": 0.27075251936912537,
    "X": 0.6789616942405701,
    "Type": "rightEyeDown"
  },
  {
    "Y": 0.3211299479007721,
    "X": 0.6324167847633362,
    "Type": "noseLeft"
  },
  {
    "Y": 0.32276326417922974,
```

```
        "X": 0.6558475494384766,  
        "Type": "noseRight"  
    },  
    {  
        "Y": 0.34385165572166443,  
        "X": 0.6444970965385437,  
        "Type": "mouthUp"  
    },  
    {  
        "Y": 0.3671635091304779,  
        "X": 0.6459195017814636,  
        "Type": "mouthDown"  
    }  
],  
"Pose": {  
    "Yaw": -9.54541015625,  
    "Roll": -0.5709401965141296,  
    "Pitch": 0.6045494675636292  
},  
"Emotions": [  
    {  
        "Confidence": 39.90074157714844,  
        "Type": "HAPPY"  
    },  
    {  
        "Confidence": 23.38753890991211,  
        "Type": "CALM"  
    },  
    {  
        "Confidence": 5.840933322906494,  
        "Type": "CONFUSED"  
    }  
],  
"AgeRange": {  
    "High": 63,  
    "Low": 45  
},  
"EyesOpen": {  
    "Confidence": 99.80887603759766,  
    "Value": true  
},  
"BoundingBox": {  
    "Width": 0.18562500178813934,  
    "Top": 0.1618015021085739,
```

```
        "Left": 0.5575000047683716,  
        "Height": 0.24770642817020416  
    },  
    "Smile": {  
        "Confidence": 99.69740295410156,  
        "Value": false  
    },  
    "MouthOpen": {  
        "Confidence": 99.97393798828125,  
        "Value": false  
    },  
    "Quality": {  
        "Sharpness": 95.54405975341797,  
        "Brightness": 63.867706298828125  
    },  
    "Mustache": {  
        "Confidence": 97.05007934570312,  
        "Value": false  
    },  
    "Beard": {  
        "Confidence": 87.34505462646484,  
        "Value": false  
    }  
},  
"Face": {  
    "BoundingBox": {  
        "Width": 0.18562500178813934,  
        "Top": 0.1618015021085739,  
        "Left": 0.5575000047683716,  
        "Height": 0.24770642817020416  
    },  
    "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",  
    "ExternalImageId": "example-image.jpg",  
    "Confidence": 99.993408203125,  
    "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"  
}  
}  
],  
"UnindexedFaces": [],  
"FaceModelVersion": "3.0",  
"OrientationCorrection": "ROTATE_0"  
}
```

Untuk informasi selengkapnya, lihat [Menambahkan Wajah ke Koleksi](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [IndexFaces](#) di Referensi AWS CLI Perintah.

## list-collections

Contoh kode berikut menunjukkan cara menggunakan `list-collections`.

Untuk informasi selengkapnya, lihat [Daftar koleksi](#).

### AWS CLI

Untuk daftar koleksi yang tersedia

`list-collections` Perintah berikut mencantumkan koleksi yang tersedia di AWS akun.

```
aws rekognition list-collections
```

Output:

```
{
  "FaceModelVersions": [
    "2.0",
    "3.0",
    "3.0",
    "3.0",
    "4.0",
    "1.0",
    "3.0",
    "4.0",
    "4.0",
    "4.0"
  ],
  "CollectionIds": [
    "MyCollection1",
    "MyCollection2",
    "MyCollection3",
    "MyCollection4",
    "MyCollection5",
    "MyCollection6",
    "MyCollection7",
    "MyCollection8",
  ]
}
```

```
    "MyCollection9",  
    "MyCollection10"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Daftar Koleksi](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [ListCollections](#) di Referensi AWS CLI Perintah.

## list-faces

Contoh kode berikut menunjukkan cara menggunakan `list-faces`.

Untuk informasi selengkapnya, lihat [Daftar wajah dalam koleksi](#).

### AWS CLI

Untuk membuat daftar wajah dalam koleksi

`list-faces` Perintah berikut mencantumkan wajah-wajah dalam koleksi yang ditentukan.

```
aws rekognition list-faces \  
  --collection-id MyCollection
```

### Output:

```
{  
  "FaceModelVersion": "3.0",  
  "Faces": [  
    {  
      "BoundingBox": {  
        "Width": 0.5216310024261475,  
        "Top": 0.3256250023841858,  
        "Left": 0.13394300639629364,  
        "Height": 0.3918749988079071  
      },  
      "FaceId": "0040279c-0178-436e-b70a-e61b074e96b0",  
      "ExternalImageId": "image1.jpg",  
      "Confidence": 100.0,  
      "ImageId": "f976e487-3719-5e2d-be8b-ea2724c26991"  
    },  
    {
```

```
"BoundingBox": {
  "Width": 0.5074880123138428,
  "Top": 0.3774999976158142,
  "Left": 0.18302799761295319,
  "Height": 0.3812499940395355
},
"FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
"ExternalImageId": "image2.jpg",
"Confidence": 99.99930572509766,
"ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
},
{
  "BoundingBox": {
    "Width": 0.5574039816856384,
    "Top": 0.37187498807907104,
    "Left": 0.14559100568294525,
    "Height": 0.4181250035762787
  },
  "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
  "ExternalImageId": "image3.jpg",
  "Confidence": 99.99960327148438,
  "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
},
{
  "BoundingBox": {
    "Width": 0.18562500178813934,
    "Top": 0.1618019938468933,
    "Left": 0.5575000047683716,
    "Height": 0.24770599603652954
  },
  "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
  "ExternalImageId": "image4.jpg",
  "Confidence": 99.99340057373047,
  "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
},
{
  "BoundingBox": {
    "Width": 0.5307819843292236,
    "Top": 0.2862499952316284,
    "Left": 0.1564060002565384,
    "Height": 0.3987500071525574
  },
  "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
  "ExternalImageId": "image5.jpg",
```

```
"Confidence": 99.99970245361328,  
"ImageId": "3c314792-197d-528d-bbb6-798ed012c150"  
},  
{  
  "BoundingBox": {  
    "Width": 0.5773710012435913,  
    "Top": 0.34437501430511475,  
    "Left": 0.12396000325679779,  
    "Height": 0.4337500035762787  
  },  
  "FaceId": "57189455-42b0-4839-a86c-abda48b13174",  
  "ExternalImageId": "image6.jpg",  
  "Confidence": 100.0,  
  "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"  
},  
{  
  "BoundingBox": {  
    "Width": 0.5349419713020325,  
    "Top": 0.29124999046325684,  
    "Left": 0.16389399766921997,  
    "Height": 0.40187498927116394  
  },  
  "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",  
  "ExternalImageId": "image7.jpg",  
  "Confidence": 99.99979400634766,  
  "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"  
},  
{  
  "BoundingBox": {  
    "Width": 0.41499999165534973,  
    "Top": 0.09187500178813934,  
    "Left": 0.28083300590515137,  
    "Height": 0.3112500011920929  
  },  
  "FaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",  
  "ExternalImageId": "image8.jpg",  
  "Confidence": 99.99769592285156,  
  "ImageId": "a294da46-2cb1-5cc4-9045-61d7ca567662"  
},  
{  
  "BoundingBox": {  
    "Width": 0.48166701197624207,  
    "Top": 0.20999999344348907,  
    "Left": 0.21250000596046448,
```

```
        "Height": 0.36125001311302185
      },
      "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
      "ExternalImageId": "image9.jpg",
      "Confidence": 99.99949645996094,
      "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
    },
    {
      "BoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618019938468933,
        "Left": 0.5575000047683716,
        "Height": 0.24770599603652954
      },
      "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
      "ExternalImageId": "image10.jpg",
      "Confidence": 99.99340057373047,
      "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Daftar Wajah dalam Koleksi](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [ListFaces](#) di Referensi AWS CLI Perintah.

## list-stream-processors

Contoh kode berikut menunjukkan cara menggunakan `list-stream-processors`.

### AWS CLI

Untuk mencantumkan prosesor streaming di akun Anda

`list-stream-processors` Perintah berikut mencantumkan prosesor aliran di akun Anda dan status masing-masing.

```
aws rekognition list-stream-processors
```

Output:

```
{
```



```
"StreamProcessors": [  
  {  
    "Status": "STOPPED",  
    "Name": "my-stream-processor"  
  }  
]
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Video Streaming](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [ListStreamProcessors](#) di Referensi AWS CLI Perintah.

## recognize-celebrities

Contoh kode berikut menunjukkan cara menggunakan `recognize-celebrities`.

Untuk informasi selengkapnya, lihat [Mengenali selebriti dalam sebuah gambar](#).

### AWS CLI

Untuk mengenali selebriti dalam sebuah gambar

`recognize-celebrities` Perintah berikut mengenali selebriti dalam gambar tertentu yang disimpan dalam bucket Amazon S3. :

```
aws rekognition recognize-celebrities \  
  --image "S3object={Bucket=MyImageS3Bucket,Name=moviestars.jpg}"
```

Output:

```
{  
  "UnrecognizedFaces": [  
    {  
      "BoundingBox": {  
        "Width": 0.14416666328907013,  
        "Top": 0.077777778059244156,  
        "Left": 0.625,  
        "Height": 0.2746031880378723  
      },  
      "Confidence": 99.9990234375,  
      "Pose": {  
        "Yaw": 10.80408763885498,  

```

```
        "Roll": -12.761146545410156,
        "Pitch": 10.96889877319336
    },
    "Quality": {
        "Sharpness": 94.1185531616211,
        "Brightness": 79.18367004394531
    },
    "Landmarks": [
        {
            "Y": 0.18220913410186768,
            "X": 0.6702951788902283,
            "Type": "eyeLeft"
        },
        {
            "Y": 0.16337193548679352,
            "X": 0.7188183665275574,
            "Type": "eyeRight"
        },
        {
            "Y": 0.20739148557186127,
            "X": 0.7055801749229431,
            "Type": "nose"
        },
        {
            "Y": 0.2889308035373688,
            "X": 0.687512218952179,
            "Type": "mouthLeft"
        },
        {
            "Y": 0.2706988751888275,
            "X": 0.7250053286552429,
            "Type": "mouthRight"
        }
    ]
}
],
"CelebrityFaces": [
    {
        "MatchConfidence": 100.0,
        "Face": {
            "BoundingBox": {
                "Width": 0.14000000059604645,
                "Top": 0.1190476194024086,
                "Left": 0.82833331823349,
```

```
        "Height": 0.2666666805744171
      },
      "Confidence": 99.99359130859375,
      "Pose": {
        "Yaw": -10.509642601013184,
        "Roll": -14.51749324798584,
        "Pitch": 13.799399375915527
      },
      "Quality": {
        "Sharpness": 78.74752044677734,
        "Brightness": 42.201324462890625
      },
      "Landmarks": [
        {
          "Y": 0.2290833294391632,
          "X": 0.8709492087364197,
          "Type": "eyeLeft"
        },
        {
          "Y": 0.20639978349208832,
          "X": 0.9153988361358643,
          "Type": "eyeRight"
        },
        {
          "Y": 0.25417643785476685,
          "X": 0.8907724022865295,
          "Type": "nose"
        },
        {
          "Y": 0.32729196548461914,
          "X": 0.8876466155052185,
          "Type": "mouthLeft"
        },
        {
          "Y": 0.3115464746952057,
          "X": 0.9238573312759399,
          "Type": "mouthRight"
        }
      ]
    },
    "Name": "Celeb A",
    "Urls": [
      "www.imdb.com/name/aaaaaaaaa"
    ]
  ],
```

```
    "Id": "11111111"
  },
  {
    "MatchConfidence": 97.0,
    "Face": {
      "BoundingBox": {
        "Width": 0.13333334028720856,
        "Top": 0.24920634925365448,
        "Left": 0.4449999928474426,
        "Height": 0.2539682686328888
      },
      "Confidence": 99.99979400634766,
      "Pose": {
        "Yaw": 6.557040691375732,
        "Roll": -7.316643714904785,
        "Pitch": 9.272967338562012
      },
      "Quality": {
        "Sharpness": 83.23492431640625,
        "Brightness": 78.83267974853516
      },
      "Landmarks": [
        {
          "Y": 0.3625510632991791,
          "X": 0.48898839950561523,
          "Type": "eyeLeft"
        },
        {
          "Y": 0.35366007685661316,
          "X": 0.5313721299171448,
          "Type": "eyeRight"
        },
        {
          "Y": 0.3894785940647125,
          "X": 0.5173314809799194,
          "Type": "nose"
        },
        {
          "Y": 0.44889405369758606,
          "X": 0.5020005702972412,
          "Type": "mouthLeft"
        },
        {
          "Y": 0.4408611059188843,
```

```
        "X": 0.5351271629333496,
        "Type": "mouthRight"
      }
    ]
  },
  "Name": "Celeb B",
  "Urls": [
    "www.imdb.com/name/bbbbbbbbbb"
  ],
  "Id": "2222222"
},
{
  "MatchConfidence": 100.0,
  "Face": {
    "BoundingBox": {
      "Width": 0.12416666746139526,
      "Top": 0.2968254089355469,
      "Left": 0.2150000035762787,
      "Height": 0.23650793731212616
    },
    "Confidence": 99.99958801269531,
    "Pose": {
      "Yaw": 7.801797866821289,
      "Roll": -8.326810836791992,
      "Pitch": 7.844768047332764
    },
    "Quality": {
      "Sharpness": 86.93206024169922,
      "Brightness": 79.81291198730469
    },
    "Landmarks": [
      {
        "Y": 0.4027804136276245,
        "X": 0.2575301229953766,
        "Type": "eyeLeft"
      },
      {
        "Y": 0.3934555947780609,
        "X": 0.2956969439983368,
        "Type": "eyeRight"
      },
      {
        "Y": 0.4309830069541931,
        "X": 0.2837020754814148,
```

```
        "Type": "nose"
      },
      {
        "Y": 0.48186683654785156,
        "X": 0.26812544465065,
        "Type": "mouthLeft"
      },
      {
        "Y": 0.47338807582855225,
        "X": 0.29905644059181213,
        "Type": "mouthRight"
      }
    ]
  },
  "Name": "Celeb C",
  "Urls": [
    "www.imdb.com/name/ccccccccc"
  ],
  "Id": "3333333"
},
{
  "MatchConfidence": 97.0,
  "Face": {
    "BoundingBox": {
      "Width": 0.11916666477918625,
      "Top": 0.3698412775993347,
      "Left": 0.008333333767950535,
      "Height": 0.22698412835597992
    },
    "Confidence": 99.99999237060547,
    "Pose": {
      "Yaw": 16.38478660583496,
      "Roll": -1.0260354280471802,
      "Pitch": 5.975185394287109
    },
    "Quality": {
      "Sharpness": 83.23492431640625,
      "Brightness": 61.408443450927734
    },
    "Landmarks": [
      {
        "Y": 0.4632347822189331,
        "X": 0.049406956881284714,
        "Type": "eyeLeft"
      }
    ]
  }
}
```

```
    },
    {
      "Y": 0.46388113498687744,
      "X": 0.08722897619009018,
      "Type": "eyeRight"
    },
    {
      "Y": 0.5020678639411926,
      "X": 0.0758260041475296,
      "Type": "nose"
    },
    {
      "Y": 0.544157862663269,
      "X": 0.054029736667871475,
      "Type": "mouthLeft"
    },
    {
      "Y": 0.5463630557060242,
      "X": 0.08464983850717545,
      "Type": "mouthRight"
    }
  ]
},
"Name": "Celeb D",
"Urls": [
  "www.imdb.com/name/ddddddddd"
],
"Id": "44444444"
}
]
```

Untuk informasi selengkapnya, lihat [Mengenal Selebriti dalam Gambar](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [RecognizeCelebrities](#) di Referensi AWS CLI Perintah.

## search-faces-by-image

Contoh kode berikut menunjukkan cara menggunakan `search-faces-by-image`.

Untuk informasi selengkapnya, lihat [Mencari wajah \(gambar\)](#).

## AWS CLI

Untuk mencari wajah dalam koleksi yang cocok dengan wajah terbesar dalam sebuah gambar.

`search-faces-by-image` Perintah berikut mencari wajah dalam koleksi yang cocok dengan wajah terbesar dalam gambar yang ditentukan. :

```
aws rekognition search-faces-by-image \  
  --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"ExamplePerson.jpg"}}' \  
  --collection-id MyFaceImageCollection  
  
{  
  "SearchedFaceBoundingBox": {  
    "Width": 0.18562500178813934,  
    "Top": 0.1618015021085739,  
    "Left": 0.5575000047683716,  
    "Height": 0.24770642817020416  
  },  
  "SearchedFaceConfidence": 99.993408203125,  
  "FaceMatches": [  
    {  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.18562500178813934,  
          "Top": 0.1618019938468933,  
          "Left": 0.5575000047683716,  
          "Height": 0.24770599603652954  
        },  
        "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",  
        "ExternalImageId": "example-image.jpg",  
        "Confidence": 99.99340057373047,  
        "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"  
      },  
      "Similarity": 99.97913360595703  
    },  
    {  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.18562500178813934,  
          "Top": 0.1618019938468933,  
          "Left": 0.5575000047683716,  
          "Height": 0.24770599603652954  
        },  
        "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
```



```
        "ExternalImageId": "image3.jpg",
        "Confidence": 99.99340057373047,
        "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
    },
    "Similarity": 99.97913360595703
},
{
    "Face": {
        "BoundingBox": {
            "Width": 0.41499999165534973,
            "Top": 0.09187500178813934,
            "Left": 0.28083300590515137,
            "Height": 0.3112500011920929
        },
        "FaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",
        "ExternalImageId": "image2.jpg",
        "Confidence": 99.99769592285156,
        "ImageId": "a294da46-2cb1-5cc4-9045-61d7ca567662"
    },
    "Similarity": 99.18069458007812
},
{
    "Face": {
        "BoundingBox": {
            "Width": 0.48166701197624207,
            "Top": 0.20999999344348907,
            "Left": 0.21250000596046448,
            "Height": 0.36125001311302185
        },
        "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
        "ExternalImageId": "image1.jpg",
        "Confidence": 99.99949645996094,
        "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
    },
    "Similarity": 98.66607666015625
},
{
    "Face": {
        "BoundingBox": {
            "Width": 0.5349419713020325,
            "Top": 0.29124999046325684,
            "Left": 0.16389399766921997,
            "Height": 0.40187498927116394
        },
```

```
    "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
    "ExternalImageId": "image9.jpg",
    "Confidence": 99.99979400634766,
    "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
  },
  "Similarity": 98.24278259277344
},
{
  "Face": {
    "BoundingBox": {
      "Width": 0.5307819843292236,
      "Top": 0.2862499952316284,
      "Left": 0.1564060002565384,
      "Height": 0.3987500071525574
    },
    "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
    "ExternalImageId": "image10.jpg",
    "Confidence": 99.99970245361328,
    "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
  },
  "Similarity": 98.10665893554688
},
{
  "Face": {
    "BoundingBox": {
      "Width": 0.5074880123138428,
      "Top": 0.3774999976158142,
      "Left": 0.18302799761295319,
      "Height": 0.3812499940395355
    },
    "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
    "ExternalImageId": "image6.jpg",
    "Confidence": 99.99930572509766,
    "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
  },
  "Similarity": 98.10526275634766
},
{
  "Face": {
    "BoundingBox": {
      "Width": 0.5574039816856384,
      "Top": 0.37187498807907104,
      "Left": 0.14559100568294525,
      "Height": 0.4181250035762787
```

```

    },
    "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
    "ExternalImageId": "image5.jpg",
    "Confidence": 99.99960327148438,
    "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
  },
  "Similarity": 97.94659423828125
},
{
  "Face": {
    "BoundingBox": {
      "Width": 0.5773710012435913,
      "Top": 0.34437501430511475,
      "Left": 0.12396000325679779,
      "Height": 0.4337500035762787
    },
    "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
    "ExternalImageId": "image8.jpg",
    "Confidence": 100.0,
    "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
  },
  "Similarity": 97.93476867675781
}
],
"FaceModelVersion": "3.0"
}

```

Untuk informasi selengkapnya, lihat [Mencari Wajah Menggunakan Gambar](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [SearchFacesByImage](#) di Referensi AWS CLI Perintah.

## search-faces

Contoh kode berikut menunjukkan cara menggunakan `search-faces`.

Untuk informasi selengkapnya, lihat [Mencari wajah \(ID wajah\)](#).

### AWS CLI

Untuk mencari wajah dalam koleksi yang cocok dengan ID wajah.

`search-faces` Perintah berikut mencari wajah dalam koleksi yang cocok dengan ID wajah yang ditentukan.

```
aws rekognition search-faces \  
  --face-id 8d3cfc70-4ba8-4b36-9644-90fba29c2dac \  
  --collection-id MyCollection
```

Output:

```
{  
  "SearchedFaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",  
  "FaceModelVersion": "3.0",  
  "FaceMatches": [  
    {  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.48166701197624207,  
          "Top": 0.20999999344348907,  
          "Left": 0.21250000596046448,  
          "Height": 0.36125001311302185  
        },  
        "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",  
        "ExternalImageId": "image1.jpg",  
        "Confidence": 99.99949645996094,  
        "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"  
      },  
      "Similarity": 99.30997467041016  
    },  
    {  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.18562500178813934,  
          "Top": 0.1618019938468933,  
          "Left": 0.5575000047683716,  
          "Height": 0.24770599603652954  
        },  
        "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",  
        "ExternalImageId": "example-image.jpg",  
        "Confidence": 99.99340057373047,  
        "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"  
      },  
      "Similarity": 99.24862670898438  
    },  
  ],  
}
```

```
{
  "Face": {
    "BoundingBox": {
      "Width": 0.18562500178813934,
      "Top": 0.1618019938468933,
      "Left": 0.5575000047683716,
      "Height": 0.24770599603652954
    },
    "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
    "ExternalImageId": "image3.jpg",
    "Confidence": 99.99340057373047,
    "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
  },
  "Similarity": 99.24862670898438
},
{
  "Face": {
    "BoundingBox": {
      "Width": 0.5349419713020325,
      "Top": 0.29124999046325684,
      "Left": 0.16389399766921997,
      "Height": 0.40187498927116394
    },
    "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
    "ExternalImageId": "image9.jpg",
    "Confidence": 99.99979400634766,
    "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
  },
  "Similarity": 96.73158264160156
},
{
  "Face": {
    "BoundingBox": {
      "Width": 0.5307819843292236,
      "Top": 0.2862499952316284,
      "Left": 0.1564060002565384,
      "Height": 0.3987500071525574
    },
    "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
    "ExternalImageId": "image10.jpg",
    "Confidence": 99.99970245361328,
    "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
  },
  "Similarity": 96.48291015625
}
```

```
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5074880123138428,
        "Top": 0.3774999976158142,
        "Left": 0.18302799761295319,
        "Height": 0.3812499940395355
      },
      "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
      "ExternalImageId": "image6.jpg",
      "Confidence": 99.99930572509766,
      "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
    },
    "Similarity": 96.43287658691406
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5574039816856384,
        "Top": 0.37187498807907104,
        "Left": 0.14559100568294525,
        "Height": 0.4181250035762787
      },
      "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
      "ExternalImageId": "image5.jpg",
      "Confidence": 99.99960327148438,
      "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
    },
    "Similarity": 95.25305938720703
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5773710012435913,
        "Top": 0.34437501430511475,
        "Left": 0.12396000325679779,
        "Height": 0.4337500035762787
      },
      "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
      "ExternalImageId": "image8.jpg",
      "Confidence": 100.0,
      "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
    },
  },
```

```
    "Similarity": 95.22837829589844
  }
]
}
```

Untuk informasi selengkapnya, lihat [Mencari Wajah Menggunakan ID Wajahnya](#) di Panduan Pengembang Amazon Rekognition.

- Untuk detail API, lihat [SearchFaces](#) di Referensi AWS CLI Perintah.

## start-celebrity-recognition

Contoh kode berikut menunjukkan cara menggunakan `start-celebrity-recognition`.

### AWS CLI

Untuk memulai pengakuan selebriti dalam video yang disimpan

`start-celebrity-recognition` Perintah berikut memulai pekerjaan untuk mencari selebriti dalam file video tertentu yang disimpan dalam bucket Amazon S3.

```
aws rekognition start-celebrity-recognition \
  --video "S3object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Untuk informasi selengkapnya, lihat [Mengenali Selebriti dalam Video Tersimpan](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [StartCelebrityRecognition](#) di Referensi AWS CLI Perintah.

## start-content-moderation

Contoh kode berikut menunjukkan cara menggunakan `start-content-moderation`.

### AWS CLI

Untuk memulai pengenalan konten yang tidak aman dalam video yang disimpan

`start-content-moderation` Perintah berikut memulai pekerjaan untuk mendeteksi konten yang tidak aman dalam file video tertentu yang disimpan dalam bucket Amazon S3.

```
aws rekognition start-content-moderation \  
  --video "S3object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Video Tersimpan Tidak Aman di Panduan Pengembang Rekognition Amazon](#).

- Untuk detail API, lihat [StartContentModeration](#) di Referensi AWS CLI Perintah.

## **start-face-detection**

Contoh kode berikut menunjukkan cara menggunakan `start-face-detection`.

### AWS CLI

Untuk mendeteksi wajah dalam video

`start-face-detection` Perintah berikut memulai pekerjaan untuk mendeteksi wajah dalam file video tertentu yang disimpan dalam bucket Amazon S3.

```
aws rekognition start-face-detection  
  --video "S3object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"  
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Wajah dalam Video Tersimpan](#) di Panduan Pengembang Rekognition Amazon.



- Untuk detail API, lihat [StartFaceDetection](#) di Referensi AWS CLI Perintah.

## start-face-search

Contoh kode berikut menunjukkan cara menggunakan `start-face-search`.

### AWS CLI

Untuk mencari wajah dalam koleksi yang cocok dengan wajah yang terdeteksi dalam video

`start-face-search` Perintah berikut memulai pekerjaan untuk mencari wajah dalam koleksi yang cocok dengan wajah yang terdeteksi dalam file video yang ditentukan dalam bucket Amazon S3.

```
aws rekognition start-face-search \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}" \  
  --collection-id collection
```

Output:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"  
}
```

Untuk informasi selengkapnya, lihat [Mencari Video Tersimpan untuk Wajah](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [StartFaceSearch](#) di Referensi AWS CLI Perintah.

## start-label-detection

Contoh kode berikut menunjukkan cara menggunakan `start-label-detection`.

### AWS CLI

Untuk mendeteksi objek dan adegan dalam video

`start-label-detection` Perintah berikut memulai pekerjaan untuk mendeteksi objek dan adegan dalam file video tertentu yang disimpan dalam bucket Amazon S3.

```
aws rekognition start-label-detection \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

```
--video "S3object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Untuk informasi selengkapnya, lihat [Mendeteksi Label dalam Video](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [StartLabelDetection](#) di Referensi AWS CLI Perintah.

## start-person-tracking

Contoh kode berikut menunjukkan cara menggunakan `start-person-tracking`.

AWS CLI

Untuk memulai jalur orang dalam video yang disimpan

`start-person-tracking` Perintah berikut memulai pekerjaan untuk melacak jalur yang diambil orang di bagian video tertentu yang disimpan di bucket Amazon S3. :

```
aws rekognition start-person-tracking \
  --video "S3object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Untuk informasi selengkapnya, lihat [People Pathing](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [StartPersonTracking](#) di Referensi AWS CLI Perintah.

## start-stream-processor

Contoh kode berikut menunjukkan cara menggunakan `start-stream-processor`.

## AWS CLI

Untuk memulai prosesor streaming

`start-stream-processor` Perintah berikut memulai prosesor aliran video yang ditentukan.

```
aws rekognition start-stream-processor \  
  --name my-stream-processor
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Video Streaming](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [StartStreamProcessor](#) di Referensi AWS CLI Perintah.

## **stop-stream-processor**

Contoh kode berikut menunjukkan cara menggunakan `stop-stream-processor`.

## AWS CLI

Untuk menghentikan prosesor aliran yang sedang berjalan

`stop-stream-processor` Perintah berikut menghentikan prosesor aliran berjalan yang ditentukan.

```
aws rekognition stop-stream-processor \  
  --name my-stream-processor
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Video Streaming](#) di Panduan Pengembang Rekognition Amazon.

- Untuk detail API, lihat [StopStreamProcessor](#) di Referensi AWS CLI Perintah.

## AWS RAM contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS RAM.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **accept-resource-share-invitation**

Contoh kode berikut menunjukkan cara menggunakan `accept-resource-share-invitation`.

AWS CLI

Untuk menerima undangan berbagi sumber daya

`accept-resource-share-invitation` Contoh berikut menerima undangan berbagi sumber daya yang ditentukan. Kepala sekolah di akun yang diundang dapat segera mulai menggunakan sumber daya di bagian.

```
aws ram accept-resource-share-invitation \  
  --resource-share-invitation-arn arn:aws:ram:us-west-2:111111111111:resource-  
share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE
```

Output:

```
{  
  "resourceShareInvitation": {  
    "resourceShareInvitationArn": "arn:aws:ram:us-west-2:111111111111:resource-  
share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE",  
    "resourceShareName": "MyLicenseShare",  
    "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-  
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",
```

```
    "senderAccountId": "111111111111",
    "receiverAccountId": "222222222222",
    "invitationTimestamp": "2021-09-22T15:07:35.620000-07:00",
    "status": "ACCEPTED"
  }
}
```

- Untuk detail API, lihat [AcceptResourceShareInvitation](#) di Referensi AWS CLI Perintah.

## associate-resource-share-permission

Contoh kode berikut menunjukkan cara menggunakan `associate-resource-share-permission`.

### AWS CLI

Untuk mengaitkan izin yang dikelola RAM dengan pembagian sumber daya

`associate-resource-share-permission` Contoh berikut menggantikan izin terkelola yang ada untuk jenis sumber daya yang relevan dengan izin terkelola yang ditentukan. Akses ke semua sumber daya dari jenis sumber daya yang relevan diatur oleh izin baru.

```
aws ram associate-resource-share-permission \
  --permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite \
  --replace \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE
```

Output:

```
{
  "returnValue": true
}
```

- Untuk detail API, lihat [AssociateResourceSharePermission](#) di Referensi AWS CLI Perintah.

## associate-resource-share

Contoh kode berikut menunjukkan cara menggunakan `associate-resource-share`.

## AWS CLI

Contoh 1: Untuk mengaitkan sumber daya dengan pembagian sumber daya

`associate-resource-share` Contoh berikut menambahkan konfigurasi lisensi untuk berbagi sumber daya tertentu.

```
aws ram associate-resource-share \
  --resource-share arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
  --resource-arns arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE
```

Output:

```
{
  "resourceShareAssociations": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",
      "associatedEntity": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
      "associationType": "RESOURCE",
      "status": "ASSOCIATING",
      "external": false
    }
  ]
}
```

Contoh 2: Untuk mengaitkan prinsipal dengan pembagian sumber daya

`associate-resource-share` Contoh berikut memberikan akses untuk berbagi sumber daya yang ditentukan ke semua akun di unit organisasi yang ditentukan.

```
aws ram associate-resource-share \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
  --principals arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-
rEXAMPLE
```

Output:

```
{
  "resourceShareAssociations": [
    {
      "status": "ASSOCIATING",
      "associationType": "PRINCIPAL",
      "associatedEntity": "arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-rEXAMPLE",
      "external": false,
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE"
    }
  ]
}
```

- Untuk detail API, lihat [AssociateResourceShare](#) di Referensi AWS CLI Perintah.

## create-resource-share

Contoh kode berikut menunjukkan cara menggunakan `create-resource-share`.

### AWS CLI

Contoh 1: Untuk membuat pembagian sumber daya

`create-resource-share` Contoh berikut menciptakan berbagi sumber daya kosong dengan nama yang ditentukan. Anda harus secara terpisah menambahkan sumber daya, prinsipal, dan izin untuk berbagi.

```
aws ram create-resource-share \
  --name MyNewResourceShare
```

Output:

```
{
  "resourceShare": {
    "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/4476c27d-8feb-4b21-afe9-7de23EXAMPLE",
    "name": "MyNewResourceShare",
    "owningAccountId": "123456789012",
    "allowExternalPrincipals": true,
    "status": "ACTIVE",
```

```
    "creationTime": 1634586271.302,  
    "lastUpdatedTime": 1634586271.302  
  }  
}
```

Contoh 2: Untuk membuat pembagian sumber daya dengan AWS akun sebagai prinsipal

`create-resource-share` Contoh berikut membuat berbagi sumber daya dan memberikan akses ke AWS akun yang ditentukan (222222222222). Jika prinsipal yang ditentukan bukan bagian dari AWS Organisasi yang sama, maka undangan dikirim dan harus diterima sebelum akses diberikan.

```
aws ram create-resource-share \  
  --name MyNewResourceShare \  
  --principals 222222222222
```

Contoh 3: Untuk membuat pembagian sumber daya yang dibatasi untuk AWS Organisasi Anda

`create-resource-share` Contoh berikut membuat pembagian sumber daya yang dibatasi untuk akun di AWS Organisasi tempat akun Anda menjadi anggotanya, dan menambahkan OU yang ditentukan sebagai prinsipal. Semua akun di OU itu dapat menggunakan sumber daya dalam pembagian sumber daya.

```
aws ram create-resource-share \  
  --name MyNewResourceShare \  
  --no-allow-external-principals \  
  --principals arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-  
rEXAMPLE
```

Output:

```
{  
  "resourceShare": {  
    "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",  
    "name": "MyNewResourceShare",  
    "owningAccountId": "123456789012",  
    "allowExternalPrincipals": false,  
    "status": "ACTIVE",  
    "creationTime": 1634587042.49,  
    "lastUpdatedTime": 1634587042.49  
  }  
}
```



```
}  
}
```

- Untuk detail API, lihat [CreateResourceShare](#) di Referensi AWS CLI Perintah.

## **delete-resource-share**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-share`.

### AWS CLI

Untuk menghapus pembagian sumber daya

`delete-resource-share` Contoh berikut menghapus share sumber daya yang ditentukan.

```
aws ram delete-resource-share \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

Output berikut menunjukkan keberhasilan:

```
{  
  "returnValue": true  
}
```

- Untuk detail API, lihat [DeleteResourceShare](#) di Referensi AWS CLI Perintah.

## **disassociate-resource-share-permission**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-resource-share-permission`.

### AWS CLI

Untuk menghapus izin yang dikelola RAM untuk jenis sumber daya dari pembagian sumber daya

`disassociate-resource-share-permission` Contoh berikut menghapus izin terkelola RAM untuk database Glue dari pembagian sumber daya yang ditentukan.

```
aws ram disassociate-resource-share-permission \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

```
--resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
--permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite
```

Output:

```
{
  "returnValue": true
}
```

- Untuk detail API, lihat [DisassociateResourceSharePermission](#) di Referensi AWS CLI Perintah.

## disassociate-resource-share

Contoh kode berikut menunjukkan cara menggunakan `disassociate-resource-share`.

### AWS CLI

Untuk menghapus sumber daya dari pembagian sumber daya

`disassociate-resource-share` Contoh berikut menghapus sumber daya yang ditentukan, dalam hal ini subnet VPC, dari berbagi sumber daya yang ditentukan. Setiap prinsipal dengan akses ke pembagian sumber daya tidak dapat lagi melakukan operasi pada sumber daya itu.

```
aws ram disassociate-resource-share \
  --resource-arns arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-
b505-7e2a-420d-6f5d3EXAMPLE
```

Output:

```
{
  "resourceShareAssociations": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "associatedEntity": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE",
      "associationType": "RESOURCE",
    }
  ]
}
```

```
    "status": "DISASSOCIATING",  
    "external": false  
  ]  
}
```

- Untuk detail API, lihat [DisassociateResourceShare](#) di Referensi AWS CLI Perintah.

## **enable-sharing-with-aws-organization**

Contoh kode berikut menunjukkan cara menggunakan `enable-sharing-with-aws-organization`.

### AWS CLI

Untuk mengaktifkan berbagi sumber daya di seluruh AWS Organizations

`enable-sharing-with-aws-organization` Contoh berikut memungkinkan berbagi sumber daya di seluruh organisasi dan unit organisasi Anda.

```
aws ram enable-sharing-with-aws-organization
```

Output berikut menunjukkan keberhasilan.

```
{  
  "returnValue": true  
}
```

- Untuk detail API, lihat [EnableSharingWithAwsOrganization](#) di Referensi AWS CLI Perintah.

## **get-permission**

Contoh kode berikut menunjukkan cara menggunakan `get-permission`.

### AWS CLI

Untuk mengambil detail untuk izin yang dikelola RAM

`get-permission` Contoh berikut menampilkan rincian untuk versi default dari izin terkelola RAM yang ditentukan.

```
aws ram get-permission \
  --permission-arn arn:aws:ram::aws:permission/
  AWSRAMPermissionGlueTableReadWriteForDatabase
```

Output:

```
{
  "permission": {
    "arn": "arn:aws:ram::aws:permission/
  AWSRAMPermissionGlueTableReadWriteForDatabase",
    "version": "2",
    "defaultVersion": true,
    "name": "AWSRAMPermissionGlueTableReadWriteForDatabase",
    "resourceType": "glue:Database",
    "permission": "[{\"Effect\":\"Allow\",\"Action\":[\"glue:GetTable
\", \"glue:UpdateTable\", \"glue>DeleteTable\", \"glue:BatchDeleteTable\",
\", \"glue:BatchDeleteTableVersion\", \"glue:GetTableVersion\", \"glue:GetTableVersions
\", \"glue:GetPartition\", \"glue:GetPartitions\", \"glue:BatchGetPartition\",
\", \"glue:BatchCreatePartition\", \"glue>CreatePartition\", \"glue:UpdatePartition
\", \"glue:BatchDeletePartition\", \"glue>DeletePartition\", \"glue:GetTables\",
\", \"glue:SearchTables\"]}]",
    "creationTime": 1624912434.431,
    "lastUpdatedTime": 1624912434.431,
    "isResourceTypeDefault": false
  }
}
```

- Untuk detail API, lihat [GetPermission](#) di Referensi AWS CLI Perintah.

## get-resource-policies

Contoh kode berikut menunjukkan cara menggunakan `get-resource-policies`.

### AWS CLI

Untuk mendapatkan kebijakan untuk sumber daya

`get-resource-policies` Contoh berikut menampilkan kebijakan izin berbasis sumber daya untuk sumber daya tertentu yang terkait dengan pembagian sumber daya.

```
aws ram get-resource-policies \
```

```
--resource-arns arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE
```

### Output:

```
{
  "policies": [
    [{"Version": "2008-10-17", "Statement": [{"Sid": "RamStatement1",
      "Effect": "Allow", "Principal": {"AWS": []}, "Action": ["ec2:RunInstances",
      "ec2:CreateNetworkInterface", "ec2:DescribeSubnets"], "Resource":
      "arn:aws:ec2:us-west-2:123456789012:subnet/subnet-0250c25a1fEXAMPLE"}]}]
  ]
}
```

- Untuk detail API, lihat [GetResourcePolicies](#) di Referensi AWS CLI Perintah.

## get-resource-share-associations

Contoh kode berikut menunjukkan cara menggunakan `get-resource-share-associations`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua asosiasi sumber daya untuk semua jenis sumber daya

`get-resource-share-associations` Contoh berikut mencantumkan asosiasi sumber daya untuk semua jenis sumber daya di semua pembagian sumber daya Anda.

```
aws ram get-resource-share-associations \
  --association-type RESOURCE
```

### Output:

```
{
  "resourceShareAssociations": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "associatedEntity": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE",
      "resourceShareName": "MySubnetShare",
    }
  ]
}
```

```

        "associationType": "RESOURCE",
        "status": "ASSOCIATED",
        "creationTime": 1565303590.973,
        "lastUpdatedTime": 1565303591.695,
        "external": false
    },
    {
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/8167bdfe-4480-4a01-8632-315e0EXAMPLE",
        "associatedEntity": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
        "resourceShareName": "MyLicenseShare",
        "associationType": "RESOURCE",
        "status": "ASSOCIATED",
        "creationTime": 1632342958.457,
        "lastUpdatedTime": 1632342958.907,
        "external": false
    }
]
}

```

Contoh 2: Untuk mencantumkan asosiasi utama untuk pembagian sumber daya

`get-resource-share-associations` Contoh berikut hanya mencantumkan asosiasi utama hanya untuk pembagian sumber daya yang ditentukan.

```

aws ram get-resource-share-associations \
  --resource-share-arns arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE \
  --association-type PRINCIPAL

```

Output:

```

{
  "resourceShareAssociations": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
      "resourceShareName": "MyNewResourceShare",
      "associatedEntity": "arn:aws:organizations::123456789012:ou/
o-63bEXAMPLE/ou-46xi-rEXAMPLE",
      "associationType": "PRINCIPAL",

```

```
    "status": "ASSOCIATED",
    "creationTime": 1634587042.49,
    "lastUpdatedTime": 1634587044.291,
    "external": false
  }
]
```

- Untuk detail API, lihat [GetResourceShareAssociations](#) di Referensi AWS CLI Perintah.

## get-resource-share-invitations

Contoh kode berikut menunjukkan cara menggunakan `get-resource-share-invitations`.

### AWS CLI

Untuk membuat daftar undangan berbagi sumber daya Anda

`get-resource-share-invitations` Contoh berikut mencantumkan undangan berbagi sumber daya Anda saat ini.

```
aws ram get-resource-share-invitations
```

Output:

```
{
  "resourceShareInvitations": [
    {
      "resourceShareInvitationArn": "arn:aws:ram:us-west-2:1:111111111111:resource-share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE",
      "resourceShareName": "project-resource-share",
      "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-share/fcb639f0-1449-4744-35bc-a983fEXAMPLE",
      "senderAccountId": "111111111111",
      "receiverAccountId": "222222222222",
      "invitationTimestamp": 1565312166.258,
      "status": "PENDING"
    }
  ]
}
```

- Untuk detail API, lihat [GetResourceShareInvitations](#) di Referensi AWS CLI Perintah.

## get-resource-shares

Contoh kode berikut menunjukkan cara menggunakan `get-resource-shares`.

### AWS CLI

Contoh 1: Untuk membuat daftar pembagian sumber daya yang Anda miliki dan bagikan dengan orang lain

`get-resource-shares` Contoh berikut mencantumkan pembagian sumber daya yang dibuat dan dibagikan dengan orang lain.

```
aws ram get-resource-shares \  
  --resource-owner SELF
```

### Output:

```
{  
  "resourceShares": [  
    {  
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/3ab63985-99d9-1cd2-7d24-75e93EXAMPLE",  
      "name": "my-resource-share",  
      "owningAccountId": "123456789012",  
      "allowExternalPrincipals": false,  
      "status": "ACTIVE",  
      "tags": [  
        {  
          "key": "project",  
          "value": "lima"  
        }  
      ]  
      "creationTime": 1565295733.282,  
      "lastUpdatedTime": 1565295733.282  
    },  
    {  
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",  
      "name": "my-resource-share",  
      "owningAccountId": "123456789012",
```



```

        "allowExternalPrincipals": true,
        "status": "ACTIVE",
        "creationTime": 1565295733.282,
        "lastUpdatedTime": 1565295733.282
    }
]
}

```

Contoh 2: Untuk mencantumkan saham sumber daya yang dimiliki oleh orang lain dan dibagikan dengan Anda

`get-resource-shares` Contoh berikut mencantumkan pembagian sumber daya yang dibuat dan dibagikan orang lain dengan Anda. Dalam contoh ini, tidak ada.

```

aws ram get-resource-shares \
  --resource-owner OTHER-ACCOUNTS

```

Output:

```

{
  "resourceShares": []
}

```

- Untuk detail API, lihat [GetResourceShares](#) di Referensi AWS CLI Perintah.

## **list-pending-invitation-resources**

Contoh kode berikut menunjukkan cara menggunakan `list-pending-invitation-resources`.

AWS CLI

Untuk membuat daftar sumber daya yang tersedia dalam pembagian sumber daya yang tertunda

`list-pending-invitation-resources` Contoh berikut mencantumkan semua sumber daya yang ada di bagian sumber daya yang terkait dengan undangan yang ditentukan.

```

aws ram list-pending-invitation-resources \
  --resource-share-invitation-arn arn:aws:ram:us-west-2:123456789012:resource-
share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE

```

**Output:**

```
{
  "resources": [
    {
      "arn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-04a555b0e6EXAMPLE",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
      "creationTime": 1634676051.269,
      "lastUpdatedTime": 1634676052.07,
      "status": "AVAILABLE",
      "type": "ec2:Subnet"
    },
    {
      "arn": "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "creationTime": 1624912434.431,
      "lastUpdatedTime": 1624912434.431,
      "status": "AVAILABLE",
      "type": "license-manager:LicenseConfiguration"
    }
  ]
}
```

- Untuk detail API, lihat [ListPendingInvitationResources](#) di Referensi AWS CLI Perintah.

**list-permissions**

Contoh kode berikut menunjukkan cara menggunakan `list-permissions`.

**AWS CLI**

Untuk mencantumkan izin terkelola RAM yang tersedia

`list-permissions` Contoh berikut mencantumkan semua izin terkelola RAM yang tersedia hanya untuk jenis sumber daya database AWS Glue.

```
aws ram list-permissions \
  --resource-type glue:Database
```

## Output:

```
{
  "permissions": [
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMDefaultPermissionGlueDatabase",
      "version": "1",
      "defaultVersion": true,
      "name": "AWSRAMDefaultPermissionGlueDatabase",
      "resourceType": "glue:Database",
      "creationTime": 1592007820.935,
      "lastUpdatedTime": 1592007820.935,
      "isResourceTypeDefault": true
    },
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueAllTablesReadWriteForDatabase",
      "version": "2",
      "defaultVersion": true,
      "name": "AWSRAMPermissionGlueAllTablesReadWriteForDatabase",
      "resourceType": "glue:Database",
      "creationTime": 1624912413.323,
      "lastUpdatedTime": 1624912413.323,
      "isResourceTypeDefault": false
    },
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite",
      "version": "2",
      "defaultVersion": true,
      "name": "AWSRAMPermissionGlueDatabaseReadWrite",
      "resourceType": "glue:Database",
      "creationTime": 1624912417.4,
      "lastUpdatedTime": 1624912417.4,
      "isResourceTypeDefault": false
    },
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase",
      "version": "2",
      "defaultVersion": true,
      "name": "AWSRAMPermissionGlueTableReadWriteForDatabase",
      "resourceType": "glue:Database",
```

```

        "creationTime": 1624912434.431,
        "lastUpdatedTime": 1624912434.431,
        "isResourceTypeDefault": false
    }
]
}

```

`list-permissions` Contoh berikut menampilkan izin terkelola RAM yang tersedia untuk semua jenis sumber daya.

```
aws ram list-permissions
```

Output:

```

{
  "permissions": [
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthority",
      "version": "1",
      "defaultVersion": true,
      "name":
"AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthority",
      "resourceType": "acm-pca:CertificateAuthority",
      "creationTime": 1623264861.085,
      "lastUpdatedTime": 1623264861.085,
      "isResourceTypeDefault": false
    },
    {
      "arn": "arn:aws:ram::aws:permission/AWSRAMDefaultPermissionAppMesh",
      "version": "1",
      "defaultVersion": true,
      "name": "AWSRAMDefaultPermissionAppMesh",
      "resourceType": "appmesh:Mesh",
      "creationTime": 1589307188.584,
      "lastUpdatedTime": 1589307188.584,
      "isResourceTypeDefault": true
    },
    ...TRUNCATED FOR BREVITY...
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMSubordinateCACertificatePathLen0IssuanceCertificateAuthority",
      "version": "1",

```

```
        "defaultVersion": true,
        "name":
"AWSRAMSubordinateCACertificatePathLen0IssuanceCertificateAuthority",
        "resourceType": "acm-pca:CertificateAuthority",
        "creationTime": 1623264876.75,
        "lastUpdatedTime": 1623264876.75,
        "isResourceTypeDefault": false
    }
]
}
```

- Untuk detail API, lihat [ListPermissions](#) di Referensi AWS CLI Perintah.

## list-principals

Contoh kode berikut menunjukkan cara menggunakan `list-principals`.

### AWS CLI

Untuk daftar prinsipal dengan akses ke sumber daya

`list-principals` Contoh berikut menampilkan daftar prinsipal yang dapat mengakses sumber daya dari jenis yang ditentukan melalui pembagian sumber daya apa pun.

```
aws ram list-principals \
  --resource-type ec2:Subnet
```

Output:

```
{
  "principals": [
    {
      "id": "arn:aws:organizations::123456789012:ou/o-gx7EXAMPLE/ou-29c5-
zEXAMPLE",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "creationTime": 1565298209.737,
      "lastUpdatedTime": 1565298211.019,
      "external": false
    }
  ]
}
```

- Untuk detail API, lihat [ListPrincipals](#) di Referensi AWS CLI Perintah.

## list-resource-share-permissions

Contoh kode berikut menunjukkan cara menggunakan `list-resource-share-permissions`.

### AWS CLI

Untuk mencantumkan semua izin yang dikelola RAM yang saat ini dilampirkan ke pembagian sumber daya

`list-resource-share-permissions` Contoh berikut mencantumkan semua izin yang dikelola RAM yang dilampirkan ke pembagian sumber daya yang ditentukan.

```
aws ram list-resource-share-permissions \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE
```

### Output:

```
{
  "permissions": [
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMDefaultPermissionLicenseConfiguration",
      "version": "1",
      "resourceType": "license-manager:LicenseConfiguration",
      "status": "ASSOCIATED",
      "lastUpdatedTime": 1632342984.234
    },
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite",
      "version": "2",
      "resourceType": "glue:Database",
      "status": "ASSOCIATED",
      "lastUpdatedTime": 1632512462.297
    }
  ]
}
```

- Untuk detail API, lihat [ListResourceSharePermissions](#) di Referensi AWS CLI Perintah.

## list-resource-types

Contoh kode berikut menunjukkan cara menggunakan `list-resource-types`.

### AWS CLI

Untuk daftar jenis sumber daya yang didukung oleh AWS RAM

`list-resource-types` Contoh berikut mencantumkan semua jenis sumber daya yang saat ini didukung oleh AWS RAM.

```
aws ram list-resource-types
```

Output:

```
{
  "resourceTypes": [
    {
      "resourceType": "route53resolver:FirewallRuleGroup",
      "serviceName": "route53resolver"
    },
    {
      "resourceType": "ec2:LocalGatewayRouteTable",
      "serviceName": "ec2"
    },
    ...OUTPUT TRUNCATED FOR BREVITY...
    {
      "resourceType": "ec2:Subnet",
      "serviceName": "ec2"
    },
    {
      "resourceType": "ec2:TransitGatewayMulticastDomain",
      "serviceName": "ec2"
    }
  ]
}
```

- Untuk detail API, lihat [ListResourceTypes](#) di Referensi AWS CLI Perintah.

## list-resources

Contoh kode berikut menunjukkan cara menggunakan `list-resources`.

## AWS CLI

Untuk membuat daftar sumber daya yang terkait dengan pembagian sumber daya

`list-resources` Contoh berikut mencantumkan semua sumber daya dalam pembagian sumber daya tertentu yang merupakan jenis sumber daya yang ditentukan.

```
aws ram list-resources \
  --resource-type ec2:Subnet \
  --resource-owner SELF \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-
b505-7e2a-420d-6f5d3EXAMPLE
```

Output:

```
{
  "resources": [
    {
      "arn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1f4e15235",
      "type": "ec2:Subnet",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "creationTime": 1565301545.023,
      "lastUpdatedTime": 1565301545.947
    }
  ]
}
```

- Untuk detail API, lihat [ListResources](#) di Referensi AWS CLI Perintah.

### **promote-resource-share-created-from-policy**

Contoh kode berikut menunjukkan cara menggunakan `promote-resource-share-created-from-policy`.

## AWS CLI

Untuk mempromosikan pembagian sumber daya berbasis kebijakan sumber daya ke fungsionalitas penuh dalam RAM AWS



`promote-resource-share-created-from-policy` Contoh berikut mengambil pembagian sumber daya yang Anda buat secara implisit dengan melampirkan kebijakan berbasis sumber daya, dan mengubahnya menjadi berfungsi penuh dengan AWS konsol RAM dan operasi CLI dan API-nya.

```
aws ram promote-resource-share-created-from-policy \  
  --resource-share-arn arn:aws:ram:us-east-1:123456789012:resource-  
share/91fa8429-2d06-4032-909a-90909EXAMPLE
```

Output:

```
{  
  "returnValue": true  
}
```

- Untuk detail API, lihat [PromoteResourceShareCreatedFromPolicy](#) di Referensi AWS CLI Perintah.

## **reject-resource-share-invitation**

Contoh kode berikut menunjukkan cara menggunakan `reject-resource-share-invitation`.

### AWS CLI

Untuk menolak undangan berbagi sumber daya

`reject-resource-share-invitation` Contoh berikut menolak undangan berbagi sumber daya yang ditentukan.

```
aws ram reject-resource-share-invitation \  
  --resource-share-invitation-arn arn:aws:ram:us-west-2:111111111111:resource-  
share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE
```

Output:

```
"resourceShareInvitations": [  
  {  
    "resourceShareInvitationArn": "arn:aws:ram:us-west2-1:111111111111:resource-  
share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE",  
    "resourceShareName": "project-resource-share",
```

```

    "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-share/
    fcb639f0-1449-4744-35bc-a983fEXAMPLE",
    "senderAccountId": "111111111111",
    "receiverAccountId": "222222222222",
    "invitationTimestamp": 1565319592.463,
    "status": "REJECTED"
  }
]

```

- Untuk detail API, lihat [RejectResourceShareInvitation](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke berbagi sumber daya

`tag-resource` Contoh berikut menambahkan kunci tag `project` dan nilai terkait `lima` untuk berbagi sumber daya tertentu.

```

aws ram tag-resource \
  --tags key=project,value=lima \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-
  b505-7e2a-420d-6f5d3EXAMPLE

```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari berbagi sumber daya

`untag-resource` Contoh berikut menghapus kunci `project` tag dan nilai terkait dari berbagi sumber daya tertentu.

```
aws ram untag-resource \  
  --tag-keys project \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-resource-share

Contoh kode berikut menunjukkan cara menggunakan `update-resource-share`.

### AWS CLI

Untuk memperbarui pembagian sumber daya

`update-resource-share` Contoh berikut mengubah pembagian sumber daya yang ditentukan untuk mengizinkan prinsipal eksternal yang tidak ada dalam Organisasi. AWS

```
aws ram update-resource-share \  
  --allow-external-principals \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

Output:

```
{  
  "resourceShare": {  
    "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",  
    "name": "my-resource-share",  
    "owningAccountId": "123456789012",  
    "allowExternalPrincipals": true,  
    "status": "ACTIVE",  
    "creationTime": 1565295733.282,  
    "lastUpdatedTime": 1565303080.023  
  }  
}
```

- Untuk detail API, lihat [UpdateResourceShare](#) di Referensi AWS CLI Perintah.

## Contoh Resource Explorer menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Resource Explorer.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-default-view**

Contoh kode berikut menunjukkan cara menggunakan `associate-default-view`.

AWS CLI

Untuk menetapkan tampilan Resource Explorer sebagai default untuk AWS Region

`associate-default-view` Contoh berikut menetapkan tampilan, seperti yang ditentukan oleh ARN, menjadi tampilan default untuk AWS Wilayah di mana Anda memanggil operasi.

```
aws resource-explorer-2 associate-default-view \
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Menyetel tampilan default di AWS Wilayah](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [AssociateDefaultView](#) di Referensi AWS CLI Perintah.

## batch-get-view

Contoh kode berikut menunjukkan cara menggunakan `batch-get-view`.

### AWS CLI

Untuk mengambil detail tentang beberapa tampilan Resource Explorer

`batch-get-view` Contoh berikut menampilkan rincian tentang dua tampilan yang ditentukan oleh ARN mereka. Gunakan spasi untuk memisahkan beberapa ARN dalam parameter `--view-arn`.

```
aws resource-explorer-2 batch-get-view \
  --view-arns arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222, \
  arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
  "Views": [
    {
      "Filters": {
        "FilterString": "service:ec2"
      },
      "IncludedProperties": [
        {
          "Name": "tags"
        }
      ],
      "LastUpdatedAt": "2022-07-13T21:33:45.249000+00:00",
      "Owner": "123456789012",
      "Scope": "arn:aws:iam::123456789012:root",
      "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-
EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
    },
    {
      "Filters": {
```

```
        "FilterString": ""
      },
      "IncludedProperties": [
        {
          "Name": "tags"
        }
      ],
      "LastUpdatedAt": "2022-07-13T20:34:11.314000+00:00",
      "Owner": "123456789012",
      "Scope": "arn:aws:iam::123456789012:root",
      "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-
Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
    }
  ]
  "Errors": []
}
```

Untuk informasi selengkapnya tentang tampilan, lihat [Tentang tampilan Resource Explorer](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [BatchGetView](#) di Referensi AWS CLI Perintah.

## create-index

Contoh kode berikut menunjukkan cara menggunakan `create-index`.

### AWS CLI

Untuk mengaktifkan Resource Explorer di AWS Region dengan membuat indeks

`create-index` Contoh berikut membuat indeks lokal di AWS Wilayah di mana operasi dipanggil. AWS CLI secara otomatis menghasilkan nilai `client-token` parameter acak dan memasukkannya ke dalam panggilan ke AWS jika Anda tidak menentukan nilai.

```
aws resource-explorer-2 create-index \
  --region us-east-1
```

Output:

```
{
  "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE22222c",
```

```
"CreatedAt": "2022-11-01T20:00:59.149Z",  
"State": "CREATING"  
}
```

Setelah Anda membuat indeks lokal, Anda dapat mengubahnya menjadi indeks agregator untuk akun dengan menjalankan [update-index-type](#) perintah.

Untuk informasi selengkapnya, lihat [Mengaktifkan Penjelajah Sumber Daya di AWS Wilayah untuk mengindeks sumber daya Anda](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [CreateIndex](#) di Referensi AWS CLI Perintah.

## create-view

Contoh kode berikut menunjukkan cara menggunakan `create-view`.

### AWS CLI

Contoh 1: Untuk membuat tampilan tanpa filter untuk indeks di Wilayah AWS

`create-view` Contoh berikut membuat tampilan di AWS Wilayah tertentu yang mengembalikan semua hasil di Wilayah tanpa pemfilteran apa pun. Tampilan menyertakan bidang Tag opsional pada hasil yang dikembalikan. Karena tampilan ini dibuat di Wilayah yang berisi indeks agregator, tampilan ini dapat menyertakan hasil dari semua Wilayah di akun yang berisi indeks Resource Explorer.

```
aws resource-explorer-2 create-view \  
  --view-name My-Main-View \  
  --included-properties Name=tags \  
  --region us-east-1
```

Output:

```
{  
  "View": {  
    "Filters": {  
      "FilterString": ""  
    },  
    "IncludedProperties": [  
      {  
        "Name": "tags"  
      }  
    ]  
  }  
}
```

```

    }
  ],
  "LastUpdatedAt": "2022-07-13T20:34:11.314000+00:00",
  "Owner": "123456789012",
  "Scope": "arn:aws:iam::123456789012:root",
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
}

```

Contoh 2: Untuk membuat tampilan yang hanya menampilkan sumber daya yang terkait dengan Amazon EC2

Berikut ini `create-view` membuat tampilan di AWS Wilayah `us-east-1` yang hanya menampilkan sumber daya tersebut di Wilayah yang terkait dengan layanan Amazon EC2. Tampilan mencakup Tags bidang opsional pada hasil yang dikembalikan. Karena tampilan ini dibuat di Wilayah yang berisi indeks agregator, tampilan ini dapat menyertakan hasil dari semua Wilayah di akun yang berisi indeks Resource Explorer.

```

aws resource-explorer-2 create-view \
  --view-name My-EC2-Only-View \
  --included-properties Name=tags \
  --filters FilterString="service:ec2" \
  --region us-east-1

```

Output:

```

{
  "View": {
    "Filters": {
      "FilterString": "service:ec2"
    },
    "IncludedProperties": [
      {
        "Name": "tags"
      }
    ],
    "LastUpdatedAt": "2022-07-13T21:35:09.059Z",
    "Owner": "123456789012",
    "Scope": "arn:aws:iam::123456789012:root",
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
  }
}

```



```
}  
}
```

Untuk informasi selengkapnya, lihat [Membuat tampilan untuk penelusuran](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [CreateView](#) di Referensi AWS CLI Perintah.

## **delete-index**

Contoh kode berikut menunjukkan cara menggunakan `delete-index`.

### AWS CLI

Untuk mematikan Resource Explorer di AWS Region dengan menghapus indeksnya

`delete-index` Contoh berikut menghapus indeks Resource Explorer yang ditentukan di AWS Wilayah tempat Anda membuat permintaan.

```
aws resource-explorer-2 delete-index \  
  --arn arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE22222 \  
  --region us-west-2
```

Output:

```
{  
  "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE22222",  
  "State": "DELETING"  
}
```

Untuk informasi selengkapnya tentang menghapus indeks, lihat [Menonaktifkan Penjelajah AWS Sumber Daya di AWS Wilayah](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [DeleteIndex](#) di Referensi AWS CLI Perintah.

## **delete-view**

Contoh kode berikut menunjukkan cara menggunakan `delete-view`.

## AWS CLI

Untuk menghapus tampilan Resource Explorer

`delete-view` Contoh berikut menghapus tampilan yang ditentukan oleh ARN nya.

```
aws resource-explorer-2 delete-view \
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-
  View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-
  View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Menghapus tampilan](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [DeleteView](#) di Referensi AWS CLI Perintah.

## **disassociate-default-view**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-default-view`.

## AWS CLI

Untuk menghapus tampilan Resource Explorer default untuk AWS Wilayah

Berikut ini `disassociate-default-view` menghapus tampilan Resource Explorer default untuk AWS Wilayah di mana Anda memanggil operasi. Setelah melakukan operasi ini, semua operasi pencarian di Wilayah harus secara eksplisit menentukan tampilan atau operasi gagal.

```
aws resource-explorer-2 disassociate-default-view
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyetel tampilan default di AWS Wilayah](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [DisassociateDefaultView](#) di Referensi AWS CLI Perintah.

## get-default-view

Contoh kode berikut menunjukkan cara menggunakan `get-default-view`.

### AWS CLI

Untuk mengambil tampilan Resource Explorer yang merupakan tampilan default untuk Region AWS

`get-default-view` Contoh berikut mengambil ARN dari tampilan yang merupakan default untuk Wilayah AWS di mana Anda memanggil operasi.

```
aws resource-explorer-2 get-default-view
```

Output:

```
{
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/default-view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Menyetel tampilan default di AWS Wilayah](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [GetDefaultView](#) di Referensi AWS CLI Perintah.

## get-index

Contoh kode berikut menunjukkan cara menggunakan `get-index`.

### AWS CLI

Contoh 1: Untuk mengambil detail untuk indeks agregator Resource Explorer

`get-index` Contoh berikut menampilkan rincian untuk indeks Resource Explorer di AWS Wilayah tertentu. Karena Region yang ditentukan berisi indeks agregator untuk akun, output mencantumkan Wilayah yang mereplikasi data ke dalam indeks Region ini.

```
aws resource-explorer-2 get-index \
  --region us-east-1
```

Output:

```
{
  "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE111111",
  "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
  "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
  "ReplicatingFrom": [
    "ap-south-1",
    "us-west-2"
  ],
  "State": "ACTIVE",
  "Tags": {},
  "Type": "AGGREGATOR"
}
```

Contoh 2: Untuk mengambil detail untuk indeks lokal Resource Explorer

`get-index` Contoh berikut menampilkan rincian untuk indeks Resource Explorer di AWS Wilayah tertentu. Karena Wilayah yang ditentukan berisi indeks lokal, output mencantumkan Wilayah tempat ia mereplikasi data dari indeks Wilayah ini.

```
aws resource-explorer-2 get-index \
  --region us-west-2
```

Output:

```
{
  "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE222222",
  "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
  "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
  "ReplicatingTo": [
    "us-west-2"
  ],
  "State": "ACTIVE",
  "Tags": {},
  "Type": "LOCAL"
}
```

Untuk informasi selengkapnya tentang indeks, lihat [Memeriksa AWS Wilayah mana yang mengaktifkan Resource Explorer di](#) Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [GetIndex](#) di Referensi AWS CLI Perintah.

## get-view

Contoh kode berikut menunjukkan cara menggunakan `get-view`.

### AWS CLI

Untuk mengambil detail tentang tampilan Resource Explorer

`get-view` Contoh berikut menampilkan rincian tentang tampilan yang ditentukan oleh ARN nya.

```
aws resource-explorer-2 get-view \  
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-  
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{  
  "Tags" : {},  
  "View" : {  
    "Filters" : {  
      "FilterString" : "service:ec2"  
    },  
    "IncludedProperties" : [  
      {  
        "Name" : "tags"  
      }  
    ],  
    "LastUpdatedAt" : "2022-07-13T21:33:45.249Z",  
    "Owner" : "123456789012",  
    "Scope" : "arn:aws:iam::123456789012:root",  
    "ViewArn" : "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-  
Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"  
  }  
}
```

Untuk informasi selengkapnya tentang tampilan, lihat [Tentang tampilan Resource Explorer](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [GetView](#) di Referensi AWS CLI Perintah.

## list-indexes

Contoh kode berikut menunjukkan cara menggunakan `list-indexes`.

## AWS CLI

Untuk membuat daftar AWS Wilayah tempat Resource Explorer memiliki indeks

`list-indexes` Contoh berikut mencantumkan indeks untuk semua Wilayah di mana Resource Explorer memiliki indeks. Respons menentukan jenis setiap indeks, AWS Wilayah, dan ARN-nya.

```
aws resource-explorer-2 list-indexes
```

Output:

```
{
  "Indexes": [
    {
      "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
      "Region": "us-west-2",
      "Type": "AGGREGATOR"
    },
    {
      "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
      "Region": "us-east-1",
      "Type": "LOCAL"
    },
    {
      "Arn": "arn:aws:resource-explorer-2:us-east-2:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE33333",
      "Region": "us-east-2",
      "Type": "LOCAL"
    },
    {
      "Arn": "arn:aws:resource-explorer-2:us-west-1:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE44444",
      "Region": "us-west-1",
      "Type": "LOCAL"
    }
  ]
}
```

Untuk informasi selengkapnya tentang indeks, lihat [Memeriksa AWS Wilayah mana yang mengaktifkan Resource Explorer di](#) Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [ListIndexes](#) di Referensi AWS CLI Perintah.

## list-supported-resource-types

Contoh kode berikut menunjukkan cara menggunakan `list-supported-resource-types`.

### AWS CLI

Untuk membuat daftar AWS Wilayah tempat Resource Explorer memiliki indeks

`list-supported-resource-types` Contoh berikut mencantumkan semua jenis sumber daya yang saat ini didukung oleh &aRexLong;. Contoh respons mencakup `NextToken` nilai, yang menunjukkan bahwa ada lebih banyak output yang tersedia untuk diambil dengan panggilan tambahan.

```
aws resource-explorer-2 list-supported-resource-types \  
  --max-items 10
```

Output:

```
{  
  "ResourceTypes": [  
    {  
      "ResourceType": "cloudfront:cache-policy",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:distribution",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:function",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:origin-access-identity",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:origin-request-policy",  
      "Service": "cloudfront"  
    }  
  ]  
}
```

```

    },
    {
      "ResourceType": "cloudfront:realtime-log-config",
      "Service": "cloudfront"
    },
    {
      "ResourceType": "cloudfront:response-headers-policy",
      "Service": "cloudfront"
    },
    {
      "ResourceType": "cloudwatch:alarm",
      "Service": "cloudwatch"
    },
    {
      "ResourceType": "cloudwatch:dashboard",
      "Service": "cloudwatch"
    },
    {
      "ResourceType": "cloudwatch:insight-rule",
      "Service": "cloudwatch"
    }
  ],
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0="
}

```

Untuk mendapatkan bagian berikutnya dari output, panggil operasi lagi, dan berikan nilai NextToken respons panggilan sebelumnya sebagai nilai untuk `--starting-token`. NextTokenUlangi sampai tidak ada respon.

```

aws resource-explorer-2 list-supported-resource-types \
  --max-items 10 \
  --starting-token
eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0=

```

Output:

```

{
  "ResourceTypes": [
    {
      "ResourceType": "cloudwatch:metric-stream",
      "Service": "cloudwatch"
    },
  ],

```



```
{
  "ResourceType": "dynamodb:table",
  "Service": "dynamodb"
},
{
  "ResourceType": "ec2:capacity-reservation",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:capacity-reservation-fleet",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:client-vpn-endpoint",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:customer-gateway",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:dedicated-host",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:dhcp-options",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:egress-only-internet-gateway",
  "Service": "ec2"
},
{
  "ResourceType": "ec2:elastic-gpu",
  "Service": "ec2"
}
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyMH0="
}
```

Untuk informasi selengkapnya tentang indeks, lihat [Memeriksa AWS Wilayah mana yang mengaktifkan Resource Explorer di](#) Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [ListSupportedResourceTypes](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang dilampirkan ke tampilan atau indeks Resource Explorer

`list-tags-for-resource` Contoh berikut mencantumkan kunci tag dan pasangan nilai yang dilampirkan untuk melihat dengan ARN yang ditentukan. Anda harus memanggil operasi dari AWS Wilayah yang berisi sumber daya.

```
aws resource-explorer-2 list-tags-for-resource \
  --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
  "Tags": {
    "application": "MainCorpApp",
    "department": "1234"
  }
}
```

Untuk informasi selengkapnya tentang menandai tampilan, lihat [Menandai tampilan untuk kontrol akses di Panduan](#) Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## list-views

Contoh kode berikut menunjukkan cara menggunakan `list-views`.

### AWS CLI

Untuk mencantumkan tampilan Resource Explorer yang tersedia di AWS Wilayah

`list-views` Contoh berikut mencantumkan semua tampilan yang tersedia di Wilayah tempat Anda menjalankan operasi.

```
aws resource-explorer-2 list-views
```

**Output:**

```
{
  "Views": [
    "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
    "arn:aws:resource-explorer-2:us-east-1:123456789012:view/Default-All-Resources-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
    "arn:aws:resource-explorer-2:us-east-1:123456789012:view/Production-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE33333"
  ]
}
```

Untuk informasi selengkapnya tentang tampilan, lihat [Tentang tampilan Resource Explorer](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [ListViews](#) di Referensi AWS CLI Perintah.

**search**

Contoh kode berikut menunjukkan cara menggunakan `search`.

**AWS CLI**

Contoh 1: Untuk mencari menggunakan tampilan default

`search` Contoh berikut menampilkan semua sumber daya dalam yang ditentukan yang terkait dengan layanan. Pencarian menggunakan tampilan default untuk Wilayah. Contoh respons mencakup `NextToken` nilai, yang menunjukkan bahwa ada lebih banyak output yang tersedia untuk diambil dengan panggilan tambahan.

```
aws resource-explorer-2 search \
  --query-string "service:iam"
```

**Output:**

```
{
  "Count": {
    "Complete": true,
    "TotalResources": 55
  },
}
```

```

    "NextToken":
    "AG9V0EF1KLEXAMPLE0hJHVwo5chEXAMPLER5XiEpNrgsEXAMPLE...b0Cm0F0ryHEXAMPLE",
    "Resources": [{
        "Arn": "arn:aws:iam::123456789012:policy/service-role/Some-Policy-For-A-
Service-Role",
        "LastReportedAt": "2022-07-21T12:34:42Z",
        "OwningAccountId": "123456789012",
        "Properties": [],
        "Region": "global",
        "ResourceType": "iam:policy",
        "Service": "iam"
    }, {
        "Arn": "arn:aws:iam::123456789012:policy/service-role/Another-Policy-For-A-
Service-Role",
        "LastReportedAt": "2022-07-21T12:34:42Z",
        "OwningAccountId": "123456789012",
        "Properties": [],
        "Region": "global",
        "ResourceType": "iam:policy",
        "Service": "iam"
    }, {
        ... TRUNCATED FOR BREVITY ...
    }],
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/my-default-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}

```

## Contoh 2: Untuk mencari menggunakan tampilan tertentu

searchContoh pencarian berikut menampilkan semua sumber daya ("\*") di AWS Wilayah tertentu yang terlihat melalui tampilan yang ditentukan. Hasilnya hanya mencakup sumber daya yang terkait dengan Amazon EC2 karena filter yang dilampirkan pada tampilan.

```

aws resource-explorer-2 search \
  -- query-string "*" \
  -- view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-view/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222

```

## Output:

```

HTTP/1.1 200 OK
Date: Tue, 01 Nov 2022 20:00:59 GMT
Content-Type: application/json

```

Content-Length: <PayloadSizeBytes>

```
{
  "Count": {
    "Complete": true,
    "TotalResources": 67
  },
  "Resources": [{
    "Arn": "arn:aws:ec2:us-east-1:123456789012:network-acl/acl-1a2b3c4d",
    "LastReportedAt": "2022-07-21T18:52:02Z",
    "OwningAccountId": "123456789012",
    "Properties": [{
      "Data": [{
        "Key": "Department",
        "Value": "AppDevelopment"
      }, {
        "Key": "Environment",
        "Value": "Production"
      }
    ],
    "LastReportedAt": "2021-11-15T14:48:29Z",
    "Name": "tags"
  }],
  "Region": "us-east-1",
  "ResourceType": "ec2:network-acl",
  "Service": "ec2"
}, {
  "Arn": "arn:aws:ec2:us-east-1:123456789012:subnet/subnet-1a2b3c4d",
  "LastReportedAt": "2022-07-21T21:22:23Z",
  "OwningAccountId": "123456789012",
  "Properties": [{
    "Data": [{
      "Key": "Department",
      "Value": "AppDevelopment"
    }, {
      "Key": "Environment",
      "Value": "Production"
    }
  ],
  "LastReportedAt": "2021-07-29T19:02:39Z",
  "Name": "tags"
}],
  "Region": "us-east-1",
  "ResourceType": "ec2:subnet",
  "Service": "ec2"
}, {
```

```

    "Arn": "arn:aws:ec2:us-east-1:123456789012:dhcp-options/dopt-1a2b3c4d",
    "LastReportedAt": "2022-07-21T06:08:53Z",
    "OwningAccountId": "123456789012",
    "Properties": [{
      "Data": [{
        "Key": "Department",
        "Value": "AppDevelopment"
      }, {
        "Key": "Environment",
        "Value": "Production"
      }],
      "LastReportedAt": "2021-11-15T15:11:05Z",
      "Name": "tags"
    }],
    "Region": "us-east-1",
    "ResourceType": "ec2:dhcpoptions",
    "Service": "ec2"
  }, {
    ... TRUNCATED FOR BREVITY ...
  }],
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
}

```

Untuk informasi selengkapnya, lihat [Menggunakan AWS Resource Explorer untuk mencari sumber daya](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [Cari](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai tampilan Resource Explorer

`tag-resource` Contoh berikut menambahkan kunci tag “lingkungan” dengan nilai “produksi” ke tampilan dengan ARN yang ditentukan.

```

aws resource-explorer-2 tag-resource \
  --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View//
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \

```

```
--tags environment=production
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai tampilan untuk kontrol akses](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari tampilan Resource Explorer

`untag-resource` Contoh berikut menghapus tag apa pun dengan nama kunci “lingkungan” dari tampilan dengan ARN yang ditentukan.

```
aws resource-explorer-2 untag-resource \
  --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View//
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \
  --tag-keys environment
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai tampilan untuk kontrol akses](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-index-type

Contoh kode berikut menunjukkan cara menggunakan `update-index-type`.

### AWS CLI

Untuk mengubah jenis indeks Resource Explorer

`update-index-type` Contoh berikut mengonversi indeks yang ditentukan dari jenis `local` ke jenis `aggregator` untuk mengaktifkan kemampuan untuk mencari sumber daya di semua AWS

Wilayah dalam akun. Anda harus mengirim permintaan ke AWS Wilayah yang berisi indeks yang ingin Anda perbarui.

```
aws resource-explorer-2 update-index-type \  
  --arn arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE11111 \  
  --type aggregator \  
  --region us-east-1
```

Output:

```
{  
  "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE11111",  
  "LastUpdatedAt": "2022-07-13T18:41:58.799Z",  
  "State": "updating",  
  "Type": "aggregator"  
}
```

Untuk informasi selengkapnya tentang mengubah jenis indeks, lihat [Mengaktifkan pencarian lintas wilayah dengan membuat indeks agregator](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [UpdateIndexType](#) di Referensi AWS CLI Perintah.

## update-view

Contoh kode berikut menunjukkan cara menggunakan `update-view`.

### AWS CLI

Contoh 1: Untuk memperbarui `IncludedProperties` bidang untuk tampilan Resource Explorer

`update-view` Contoh berikut memperbarui tampilan yang ditentukan dengan menambahkan ``tags`` ke opsional ``IncludedProperties``. Setelah menjalankan operasi ini, operasi pencarian yang menggunakan tampilan ini menyertakan informasi tentang tag yang dilampirkan ke sumber daya yang muncul di hasil.

```
aws resource-explorer-2 update-view \  
  --included-properties Name=tags \  
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/  
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```



**Output:**

```
{
  "View": {
    "Filters": {
      "FilterString": ""
    },
    "IncludedProperties": [
      {
        "Name": "tags"
      }
    ],
    "LastUpdatedAt": "2022-07-19T17:41:21.710000+00:00",
    "Owner": "123456789012",
    "Scope": "arn:aws:iam::123456789012:root",
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
  }
}
```

**Contoh 2:** Untuk memperbarui filter yang dilampirkan ke tampilan

`update-view` Contoh berikut memperbarui tampilan yang ditentukan untuk menggunakan filter yang membatasi hasil hanya pada jenis sumber daya yang terkait dengan layanan Amazon EC2.

```
aws resource-explorer-2 update-view \
  --filters FilterString="service:ec2" \
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```

**Output:**

```
{
  "View": {
    "Filters": {
      "FilterString": "service:ec2"
    },
    "IncludedProperties": [],
    "LastUpdatedAt": "2022-07-19T17:41:21.710000+00:00",
    "Owner": "123456789012",
    "Scope": "arn:aws:iam::123456789012:root",
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
  }
}
```

```
}  
}
```

Untuk informasi selengkapnya tentang tampilan, lihat [Tentang tampilan Resource Explorer](#) di Panduan Pengguna AWS Resource Explorer.

- Untuk detail API, lihat [UpdateView](#) di Referensi AWS CLI Perintah.

## Contoh Resource Groups menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Resource Groups.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

### Tindakan

#### **create-group**

Contoh kode berikut menunjukkan cara menggunakan `create-group`.

### AWS CLI

Contoh 1: Untuk membuat grup sumber daya berbasis tag

`create-group` Contoh berikut membuat grup sumber daya berbasis tag dari instans Amazon EC2 di wilayah saat ini. Ini didasarkan pada kueri untuk sumber daya yang ditandai dengan kunci `Name`, dan nilainya `WebServers`. Nama grup adalah `tbq-WebServer`. Kueri berada dalam file JSON terpisah yang diteruskan ke perintah.

```
aws resource-groups create-group \
  --name tbq-WebServer \
  --resource-query file://query.json
```

Isi dari query.json:

```
{
  "Type": "TAG_FILTERS_1_0",
  "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [ { \"Key\": \"Name\", \"Values\": [ \"WebServers\" ] } ] }"
```

Output:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
    "Name": "tbq-WebServer"
  },
  "ResourceQuery": {
    "Type": "TAG_FILTERS_1_0",
    "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [ { \"Key\": \"Name\", \"Values\": [ \"WebServers\" ] } ] }"
```

Contoh 2: Untuk membuat grup sumber CloudFormation daya berbasis tumpukan

create-group Contoh berikut membuat grup sumber daya AWS CloudFormation berbasis tumpukan bernama. sampleCFNstackgroup Kueri mencakup semua sumber daya dalam CloudFormation tumpukan tertentu yang didukung oleh AWS Resource Groups.

```
aws resource-groups create-group \
  --name cbq-CFNstackgroup \
  --resource-query file://query.json
```

Isi dari query.json:

```
{
```

```

    "Type": "CLOUDFORMATION_STACK_1_0",
    "Query": "{\"ResourceTypeFilters\": [\"AWS::AllSupported\"], \"StackIdentifier\": \"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-z39z-11z8-97z5-500z212zz6fz\"}"
  }

```

### Output:

```

{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-CFNstackgroup",
    "Name": "cbq-CFNstackgroup"
  },
  "ResourceQuery": {
    "Type": "CLOUDFORMATION_STACK_1_0",
    "Query": "{\"ResourceTypeFilters\": [\"AWS::AllSupported\"], \"StackIdentifier\": \"arn:aws:cloudformation:us-east-2:123456789012:stack/MyCFNStack/1415z9z0-z39z-11z8-97z5-500z212zz6fz\"}"
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat Grup](#) di Panduan Pengguna AWS Resource Groups.

- Untuk detail API, lihat [CreateGroup](#) di Referensi AWS CLI Perintah.

## delete-group

Contoh kode berikut menunjukkan cara menggunakan `delete-group`.

### AWS CLI

Untuk memperbarui deskripsi untuk grup sumber daya

`delete-group` Contoh berikut memperbarui grup sumber daya yang ditentukan.

```

aws resource-groups delete-group \
  --group-name tbq-WebServer

```

### Output:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:1234567890:group/tbq-WebServer",
    "Name": "tbq-WebServer"
  }
}
```

Untuk selengkapnya, lihat [Menghapus Grup](#) di Panduan Pengguna AWS Resource Groups.

- Untuk detail API, lihat [DeleteGroup](#) di Referensi AWS CLI Perintah.

## get-group-query

Contoh kode berikut menunjukkan cara menggunakan `get-group-query`.

### AWS CLI

Untuk mendapatkan kueri yang dilampirkan ke grup sumber daya

`get-group-query` Contoh berikut menampilkan query yang dilampirkan ke kelompok sumber daya tertentu.

```
aws resource-groups get-group-query \
  --group-name tbq-WebServer
```

Output:

```
{
  "GroupQuery": {
    "GroupName": "tbq-WebServer",
    "ResourceQuery": {
      "Type": "TAG_FILTERS_1_0",
      "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [{\"Key\": \"Name\", \"Values\": [\"WebServers\"]}]}"
    }
  }
}
```

- Untuk detail API, lihat [GetGroupQuery](#) di Referensi AWS CLI Perintah.

## get-group

Contoh kode berikut menunjukkan cara menggunakan `get-group`.

### AWS CLI

Untuk mendapatkan informasi tentang grup sumber daya

`get-group` Contoh berikut menampilkan rincian tentang kelompok sumber daya tertentu. Untuk mendapatkan kueri yang dilampirkan ke grup, gunakan `get-group-query`.

```
aws resource-groups get-group \  
  --group-name tbq-WebServer
```

Output:

```
{  
  "Group": {  
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-  
WebServer",  
    "Name": "tbq-WebServer",  
    "Description": "A tag-based query resource group of WebServers."  
  }  
}
```

- Untuk detail API, lihat [GetGroup](#) di Referensi AWS CLI Perintah.

## get-tags

Contoh kode berikut menunjukkan cara menggunakan `get-tags`.

### AWS CLI

Untuk mengambil tag yang dilampirkan ke grup sumber daya

`get-tags` Contoh berikut menampilkan kunci tag dan pasangan nilai yang dilampirkan ke grup sumber daya tertentu (grup itu sendiri, bukan anggotanya).

```
aws resource-groups get-tags \  
  --arn arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer
```

**Output:**

```
{
  "Arn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
  "Tags": {
    "QueryType": "tags",
    "QueryResources": "ec2-instances"
  }
}
```

- Untuk detail API, lihat [GetTags](#) di Referensi AWS CLI Perintah.

**list-group-resources**

Contoh kode berikut menunjukkan cara menggunakan `list-group-resources`.

**AWS CLI**

Untuk membuat daftar semua sumber daya dalam grup sumber daya

Contoh 1: `list-resource-groups` Contoh berikut mencantumkan semua sumber daya yang merupakan bagian dari grup sumber daya yang ditentukan.

```
aws resource-groups list-group-resources \
  --group-name tbq-WebServer
```

**Output:**

```
{
  "ResourceIdentifiers": [
    {
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-09f77fa38c12345ab",
      "ResourceType": "AWS::EC2::Instance"
    }
  ]
}
```

Contoh 2: Contoh berikut mencantumkan semua sumber daya dalam grup yang juga memiliki 'tipe sumber daya' dari ':EC2: :Instance'AWS. :

```
aws resource-groups list-group-resources --group-name tbq- WebServer --filters name=Resource-
type, Values=: :EC2: :Instance AWS
```

- Untuk detail API, lihat [ListGroupResources](#) di Referensi AWS CLI Perintah.

## list-groups

Contoh kode berikut menunjukkan cara menggunakan `list-groups`.

### AWS CLI

Untuk daftar grup sumber daya yang tersedia

`list-groups` Contoh berikut menampilkan daftar semua kelompok sumber daya.

```
aws resource-groups list-groups
```

Output:

```
{
  "GroupIdentifiers": [
    {
      "GroupName": "tbq-WebServer",
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer3"
    },
    {
      "GroupName": "cbq-CFNStackQuery",
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNStackQuery"
    }
  ],
  "Groups": [
    {
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer",
      "Name": "tbq-WebServer"
    },
    {
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNStackQuery",
      "Name": "cbq-CFNStackQuery"
    }
  ]
}
```



```
]
}
```

- Untuk detail API, lihat [ListGroups](#) di Referensi AWS CLI Perintah.

## list-resource-groups

Contoh kode berikut menunjukkan cara menggunakan `list-resource-groups`.

### AWS CLI

Untuk membuat daftar semua sumber daya dalam grup sumber daya

`list-resource-groups` Contoh berikut mencantumkan semua sumber daya yang merupakan bagian dari kelompok sumber daya yang ditentukan.

```
aws resource-groups list-group-resources \
  --group-name tbq-WebServer
```

Output:

```
{
  "ResourceIdentifiers": [
    {
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-09f77fa38c12345ab",
      "ResourceType": "AWS::EC2::Instance"
    }
  ]
}
```

- Untuk detail API, lihat [ListResourceGroups](#) di Referensi AWS CLI Perintah.

## put-group-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-group-configuration`.

### AWS CLI

Untuk melampirkan konfigurasi layanan ke grup sumber daya

Contoh 1: `put-group-configuration` Contoh berikut menentukan bahwa grup sumber daya hanya berisi reservasi kapasitas Amazon EC2 untuk instans dalam atau keluarga. C5 M5

```
aws resource-groups put-group-configuration \  
  --group MyTestGroup \  
  --configuration file:///config.json
```

Isi dari `config.json`:

```
[  
  {  
    "Type": "AWS::EC2::HostManagement",  
    "Parameters": [  
      {  
        "Name": "allowed-host-families",  
        "Values": [ "c5", "m5" ]  
      },  
      {  
        "Name": "any-host-based-license-configuration",  
        "Values": [ "true" ]  
      }  
    ]  
  },  
  {  
    "Type": "AWS::ResourceGroups::Generic",  
    "Parameters": [  
      {  
        "Name": "allowed-resource-types",  
        "Values": [ "AWS::EC2::Host" ]  
      },  
      {  
        "Name": "deletion-protection",  
        "Values": [ "UNLESS_EMPTY" ]  
      }  
    ]  
  }  
]
```

Perintah ini tidak menghasilkan output jika berhasil.

Untuk informasi selengkapnya, lihat [Konfigurasi layanan untuk grup sumber daya](#) di Panduan Referensi API Resource Groups.

- Untuk detail API, lihat [PutGroupConfiguration](#) di Referensi AWS CLI Perintah.

## search-resources

Contoh kode berikut menunjukkan cara menggunakan `search-resources`.

### AWS CLI

Untuk menemukan sumber daya yang cocok dengan kueri

`search-resources` Contoh berikut mengambil daftar semua AWS sumber daya yang cocok dengan query yang ditentukan.

```
aws resource-groups search-resources \
  --resource-query file://query.json
```

Isi dari `query.json`:

```
{
  "Type": "TAG_FILTERS_1_0",
  "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [ { \"Key\": \"Patch Group\", \"Values\": [ \"Dev\" ] } ]}"
```

Output:

```
{
  "ResourceIdentifiers": [
    {
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-01a23bc45d67890ef",
      "ResourceType": "AWS::EC2::Instance"
    }
  ]
}
```

- Untuk detail API, lihat [SearchResources](#) di Referensi AWS CLI Perintah.

## tag

Contoh kode berikut menunjukkan cara menggunakan `tag`.

## AWS CLI

Untuk melampirkan tag ke grup sumber daya

tagContoh berikut melampirkan kunci tag yang ditentukan dan pasangan nilai ke grup sumber daya tertentu (grup itu sendiri, bukan anggotanya).

```
aws resource-groups tag \  
  --tags QueryType=tags,QueryResources=ec2-instances \  
  --arn arn:aws:resource-groups:us-west-2:128716708097:group/tbq-WebServer
```

Output:

```
{  
  "Arn": "arn:aws:resource-groups:us-west-2:128716708097:group/tbq-WebServer",  
  "Tags": {  
    "QueryType": "tags",  
    "QueryResources": "ec2-instances"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola tag](#) di Panduan Pengguna AWS Resource Groups.

- Untuk detail API, lihat [Tag](#) di Referensi AWS CLI Perintah.

## untag

Contoh kode berikut menunjukkan cara menggunakan untag.

## AWS CLI

Untuk menghapus tag dari grup sumber daya

untagsContoh berikut menghapus tag apa pun dengan kunci yang ditentukan dari grup sumber daya itu sendiri, bukan anggotanya.

```
aws resource-groups untag \  
  --arn arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer \  
  --keys QueryType
```

Output:

```
{
  "Arn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
  "Keys": [
    "QueryType"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola tag](#) di Panduan Pengguna AWS Resource Groups.

- Untuk detail API, lihat [Untag](#) di Referensi AWS CLI Perintah.

## update-group-query

Contoh kode berikut menunjukkan cara menggunakan `update-group-query`.

### AWS CLI

Contoh 1: Untuk memperbarui kueri untuk grup sumber daya berbasis tag

`update-group-query` Contoh berikut memperbarui kueri yang dilampirkan ke grup sumber daya berbasis tag yang ditentukan.

```
aws resource-groups update-group-query \
  --group-name tbq-WebServer \
  --resource-query '{"Type":"TAG_FILTERS_1_0", "Query":{"ResourceTypeFilters\":[\
  "AWS::EC2::Instance"],\
  "TagFilters\":[{"Key\":"Name", "Values\":[\
  "WebServers"}]}}'
```

Output:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-east-2:123456789012:group/tbq-WebServer",
    "Name": "tbq-WebServer"
  },
  "ResourceQuery": {
    "Type": "TAG_FILTERS_1_0",
    "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":[\
  {\"Key\":"Name", \"Values\":[\"WebServers\"}]}}"
```

```
}

```

Untuk informasi selengkapnya, lihat [Memperbarui Grup](#) di Panduan Pengguna AWS Resource Groups.

Contoh 2: Untuk memperbarui kueri untuk grup sumber daya CloudFormation berbasis tumpukan update-group-queryContoh berikut memperbarui kueri yang dilampirkan ke grup sumber daya AWS CloudFormation berbasis tumpukan yang ditentukan.

```
aws resource-groups update-group-query \
  --group-name cbq-CFNstackgroup \
  --resource-query '{"Type": "CLOUDFORMATION_STACK_1_0", "Query":
  "{\\"ResourceTypeFilters\\": [\\"AWS::AllSupported\\"], \\"StackIdentifier\\":
  \\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
  z39z-11z8-97z5-500z212zz6fz\\"}"}
```

Output:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
    CFNstackgroup",
    "Name": "cbq-CFNstackgroup"
  },
  "ResourceQuery": {
    "Type": "CLOUDFORMATION_STACK_1_0",
    "Query": "{\\"ResourceTypeFilters\\": [\\"AWS::AllSupported\\"], \\"StackIdentifier
    \": \\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
    z39z-11z8-97z5-500z212zz6fz\\"}"
  }
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Grup](#) di Panduan Pengguna AWS Resource Groups.

- Untuk detail API, lihat [UpdateGroupQuery](#) di Referensi AWS CLI Perintah.

## update-group

Contoh kode berikut menunjukkan cara menggunakan update-group.

## AWS CLI

Untuk memperbarui deskripsi untuk grup sumber daya

update-groupContoh berikut memperbarui deskripsi untuk grup sumber daya yang ditentukan.

```
aws resource-groups update-group \  
  --group-name tbq-WebServer \  
  --description "Resource group for all web server resources."
```

Output:

```
{  
  "Group": {  
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-  
WebServer",  
    "Name": "tbq-WebServer"  
    "Description": "Resource group for all web server resources."  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Grup](#) di Panduan Pengguna AWS Resource Groups.

- Untuk detail API, lihat [UpdateGroup](#) di Referensi AWS CLI Perintah.

## Contoh API Penandaan Resource Groups menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan API Penandaan AWS Command Line Interface With Resource Groups.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **get-resources**

Contoh kode berikut menunjukkan cara menggunakan `get-resources`.

## AWS CLI

Untuk mendapatkan daftar sumber daya yang ditandai

`get-resources` Contoh berikut menampilkan daftar sumber daya di akun yang ditandai dengan nama kunci dan nilai yang ditentukan.

```
aws resourcegroupstaggingapi get-resources \
  --tag-filters Key=Environment,Values=Production \
  --tags-per-page 100
```

## Output:

```
{
  "ResourceTagMappingList": [
    {
      "ResourceARN": " arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-7sbz2Kz0",
      "Tags": [
        {
          "Key": "Environment",
          "Value": "Production"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [GetResources](#) di Referensi API Penandaan Resource Groups.

- Untuk detail API, lihat [GetResources](#) di Referensi AWS CLI Perintah.



## get-tag-keys

Contoh kode berikut menunjukkan cara menggunakan `get-tag-keys`.

### AWS CLI

Untuk mendapatkan daftar semua kunci tag

`get-tag-keys` Contoh berikut mengambil daftar semua nama kunci tag yang digunakan oleh sumber daya di akun.

```
aws resourcegroupstaggingapi get-tag-keys
```

Output:

```
{
  "TagKeys": [
    "Environment",
    "CostCenter",
    "Department"
  ]
}
```

Untuk informasi selengkapnya, lihat [GetTagKeys](#) di Referensi API Penandaan Resource Groups.

- Untuk detail API, lihat [GetTagKeys](#) di Referensi AWS CLI Perintah.

## get-tag-values

Contoh kode berikut menunjukkan cara menggunakan `get-tag-values`.

### AWS CLI

Untuk mendapatkan daftar semua nilai tag

`get-tag-values` Contoh berikut menampilkan semua nilai yang digunakan untuk kunci yang ditentukan untuk semua sumber daya di

```
aws resourcegroupstaggingapi get-tag-values \
  --key=Environment
```

Output:

```
{
  "TagValues": [
    "Alpha",
    "Gamma",
    "Production"
  ]
}
```

Untuk informasi selengkapnya, lihat [GetTagValues](#) di Referensi API Penandaan Resource Groups.

- Untuk detail API, lihat [GetTagValues](#) di Referensi AWS CLI Perintah.

## tag-resources

Contoh kode berikut menunjukkan cara menggunakan `tag-resources`.

### AWS CLI

Untuk melampirkan tag ke sumber daya

`tag-resources` Contoh berikut menandai sumber daya yang ditentukan dengan nama kunci dan nilai.

```
aws resourcegroupstaggingapi tag-resources \
  --resource-arn-list arn:aws:s3:::MyProductionBucket \
  --tags Environment=Production,CostCenter=1234
```

Output:

```
{
  "FailedResourcesMap": {}
}
```

Untuk informasi selengkapnya, lihat [TagResources](#) di Referensi API Penandaan Resource Groups.

- Untuk detail API, lihat [TagResources](#) di Referensi AWS CLI Perintah.

## untag-resources

Contoh kode berikut menunjukkan cara menggunakan `untag-resources`.

## AWS CLI

Untuk menghapus tag dari sumber daya

`untag-resources` Contoh berikut menghapus kunci tag yang ditentukan dan nilai terkait dari sumber daya yang ditentukan.

```
aws resourcegroupstaggingapi untag-resources \  
  --resource-arn-list arn:aws:s3:::awsexamplebucket \  
  --tag-keys Environment CostCenter
```

Output:

```
{  
  "FailedResourcesMap": {}  
}
```

Untuk informasi selengkapnya, lihat [UntagResources](#) di Referensi API Penandaan Resource Groups.

- Untuk detail API, lihat [UntagResources](#) di Referensi AWS CLI Perintah.

## AWS RoboMaker contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface with AWS RoboMaker.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

## Tindakan

### **batch-describe-simulation-job**

Contoh kode berikut menunjukkan cara menggunakan `batch-describe-simulation-job`.

#### AWS CLI

Untuk menggambarkan pekerjaan simulasi secara batch

`batch-describe-simulation-job` Contoh berikut mengambil rincian untuk tiga pekerjaan simulasi yang ditentukan.

Perintah:

```
aws robomaker batch-describe-simulation-job \  
--job arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-66bbb3gpxm8x  
arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-p0cpdrrwng2n  
arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-g8h6tgmlmblgw
```

Output:

```
{  
  "jobs": [  
    {  
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/  
sim-66bbb3gpxm8x",  
      "status": "Completed",  
      "lastUpdatedAt": 1548959178.0,  
      "failureBehavior": "Continue",  
      "clientRequestToken": "6020408e-b05c-4310-9f13-4ed71c5221ed",  
      "outputLocation": {  
        "s3Bucket": "awsrobomakerobjecttracker-1111111111-  
bundlesbucket-2lk584kiq1oa",  
        "s3Prefix": "output"  
      },  
      "maxJobDurationInSeconds": 3600,  
      "simulationTimeMillis": 0,  
      "iamRole": "arn:aws:iam::111111111111:role/  
AWSRoboMakerObjectTracker-154895-SimulationJobRole-14D5ASA7PQE3A",  
      "simulationApplications": [  
        {
```

```

        "application": "arn:aws:robomaker:us-
west-2:111111111111:simulation-application/
AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq/1548959170096",
        "applicationVersion": "$LATEST",
        "launchConfig": {
            "packageName": "object_tracker_simulation",
            "launchFile": "local_training.launch",
            "environmentVariables": {
                "MARKOV_PRESET_FILE": "object_tracker.py",
                "MODEL_S3_BUCKET": "awsrobomakerobjecttracker-111111111-
bundlesbucket-21k584kiq1oa",
                "MODEL_S3_PREFIX": "model-store",
                "ROS_AWS_REGION": "us-west-2"
            }
        }
    },
    "tags": {},
    "vpcConfig": {
        "subnets": [
            "subnet-716dd52a",
            "subnet-43c22325",
            "subnet-3f526976"
        ],
        "securityGroups": [
            "sg-3fb40545"
        ],
        "vpcId": "vpc-99895eff",
        "assignPublicIp": true
    }
},
{
    "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
p0cpcdrwng2n",
    "status": "Completed",
    "lastUpdatedAt": 1548168817.0,
    "failureBehavior": "Continue",
    "clientRequestToken": "e4a23e75-f9a7-411d-835f-21881c82c58b",
    "outputLocation": {
        "s3Bucket": "awsrobomakercloudwatch-111111111111-
bundlesbucket-14e5s9jvwtmv7",
        "s3Prefix": "output"
    },
    "maxJobDurationInSeconds": 3600,

```

```
    "simulationTimeMillis": 0,
    "iamRole": "arn:aws:iam::111111111111:role/
AWSRoboMakerCloudWatch-154766341-SimulationJobRole-G00BWTQ8YBG6",
    "robotApplications": [
      {
        "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/AWSRoboMakerCloudWatch-1547663411642_NZbpqEJ3T/1547663517377",
        "applicationVersion": "$LATEST",
        "launchConfig": {
          "packageName": "cloudwatch_robot",
          "launchFile": "await_commands.launch",
          "environmentVariables": {
            "LAUNCH_ID": "1548168752173",
            "ROS_AWS_REGION": "us-west-2"
          }
        }
      }
    ],
    "simulationApplications": [
      {
        "application": "arn:aws:robomaker:us-
west-2:111111111111:simulation-application/
AWSRoboMakerCloudWatch-1547663411642_0LI6D1h6/1547663521470",
        "applicationVersion": "$LATEST",
        "launchConfig": {
          "packageName": "cloudwatch_simulation",
          "launchFile": "bookstore_turtlebot_navigation.launch",
          "environmentVariables": {
            "LAUNCH_ID": "1548168752173",
            "ROS_AWS_REGION": "us-west-2",
            "TURTLEBOT3_MODEL": "waffle_pi"
          }
        }
      }
    ],
    "tags": {},
    "vpcConfig": {
      "subnets": [
        "subnet-716dd52a",
        "subnet-43c22325",
        "subnet-3f526976"
      ],
      "securityGroups": [
        "sg-3fb40545"
      ]
    }
  }
}
```

```
    ],
    "vpcId": "vpc-99895eff",
    "assignPublicIp": true
  }
},
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
g8h6tglmblgw",
  "status": "Canceled",
  "lastUpdatedAt": 1546543442.0,
  "failureBehavior": "Fail",
  "clientRequestToken": "d796bbb4-2a2c-1abc-f2a9-0d9e547d853f",
  "outputLocation": {
    "s3Bucket": "sample-bucket",
    "s3Prefix": "SimulationLog_115490482698"
  },
  "maxJobDurationInSeconds": 28800,
  "simulationTimeMillis": 0,
  "iamRole": "arn:aws:iam::111111111111:role/RoboMakerSampleTheFirst",
  "robotApplications": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/RoboMakerHelloWorldRobot/1546541208251",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "packageName": "hello_world_robot",
        "launchFile": "rotate.launch"
      }
    }
  ],
  "simulationApplications": [
    {
      "application": "arn:aws:robomaker:us-
west-2:111111111111:simulation-application/
RoboMakerHelloWorldSimulation/1546541198985",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "packageName": "hello_world_simulation",
        "launchFile": "empty_world.launch"
      }
    }
  ],
  "tags": {}
}
```

```

    ],
    "unprocessedJobs": []
  }

```

- Untuk detail API, lihat [BatchDescribeSimulationJob](#) di Referensi AWS CLI Perintah.

## cancel-simulation-job

Contoh kode berikut menunjukkan cara menggunakan `cancel-simulation-job`.

### AWS CLI

Untuk membatalkan pekerjaan simulasi

`cancel-simulation-job` Contoh berikut membatalkan pekerjaan simulasi yang ditentukan.

```

aws robomaker cancel-simulation-job \
  --job arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-66bbb3gpxm8x

```

- Untuk detail API, lihat [CancelSimulationJob](#) di Referensi AWS CLI Perintah.

## create-deployment-job

Contoh kode berikut menunjukkan cara menggunakan `create-deployment-job`.

### AWS CLI

Untuk membuat pekerjaan penerapan

Contoh ini membuat pekerjaan penerapan untuk armada MyFleet. Ini termasuk variabel lingkungan bernama "ENVIRONMENT". Ini juga melampirkan tag bernama "Region".

Perintah:

```

aws robomaker create-deployment-job --deployment-config
  concurrentDeploymentPercentage=20, failureThresholdPercentage=25
  --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
  Trek/1539894765711 --tags Region=West --deployment-application-configs
  application=arn:aws:robomaker:us-west-2:111111111111:robot-application/
  RoboMakerVoiceInteractionRobot/1546537110575, applicationVersion=1, launchConfig={environmentV

```



**Output:**

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/sim-0974h36s4v0t",
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
  "status": "Pending",
  "deploymentApplicationConfigs": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-application/RoboMakerVoiceInteractionRobot/1546537110575",
      "applicationVersion": "1",
      "launchConfig": {
        "packageName": "voice_interaction_robot",
        "launchFile": "await_commands.launch",
        "environmentVariables": {
          "ENVIRONMENT": "Beta"
        }
      }
    }
  ],
  "createdAt": 1550770236.0,
  "deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
  },
  "tags": {
    "Region": "West"
  }
}
```

- Untuk detail API, lihat [CreateDeploymentJob](#) di Referensi AWS CLI Perintah.

**create-fleet**

Contoh kode berikut menunjukkan cara menggunakan `create-fleet`.

**AWS CLI**

Untuk membuat armada

Contoh ini menciptakan armada. Ini melampirkan tag bernama Region.

**Perintah:**

```
aws robomaker create-fleet --name MyFleet --tags Region=East
```

**Output:**

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyOtherFleet/1550771394395",
  "name": "MyFleet",
  "createdAt": 1550771394.0,
  "tags": {
    "Region": "East"
  }
}
```

- Untuk detail API, lihat [CreateFleet](#) di Referensi AWS CLI Perintah.

**create-robot-application-version**

Contoh kode berikut menunjukkan cara menggunakan `create-robot-application-version`.

**AWS CLI**

Untuk membuat versi aplikasi robot

Contoh ini membuat versi aplikasi robot.

**Perintah:**

```
aws robomaker create-robot-application-version --application arn:aws:robomaker:us-
west-2:111111111111:robot-application/MyRobotApplication/1551201873931
```

**Output:**

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/
MyRobotApplication/1551201873931",
  "name": "MyRobotApplication",
  "version": "1",
```

```
"sources": [
  {
    "s3Bucket": "my-bucket",
    "s3Key": "my-robot-application.tar.gz",
    "etag": "f8cf5526f1c6e7b3a72c3ed3f79c5493-70",
    "architecture": "ARMHF"
  }
],
"robotSoftwareSuite": {
  "name": "ROS",
  "version": "Kinetic"
},
"lastUpdatedAt": 1551201873.0,
"revisionId": "9986bb8d-a695-4ab4-8810-9f4a74d1aa00"
"tags": {}
}
```

- Untuk detail API, lihat [CreateRobotApplicationVersion](#) di Referensi AWS CLI Perintah.

## create-robot-application

Contoh kode berikut menunjukkan cara menggunakan `create-robot-application`.

### AWS CLI

Untuk membuat aplikasi robot

Contoh ini membuat aplikasi robot.

Perintah:

```
aws robomaker create-robot-application --name MyRobotApplication --sources
s3Bucket=my-bucket,s3Key=my-robot-application.tar.gz,architecture=X86_64 --robot-
software-suite name=ROS,version=Kinetic
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/
MyRobotApplication/1551201873931",
  "name": "MyRobotApplication",
```

```
"version": "$LATEST",
"sources": [
  {
    "s3Bucket": "my-bucket",
    "s3Key": "my-robot-application.tar.gz",
    "architecture": "ARMHF"
  }
],
"robotSoftwareSuite": {
  "name": "ROS",
  "version": "Kinetic"
},
"lastUpdatedAt": 1551201873.0,
"revisionId": "1f3cb539-9239-4841-a656-d3efcffa07e1",
"tags": {}
}
```

- Untuk detail API, lihat [CreateRobotApplication](#) di Referensi AWS CLI Perintah.

## create-robot

Contoh kode berikut menunjukkan cara menggunakan `create-robot`.

### AWS CLI

Untuk membuat robot

Contoh ini menciptakan robot. Ini menggunakan arsitektur ARMHF. Ini juga melampirkan tag bernama Region.

Perintah:

```
aws robomaker create-robot --name MyRobot --architecture ARMHF --greengrass-group-id
0f728a3c-7dbf-4a3e-976d-d16a8360caba --tags Region=East
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398",
  "name": "MyRobot",
  "createdAt": 1550772325.0,
```

```
"greengrassGroupId": "0f728a3c-7dbf-4a3e-976d-d16a8360caba",
"architecture": "ARMHF",
"tags": {
  "Region": "East"
}
}
```

- Untuk detail API, lihat [CreateRobot](#) di Referensi AWS CLI Perintah.

## create-simulation-application-version

Contoh kode berikut menunjukkan cara menggunakan `create-simulation-application-version`.

### AWS CLI

Untuk membuat versi aplikasi simulasi

Contoh ini membuat versi aplikasi robot.

Perintah:

```
aws robomaker create-simulation-application-version --application
arn:aws:robomaker:us-west-2:111111111111:robot-application/
MySimulationApplication/1551203427605
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MyRobotApplication/1551203427605",
  "name": "MyRobotApplication",
  "version": "1",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-simulation-application.tar.gz",
      "etag": "00d8a94ff113856688c4fce618ae0f45-94",
      "architecture": "X86_64"
    }
  ],
}
```

```
"simulationSoftwareSuite": {
  "name": "Gazebo",
  "version": "7"
},
"robotSoftwareSuite": {
  "name": "ROS",
  "version": "Kinetic"
},
"renderingEngine": {
  "name": "OGRE",
  "version": "1.x"
},
"lastUpdatedAt": 1551203853.0,
"revisionId": "ee753e53-519c-4d37-895d-65e79bcd1914",
"tags": {}
}
```

- Untuk detail API, lihat [CreateSimulationApplicationVersion](#) di Referensi AWS CLI Perintah.

## create-simulation-application

Contoh kode berikut menunjukkan cara menggunakan `create-simulation-application`.

### AWS CLI

Untuk membuat aplikasi simulasi

Contoh ini membuat aplikasi simulasi.

Perintah:

```
aws robomaker create-simulation-application --name MyRobotApplication --sources
s3Bucket=my-bucket,s3Key=my-simulation-application.tar.gz,architecture=ARMHF
--robot-software-suite name=ROS,version=Kinetic --simulation-software-suite
name=Gazebo,version=7 --rendering-engine name=OGRE,version=1.x
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MyRobotApplication/1551203301792",
  "name": "MyRobotApplication",
```

```
"version": "$LATEST",
"sources": [
  {
    "s3Bucket": "my-bucket",
    "s3Key": "my-simulation-application.tar.gz",
    "architecture": "X86_64"
  }
],
"simulationSoftwareSuite": {
  "name": "Gazebo",
  "version": "7"
},
"robotSoftwareSuite": {
  "name": "ROS",
  "version": "Kinetic"
},
"renderingEngine": {
  "name": "OGRE",
  "version": "1.x"
},
"lastUpdatedAt": 1551203301.0,
"revisionId": "ee753e53-519c-4d37-895d-65e79bcd1914",
"tags": {}
}
```

- Untuk detail API, lihat [CreateSimulationApplication](#) di Referensi AWS CLI Perintah.

## create-simulation-job

Contoh kode berikut menunjukkan cara menggunakan `create-simulation-job`.

### AWS CLI

Untuk membuat pekerjaan simulasi

Contoh ini menciptakan pekerjaan simulasi. Ini menggunakan aplikasi robot dan aplikasi simulasi.

Perintah:

```
aws robomaker create-simulation-job --max-job-duration-
in-seconds 3600 --iam-role arn:aws:iam::111111111111:role/
AWSRoboMakerCloudWatch-154766341-SimulationJobRole-G00BWTQ8YBG6 --robot-
applications application=arn:aws:robomaker:us-west-2:111111111111:robot-application/
```

```
MyRobotApplication/1551203485821,launchConfig={packageName=hello_world_robot,launchFile=rotate.launch}
--simulation-applications application=arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MySimulationApplication/1551203427605,launchConfig={packageName=hello_world_simulation,launchFile=empty_world.launch}
--tags Region=North
```

## Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-w7m68wpr05h8",
  "status": "Pending",
  "lastUpdatedAt": 1551213837.0,
  "failureBehavior": "Fail",
  "clientRequestToken": "b283ccce-e468-43ee-8642-be76a9d69f15",
  "maxJobDurationInSeconds": 3600,
  "simulationTimeMillis": 0,
  "iamRole": "arn:aws:iam::111111111111:role/MySimulationRole",
  "robotApplications": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "packageName": "hello_world_robot",
        "launchFile": "rotate.launch"
      }
    }
  ],
  "simulationApplications": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "packageName": "hello_world_simulation",
        "launchFile": "empty_world.launch"
      }
    }
  ],
  "tags": {
    "Region": "North"
  }
}
```



- Untuk detail API, lihat [CreateSimulationJob](#) di Referensi AWS CLI Perintah.

## **delete-fleet**

Contoh kode berikut menunjukkan cara menggunakan `delete-fleet`.

### AWS CLI

Untuk menghapus armada

Contoh ini menghapus armada.

Perintah:

```
aws robomaker delete-fleet --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771394395
```

- Untuk detail API, lihat [DeleteFleet](#) di Referensi AWS CLI Perintah.

## **delete-robot-application**

Contoh kode berikut menunjukkan cara menggunakan `delete-robot-application`.

### AWS CLI

Untuk menghapus aplikasi robot

Contoh ini menghapus aplikasi robot.

Perintah:

```
aws robomaker delete-robot-application --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821
```

- Untuk detail API, lihat [DeleteRobotApplication](#) di Referensi AWS CLI Perintah.

## **delete-robot**

Contoh kode berikut menunjukkan cara menggunakan `delete-robot`.

## AWS CLI

Untuk menghapus robot

Contoh ini menghapus robot.

Perintah:

```
aws robomaker delete-robot --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1540829698778
```

- Untuk detail API, lihat [DeleteRobot](#) di Referensi AWS CLI Perintah.

## **delete-simulation-application**

Contoh kode berikut menunjukkan cara menggunakan `delete-simulation-application`.

## AWS CLI

Untuk menghapus aplikasi simulasi

Contoh ini menghapus aplikasi simulasi.

Perintah:

```
aws robomaker delete-simulation-application --application arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605
```

- Untuk detail API, lihat [DeleteSimulationApplication](#) di Referensi AWS CLI Perintah.

## **deregister-robot**

Contoh kode berikut menunjukkan cara menggunakan `deregister-robot`.

## AWS CLI

Untuk membatalkan pendaftaran robot dari armada

Contoh ini membatalkan pendaftaran robot dari armada.

Perintah:

```
aws robomaker deregister-robot --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907 --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398
```

Output:

```
{
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907",
  "robot": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398"
}
```

- Untuk detail API, lihat [DeregisterRobot](#) di Referensi AWS CLI Perintah.

## describe-deployment-job

Contoh kode berikut menunjukkan cara menggunakan `describe-deployment-job`.

### AWS CLI

Untuk menggambarkan pekerjaan penerapan

`describe-deployment-job` Contoh berikut mengambil rincian tentang pekerjaan penerapan yang ditentukan.

```
aws robomaker describe-deployment-job \
  --job arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-x18qssl6pbcn
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-x18qssl6pbcn",
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/Trek/1539894765711",
  "status": "InProgress",
  "deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
  }
}
```

```

    },
    "deploymentApplicationConfigs": [
      {
        "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/RoboMakerHelloWorldRobot/1546541208251",
        "applicationVersion": "1",
        "launchConfig": {
          "packageName": "hello_world_robot",
          "launchFile": "rotate.launch"
        }
      }
    ],
    "createdAt": 1551218369.0,
    "robotDeploymentSummary": [
      {
        "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/
MyRobot/1540834232469",
        "deploymentStartTime": 1551218376.0,
        "status": "Deploying",
        "progressDetail": {}
      }
    ],
    "tags": {}
  }

```

- Untuk detail API, lihat [DescribeDeploymentJob](#) di Referensi AWS CLI Perintah.

## describe-fleet

Contoh kode berikut menunjukkan cara menggunakan `describe-fleet`.

### AWS CLI

Untuk menggambarkan armada

`describe-fleet` Contoh berikut mengambil rincian untuk armada yang ditentukan.

```

aws robomaker describe-fleet \
  --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyFleet/1550771358907

```

Output:

```
{
  "name": "MyFleet",
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
  "robots": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1540834232469",
      "createdAt": 1540834232.0
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyOtherRobot/1540829698778",
      "createdAt": 1540829698.0
    }
  ],
  "createdAt": 1539894765.0,
  "lastDeploymentStatus": "Succeeded",
  "lastDeploymentJob": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-xl8qssl6pbcn",
  "lastDeploymentTime": 1551218369.0,
  "tags": {}
}
```

- Untuk detail API, lihat [DescribeFleet](#) di Referensi AWS CLI Perintah.

## describe-robot-application

Contoh kode berikut menunjukkan cara menggunakan `describe-robot-application`.

### AWS CLI

Untuk menggambarkan aplikasi robot

Contoh ini menjelaskan aplikasi robot.

Perintah:

```
aws robomaker describe-robot-application --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/
MyRobotApplication/1551203485821",
  "name": "MyRobotApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-robot-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "revisionId": "e72efe0d-f44f-4333-b604-f6fa5c6bb50b",
  "lastUpdatedAt": 1551203485.0,
  "tags": {}
}
```

- Untuk detail API, lihat [DescribeRobotApplication](#) di Referensi AWS CLI Perintah.

## describe-robot

Contoh kode berikut menunjukkan cara menggunakan `describe-robot`.

### AWS CLI

Untuk menggambarkan robot

Contoh ini menggambarkan robot.

Perintah:

```
aws robomaker describe-robot --robot arn:aws:robomaker:us-west-2:111111111111:robot/
MyRobot/1550772324398
```

Output:

```
{
```

```
"arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398",
"name": "MyRobot",
"status": "Available",
"greengrassGroupId": "0f728a3c-7dbf-4a3e-976d-d16a8360caba",
"createdAt": 1550772325.0,
"architecture": "ARMHF",
"tags": {
  "Region": "East"
}
}
```

- Untuk detail API, lihat [DescribeRobot](#) di Referensi AWS CLI Perintah.

## describe-simulation-application

Contoh kode berikut menunjukkan cara menggunakan `describe-simulation-application`.

### AWS CLI

Untuk menggambarkan aplikasi simulasi

Contoh ini menjelaskan aplikasi simulasi.

Perintah:

```
aws robomaker describe-simulation-application --application arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605",
  "name": "MySimulationApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-simulation-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
}
```

```
"simulationSoftwareSuite": {
  "name": "Gazebo",
  "version": "7"
},
"robotSoftwareSuite": {
  "name": "ROS",
  "version": "Kinetic"
},
"renderingEngine": {
  "name": "OGRE",
  "version": "1.x"
},
"revisionId": "783674ab-b7b8-42d9-b01f-9373907987e5",
"lastUpdatedAt": 1551203427.0,
"tags": {}
}
```

- Untuk detail API, lihat [DescribeSimulationApplication](#) di Referensi AWS CLI Perintah.

## describe-simulation-job

Contoh kode berikut menunjukkan cara menggunakan `describe-simulation-job`.

### AWS CLI

Untuk menggambarkan pekerjaan simulasi

Contoh ini menjelaskan pekerjaan simulasi.

Perintah:

```
aws robomaker describe-simulation-job --job arn:aws:robomaker:us-
west-2:111111111111:simulation-job/sim-pql32v7pfjy6
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-pql32v7pfjy6",
  "status": "Running",
  "lastUpdatedAt": 1551219349.0,
  "failureBehavior": "Continue",
  "clientRequestToken": "a19ec4b5-e50d-3591-33da-c2e593c60615",
```



```

"outputLocation": {
  "s3Bucket": "my-output-bucket",
  "s3Prefix": "output"
},
"maxJobDurationInSeconds": 3600,
"simulationTimeMillis": 0,
"iamRole": "arn:aws:iam::111111111111:role/MySimulationRole",
"robotApplications": [
  {
    "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/MyRobotApplication/1551206341136",
    "applicationVersion": "$LATEST",
    "launchConfig": {
      "packageName": "hello_world_robot",
      "launchFile": "rotate.launch"
    }
  }
],
"simulationApplications": [
  {
    "application": "arn:aws:robomaker:us-west-2:111111111111:simulation-
application/MySimulationApplication/1551206347967",
    "applicationVersion": "$LATEST",
    "launchConfig": {
      "packageName": "hello_world_simulation",
      "launchFile": "empty_world.launch"
    }
  }
],
"tags": {}
}

```

- Untuk detail API, lihat [DescribeSimulationJob](#) di Referensi AWS CLI Perintah.

## list-deployment-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-deployment-jobs`.

### AWS CLI

Untuk membuat daftar pekerjaan penerapan

`list-deployment-jobs` Contoh berikut mengambil daftar pekerjaan deployment.

```
aws robomaker list-deployment-jobs
```

**Output:**

```
{
  "deploymentJobs": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/sim-6293szzm56rv",
      "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
      "status": "InProgress",
      "deploymentApplicationConfigs": [
        {
          "application": "arn:aws:robomaker:us-west-2:111111111111:robot-application/HelloWorldRobot/1546537110575",
          "applicationVersion": "1",
          "launchConfig": {
            "packageName": "hello_world_robot",
            "launchFile": "rotate.launch",
            "environmentVariables": {
              "ENVIRONMENT": "Desert"
            }
          }
        }
      ],
      "deploymentConfig": {
        "concurrentDeploymentPercentage": 20,
        "failureThresholdPercentage": 25
      },
      "createdAt": 1550689373.0
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-4w4g69p25zdb",
      "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
      "status": "Pending",
      "deploymentApplicationConfigs": [
        {
          "application": "arn:aws:robomaker:us-west-2:111111111111:robot-application/AWSRoboMakerHelloWorld-1544562726923_YGHM_sh5M/1544562822877",
          "applicationVersion": "1",

```

```
        "launchConfig": {
            "packageName": "fail",
            "launchFile": "fail"
        }
    ],
    "deploymentConfig": {
        "concurrentDeploymentPercentage": 20,
        "failureThresholdPercentage": 25
    },
    "failureReason": "",
    "failureCode": "",
    "createdAt": 1544719763.0
}
]
```

- Untuk detail API, lihat [ListDeploymentJobs](#) di Referensi AWS CLI Perintah.

## list-fleets

Contoh kode berikut menunjukkan cara menggunakan `list-fleets`.

### AWS CLI

Untuk membuat daftar armada

Contoh ini mencantumkan armada. Maksimal 20 armada akan dikembalikan.

Perintah:

```
aws robomaker list-fleets --max-items 20
```

Output:

```
{
  "fleetDetails": [
    {
      "name": "Trek",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
      "createdAt": 1539894765.0,

```

```
        "lastDeploymentStatus": "Failed",
        "lastDeploymentJob": "arn:aws:robomaker:us-west-2:111111111111:deployment-
job/deployment-4w4g69p25zdb",
        "lastDeploymentTime": 1544719763.0
    }
]
}
```

- Untuk detail API, lihat [ListFleets](#) di Referensi AWS CLI Perintah.

## list-robot-applications

Contoh kode berikut menunjukkan cara menggunakan `list-robot-applications`.

### AWS CLI

Untuk daftar aplikasi robot

Contoh ini mencantumkan aplikasi robot. Hasilnya terbatas pada 20 aplikasi robot.

Perintah:

```
aws robomaker list-robot-applications --max-results 20
```

Output:

```
{
  "robotApplicationSummaries": [
    {
      "name": "MyRobot",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/
MyRobot/1546537110575",
      "version": "$LATEST",
      "lastUpdatedAt": 1546540372.0
    },
    {
      "name": "AnotherRobot",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/
AnotherRobot/1546541208251",
      "version": "$LATEST",
      "lastUpdatedAt": 1546541208.0
    },
  ],
}
```

```
{
  "name": "MySuperRobot",
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/
MySuperRobot/1547663517377",
  "version": "$LATEST",
  "lastUpdatedAt": 1547663517.0
}
]
```

- Untuk detail API, lihat [ListRobotApplications](#) di Referensi AWS CLI Perintah.

## list-robots

Contoh kode berikut menunjukkan cara menggunakan `list-robots`.

### AWS CLI

Untuk membuat daftar robot

Contoh ini mencantumkan robot. Maksimal 20 robot akan dikembalikan.

Perintah:

```
aws robomaker list-robots --max-results 20
```

Output:

```
{
  "robots": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/
Robot100/1544035373264",
      "name": "Robot100",
      "status": "Available",
      "createdAt": 1544035373.0,
      "architecture": "X86_64"
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/
Robot101/1542146976587",
      "name": "Robot101",
```

```

    "status": "Available",
    "createdAt": 1542146976.0,
    "architecture": "X86_64"
  },
  {
    "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/
Robot102/1540834232469",
    "name": "Robot102",
    "fleetArn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
Trek/1539894765711",
    "status": "Available",
    "createdAt": 1540834232.0,
    "architecture": "X86_64",
    "lastDeploymentJob": "arn:aws:robomaker:us-west-2:111111111111:deployment-
job/deployment-jb007b75gl5f",
    "lastDeploymentTime": 1550689533.0
  },
  {
    "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/
MyRobot/1540829698778",
    "name": "MyRobot",
    "status": "Registered",
    "createdAt": 1540829698.0,
    "architecture": "X86_64"
  }
]
}

```

- Untuk detail API, lihat [ListRobots](#) di Referensi AWS CLI Perintah.

## list-simulation-applications

Contoh kode berikut menunjukkan cara menggunakan `list-simulation-applications`.

### AWS CLI

Untuk daftar aplikasi simulasi

Contoh ini mencantumkan aplikasi simulasi. Maksimal 20 aplikasi simulasi akan dikembalikan.

Perintah:

```
aws robomaker list-simulation-applications --max-results 20
```

## Output:

```
{
  "simulationApplicationSummaries": [
    {
      "name": "AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq/1548959170096",
      "version": "$LATEST",
      "lastUpdatedAt": 1548959170.0
    },
    {
      "name": "RoboMakerHelloWorldSimulation",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/RoboMakerHelloWorldSimulation/1546541198985",
      "version": "$LATEST",
      "lastUpdatedAt": 1546541198.0
    },
    {
      "name": "RoboMakerObjectTrackerSimulation",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/RoboMakerObjectTrackerSimulation/1545846795615",
      "version": "$LATEST",
      "lastUpdatedAt": 1545847405.0
    },
    {
      "name": "RoboMakerVoiceInteractionSimulation",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/RoboMakerVoiceInteractionSimulation/1546537100507",
      "version": "$LATEST",
      "lastUpdatedAt": 1546540352.0
    },
    {
      "name": "AWSRoboMakerCloudWatch-1547663411642_0LI6D1h6",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/AWSRoboMakerCloudWatch-1547663411642_0LI6D1h6/1547663521470",
      "version": "$LATEST",
      "lastUpdatedAt": 1547663521.0
    },
    {
      "name": "AWSRoboMakerDeepRacer-1545848257672_1YZCaieQ-",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/AWSRoboMakerDeepRacer-1545848257672_1YZCaieQ-/1545848370525",
      "version": "$LATEST",
```

```
    "lastUpdatedAt": 1545848370.0
  }
]
}
```

- Untuk detail API, lihat [ListSimulationApplications](#) di Referensi AWS CLI Perintah.

## list-simulation-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-simulation-jobs`.

### AWS CLI

Untuk daftar pekerjaan simulasi

Contoh ini mencantumkan pekerjaan simulasi.

Perintah:

```
aws robomaker list-simulation-jobs
```

Output:

```
{
  "simulationJobSummaries": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-66bbb3gpxm8x",
      "lastUpdatedAt": 1548959178.0,
      "status": "Completed",
      "simulationApplicationNames": [
        "AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq"
      ],
      "robotApplicationNames": [
        null
      ]
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-b27c4rkrtzcw",
      "lastUpdatedAt": 1543514088.0,
      "status": "Canceled",
      "simulationApplicationNames": [
```



```
        "AWSRoboMakerPersonDetection-1543513948280_T8rHW2_lu"  
    ],  
    "robotApplicationNames": [  
        "AWSRoboMakerPersonDetection-1543513948280_EYaMT0mYb"  
    ]  
},  
{  
    "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/  
sim-51vxjbyz4q8t",  
    "lastUpdatedAt": 1543508858.0,  
    "status": "Canceled",  
    "simulationApplicationNames": [  
        "AWSRoboMakerCloudWatch-1543504747391_lFF9ZQyx6"  
    ],  
    "robotApplicationNames": [  
        "AWSRoboMakerCloudWatch-1543504747391_axbYa3S3K"  
    ]  
},  
{  
    "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-  
kgf1fqxflqbx",  
    "lastUpdatedAt": 1543504862.0,  
    "status": "Completed",  
    "simulationApplicationNames": [  
        "AWSRoboMakerCloudWatch-1543504747391_lFF9ZQyx6"  
    ],  
    "robotApplicationNames": [  
        "AWSRoboMakerCloudWatch-1543504747391_axbYa3S3K"  
    ]  
},  
{  
    "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-  
vw8lvh061nqt",  
    "lastUpdatedAt": 1543441430.0,  
    "status": "Completed",  
    "simulationApplicationNames": [  
        "AWSRoboMakerHelloWorld-1543437372341__yb_Jg961"  
    ],  
    "robotApplicationNames": [  
        "AWSRoboMakerHelloWorld-1543437372341_lNbmKHvs9"  
    ]  
},  
{
```

```
    "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
txy5ypxmh84",
    "lastUpdatedAt": 1543437488.0,
    "status": "Completed",
    "simulationApplicationNames": [
      "AWSRoboMakerHelloWorld-1543437372341__yb_Jg961"
    ],
    "robotApplicationNames": [
      "AWSRoboMakerHelloWorld-1543437372341_lNbmKHvs9"
    ]
  }
]
}
```

- Untuk detail API, lihat [ListSimulationJobs](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag untuk sumber daya

Contoh ini mencantumkan tag untuk AWS RoboMaker sumber daya.

Perintah:

```
aws robomaker list-tags-for-resource --resource-arn "arn:aws:robomaker:us-
west-2:111111111111:robot/Robby_the_Robot/1544035373264"
```

Output:

```
{
  "tags": {
    "Region": "North",
    "Stage": "Initial"
  }
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## register-robot

Contoh kode berikut menunjukkan cara menggunakan `register-robot`.

### AWS CLI

Untuk mendaftarkan robot

Contoh ini mendaftarkan robot ke armada.

Perintah:

```
aws robomaker register-robot --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907 --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398
```

Output:

```
{
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907",
  "robot": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398"
}
```

- Untuk detail API, lihat [RegisterRobot](#) di Referensi AWS CLI Perintah.

## restart-simulation-job

Contoh kode berikut menunjukkan cara menggunakan `restart-simulation-job`.

### AWS CLI

Untuk memulai ulang simulasi

Contoh ini memulai ulang simulasi.

Perintah:

```
aws robomaker restart-simulation-job --job arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-t6rdgt70mftr
```

- Untuk detail API, lihat [RestartSimulationJob](#) di Referensi AWS CLI Perintah.

## sync-deployment-job

Contoh kode berikut menunjukkan cara menggunakan `sync-deployment-job`.

### AWS CLI

Untuk menyinkronkan pekerjaan penerapan

Contoh ini menyinkronkan pekerjaan penerapan.

Perintah:

```
aws robomaker sync-deployment-job --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/Trek/1539894765711
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/
deployment-09ccxs3tlfms",
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyFleet/1539894765711",
  "status": "Pending",
  "deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
  },
  "deploymentApplicationConfigs": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/MyRobotApplication/1546541208251",
      "applicationVersion": "1",
      "launchConfig": {
        "packageName": "hello_world_simulation",
        "launchFile": "empty_world.launch"
      }
    }
  ],
  "createdAt": 1551286954.0
}
```

- Untuk detail API, lihat [SyncDeploymentJob](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menandai sumber daya

Contoh ini menandai sumber daya. Ini melampirkan dua tag: Region dan Stage.

Perintah:

```
aws robomaker tag-resource --resource-arn "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1544035373264" --tags Region=North,Stage=Initial
```

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag sumber daya

Contoh ini menghapus tag dari sumber daya. Ini menghapus tag Wilayah.

Perintah:

```
aws robomaker untag-resource --resource-arn "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1544035373264" --tag-keys Region
```

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-robot-application

Contoh kode berikut menunjukkan cara menggunakan `update-robot-application`.

### AWS CLI

Untuk memperbarui aplikasi robot

Contoh ini memperbarui aplikasi robot.

Perintah:

```
aws robomaker update-robot-application --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821 --sources s3Bucket=my-bucket,s3Key=my-robot-application.tar.gz,architecture=X86_64 --robot-software-suite name=ROS,version=Kinetic
```

Output:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821",
  "name": "MyRobotApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-robot-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "lastUpdatedAt": 1551287993.0,
  "revisionId": "20b5e331-24fd-4504-8b8c-531afe5f4c94"
}
```

- Untuk detail API, lihat [UpdateRobotApplication](#) di Referensi AWS CLI Perintah.

## update-simulation-application

Contoh kode berikut menunjukkan cara menggunakan `update-simulation-application`.

AWS CLI

Untuk memperbarui aplikasi simulasi

Contoh ini memperbarui aplikasi simulasi.

**Perintah:**

```
aws robomaker update-simulation-application --application
arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MySimulationApplication/1551203427605 --sources s3Bucket=my-bucket,s3Key=my-
simulation-application.tar.gz,architecture=X86_64 --robot-software-suite
name=ROS,version=Kinetic --simulation-software-suite name=Gazebo,version=7 --
rendering-engine name=OGRE,version=1.x
```

**Output:**

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MySimulationApplication/1551203427605",
  "name": "MySimulationApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-simulation-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
  "simulationSoftwareSuite": {
    "name": "Gazebo",
    "version": "7"
  },
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "renderingEngine": {
    "name": "OGRE",
    "version": "1.x"
  },
  "lastUpdatedAt": 1551289361.0,
  "revisionId": "4a22cb5d-93c5-4cef-9311-52bdd119b79e"
}
```

- Untuk detail API, lihat [UpdateSimulationApplication](#) di Referensi AWS CLI Perintah.

## Rute 53 contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Route 53.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **change-resource-record-sets**

Contoh kode berikut menunjukkan cara menggunakan `change-resource-record-sets`.

AWS CLI

Untuk membuat, memperbarui, atau menghapus kumpulan catatan sumber daya

`change-resource-record-sets` Perintah berikut membuat catatan sumber daya yang ditetapkan menggunakan `hosted-zone-id` `Z1R8UBAEXAMPLE` dan konfigurasi berformat JSON dalam file: `C:\awscli\route53\change-resource-record-sets.json`

```
aws route53 change-resource-record-sets --hosted-zone-id Z1R8UBAEXAMPLE --change-batch file://C:\awscli\route53\change-resource-record-sets.json
```

Untuk informasi selengkapnya, lihat `POST ChangeResourceRecordSets` di Referensi API Amazon Route 53.

Konfigurasi dalam file JSON tergantung pada jenis kumpulan catatan sumber daya yang ingin Anda buat:



## BasicWeightedAliasWeighted AliasLatencyLatency AliasFailoverFailover Alias

### Sintaks dasar:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ]
      }
    },
    {...}
  ]
}
```

### Sintaks tertimbang:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Weight": value between 0 and 255,
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ]
      }
    }
  ]
}
```

```

    ],
    "HealthCheckId": "optional ID of an Amazon Route 53 health check"
  }
},
{...}
]
}

```

### Sintaks Alias:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
          S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
          bucket, Elastic Load Balancing load balancer, or another resource record set in
          this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}

```

### Sintaks Alias Tertimbang:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",

```

```

    "SetIdentifier": "unique description for this resource record set",
    "Weight": value between 0 and 255,
    "AliasTarget": {
        "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
        "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
bucket, Elastic Load Balancing load balancer, or another resource record set in
this hosted zone",
        "EvaluateTargetHealth": true|false
    },
    "HealthCheckId": "optional ID of an Amazon Route 53 health check"
  }
},
{...}
]
}

```

### Sintaks Latensi:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Region": "Amazon EC2 region name",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}

```

## Sintaks Alias Latensi:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Region": "Amazon EC2 region name",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
          S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
          bucket, Elastic Load Balancing load balancer, or another resource record set in
          this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

## Sintaks Failover:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Failover": "PRIMARY" | "SECONDARY",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          }
        ]
      }
    }
  ]
}
```

```

    },
    {...}
  ],
  "HealthCheckId": "ID of an Amazon Route 53 health check"
}
},
{...}
]
}

```

### Sintaks Alias Failover:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Failover": "PRIMARY" | "SECONDARY",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
bucket, Elastic Load Balancing load balancer, or another resource record set in
this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}

```

- Untuk detail API, lihat [ChangeResourceRecordSets](#) di Referensi AWS CLI Perintah.

### change-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `change-tags-for-resource`.

## AWS CLI

Perintah berikut menambahkan tag bernama `owner` ke sumber daya pemeriksaan kesehatan yang ditentukan oleh ID:

```
aws route53 change-tags-for-resource --resource-type healthcheck --resource-id 6233434j-18c1-34433-ba8e-3443434 --add-tags Key=owner,Value=myboss
```

Perintah berikut menghapus tag bernama `owner` dari sumber daya zona host yang ditentukan oleh ID:

```
aws route53 change-tags-for-resource --resource-type hostedzone --resource-id Z1523434445 --remove-tag-keys owner
```

- Untuk detail API, lihat [ChangeTagsForResource](#) di Referensi AWS CLI Perintah.

## create-health-check

Contoh kode berikut menunjukkan cara menggunakan `create-health-check`.

### AWS CLI

Untuk membuat pemeriksaan kesehatan

`create-health-check` Perintah berikut membuat pemeriksaan kesehatan menggunakan referensi pemanggil `2014-04-01-18:47` dan konfigurasi berformat JSON dalam file: `C:\awscli\route53\create-health-check.json`

```
aws route53 create-health-check --caller-reference 2014-04-01-18:47 --health-check-config file://C:\awscli\route53\create-health-check.json
```

Sintaks JSON:

```
{
  "IPAddress": "IP address of the endpoint to check",
  "Port": port on the endpoint to check--required when Type is "TCP",
  "Type": "HTTP"|"HTTPS"|"HTTP_STR_MATCH"|"HTTPS_STR_MATCH"|"TCP",
  "ResourcePath": "path of the file that you want Amazon Route 53 to request--all Types except TCP",
  "FullyQualifiedDomainName": "domain name of the endpoint to check--all Types except TCP",
```

```
"SearchString": "if Type is HTTP_STR_MATCH or HTTPS_STR_MATCH, the string to
search for in the response body from the specified resource",
"RequestInterval": 10 | 30,
"FailureThreshold": integer between 1 and 10
}
```

Untuk menambahkan pemeriksaan kesehatan ke kumpulan catatan sumber daya Route 53, gunakan `change-resource-record-sets` perintah.

Untuk informasi selengkapnya, lihat Amazon Route 53 Health Checks dan DNS Failover di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CreateHealthCheck](#) di Referensi AWS CLI Perintah.

## **create-hosted-zone**

Contoh kode berikut menunjukkan cara menggunakan `create-hosted-zone`.

### AWS CLI

Untuk membuat zona yang dihosting

`create-hosted-zone` Perintah berikut menambahkan zona host bernama `example.com` menggunakan referensi `2014-04-01-18:47` pemanggil. Komentar opsional mencakup spasi, sehingga harus dilampirkan dalam tanda kutip:

```
aws route53 create-hosted-zone --name example.com --caller-reference
2014-04-01-18:47 --hosted-zone-config Comment="command-line version"
```

Untuk informasi selengkapnya, lihat Bekerja dengan Zona yang Dihosting di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CreateHostedZone](#) di Referensi AWS CLI Perintah.

## **delete-health-check**

Contoh kode berikut menunjukkan cara menggunakan `delete-health-check`.

### AWS CLI

Untuk menghapus pemeriksaan kesehatan

`delete-health-check` Perintah berikut menghapus pemeriksaan kesehatan dengan `health-check-id` dari `e75b48d9-547a-4c3d-88a5-ae4002397608`:

```
aws route53 delete-health-check --health-check-id e75b48d9-547a-4c3d-88a5-ae4002397608
```

- Untuk detail API, lihat [DeleteHealthCheck](#) di Referensi AWS CLI Perintah.

## **delete-hosted-zone**

Contoh kode berikut menunjukkan cara menggunakan `delete-hosted-zone`.

### AWS CLI

Untuk menghapus zona yang dihosting

`delete-hosted-zone` Perintah berikut menghapus zona yang dihosting dengan `id` dari `Z36KTIQEXAMPLE`:

```
aws route53 delete-hosted-zone --id Z36KTIQEXAMPLE
```

- Untuk detail API, lihat [DeleteHostedZone](#) di Referensi AWS CLI Perintah.

## **get-change**

Contoh kode berikut menunjukkan cara menggunakan `get-change`.

### AWS CLI

Untuk mendapatkan status perubahan pada kumpulan rekaman sumber daya

`get-change` Perintah berikut mendapatkan status dan informasi lain tentang `change-resource-record-sets` permintaan yang memiliki `Id` dari `/change/CWPIK4URU2I5S`:

```
aws route53 get-change --id /change/CWPIK4URU2I5S
```

- Untuk detail API, lihat [GetChange](#) di Referensi AWS CLI Perintah.

## **get-health-check**

Contoh kode berikut menunjukkan cara menggunakan `get-health-check`.



## AWS CLI

Untuk mendapatkan informasi tentang pemeriksaan kesehatan

`get-health-check` Perintah berikut mendapatkan informasi tentang pemeriksaan kesehatan `health-check-id` yang memiliki `02ec8401-9879-4259-91fa-04e66d094674`:

```
aws route53 get-health-check --health-check-id 02ec8401-9879-4259-91fa-04e66d094674
```

- Untuk detail API, lihat [GetHealthCheck](#) di Referensi AWS CLI Perintah.

## `get-hosted-zone`

Contoh kode berikut menunjukkan cara menggunakan `get-hosted-zone`.

### AWS CLI

Untuk mendapatkan informasi tentang zona yang dihosting

`get-hosted-zone` Perintah berikut mendapatkan informasi tentang zona yang dihosting dengan `id` dari `Z1R8UBAEXAMPLE`:

```
aws route53 get-hosted-zone --id Z1R8UBAEXAMPLE
```

- Untuk detail API, lihat [GetHostedZone](#) di Referensi AWS CLI Perintah.

## `list-health-checks`

Contoh kode berikut menunjukkan cara menggunakan `list-health-checks`.

### AWS CLI

Untuk membuat daftar pemeriksaan kesehatan yang terkait dengan AWS akun saat ini

`list-health-checks` Perintah berikut mencantumkan informasi terperinci tentang 100 pemeriksaan kesehatan pertama yang terkait dengan AWS akun saat ini. :

```
aws route53 list-health-checks
```

Jika Anda memiliki lebih dari 100 pemeriksaan kesehatan, atau jika Anda ingin mencantumkannya dalam kelompok yang lebih kecil dari 100, sertakan `--max-items` parameternya. Misalnya, untuk membuat daftar pemeriksaan kesehatan satu per satu, gunakan perintah berikut:

```
aws route53 list-health-checks --max-items 1
```

Untuk melihat pemeriksaan kesehatan berikutnya, ambil nilai `NextToken` dari respons ke perintah sebelumnya, dan sertakan dalam `--starting-token` parameter, misalnya:

```
aws route53 list-health-checks --max-items 1 --starting-token Z3M3LMPEXAMPLE
```

- Untuk detail API, lihat [ListHealthChecks](#) di Referensi AWS CLI Perintah.

## list-hosted-zones-by-name

Contoh kode berikut menunjukkan cara menggunakan `list-hosted-zones-by-name`.

### AWS CLI

Perintah berikut mencantumkan hingga 100 zona yang dihosting yang diurutkan berdasarkan nama domain:

```
aws route53 list-hosted-zones-by-name
```

Output:

```
{
  "HostedZones": [
    {
      "ResourceRecordSetCount": 2,
      "CallerReference": "test20150527-2",
      "Config": {
        "Comment": "test2",
        "PrivateZone": false
      },
      "Id": "/hostedzone/Z119WBBTVP5WFX",
      "Name": "2.example.com."
    },
    {
      "ResourceRecordSetCount": 2,
      "CallerReference": "test20150527-1",
```

```
    "Config": {
      "Comment": "test",
      "PrivateZone": false
    },
    "Id": "/hostedzone/Z3P5QSUBK4P0TI",
    "Name": "www.example.com."
  }
],
"IsTruncated": false,
"MaxItems": "100"
}
```

Perintah berikut mencantumkan zona yang dihosting yang diurutkan berdasarkan nama, dimulai dengan `www.example.com`:

```
aws route53 list-hosted-zones-by-name --dns-name www.example.com
```

Output:

```
{
  "HostedZones": [
    {
      "ResourceRecordSetCount": 2,
      "CallerReference": "mwunderl20150527-1",
      "Config": {
        "Comment": "test",
        "PrivateZone": false
      },
      "Id": "/hostedzone/Z3P5QSUBK4P0TI",
      "Name": "www.example.com."
    }
  ],
  "DNSName": "www.example.com",
  "IsTruncated": false,
  "MaxItems": "100"
}
```

- Untuk detail API, lihat [ListHostedZonesByName](#) di Referensi AWS CLI Perintah.

## list-hosted-zones

Contoh kode berikut menunjukkan cara menggunakan `list-hosted-zones`.

## AWS CLI

Untuk membuat daftar zona yang dihosting yang terkait dengan AWS akun saat ini

`list-hosted-zones` Perintah berikut mencantumkan informasi ringkasan tentang 100 zona host pertama yang terkait dengan AWS akun saat ini. :

```
aws route53 list-hosted-zones
```

Jika Anda memiliki lebih dari 100 zona yang dihosting, atau jika Anda ingin mencantumkannya dalam grup yang lebih kecil dari 100, sertakan `--max-items` parameternya. Misalnya, untuk membuat daftar zona yang dihosting satu per satu, gunakan perintah berikut:

```
aws route53 list-hosted-zones --max-items 1
```

Untuk melihat informasi tentang zona host berikutnya, ambil nilai `NextToken` dari respon ke perintah sebelumnya, dan sertakan dalam `--starting-token` parameter, misalnya:

```
aws route53 list-hosted-zones --max-items 1 --starting-token Z3M3LMPEXAMPLE
```

- Untuk detail API, lihat [ListHostedZones](#) di Referensi AWS CLI Perintah.

## `list-query-logging-configs`

Contoh kode berikut menunjukkan cara menggunakan `list-query-logging-configs`.

### AWS CLI

Untuk membuat daftar konfigurasi pencatatan kueri

`list-query-logging-configs` Contoh berikut mencantumkan informasi tentang 100 konfigurasi pencatatan kueri pertama di AWS akun Anda, untuk zona `Z10X3WQEXAMPLE` yang dihosting.

```
aws route53 list-query-logging-configs \  
  --hosted-zone-id Z10X3WQEXAMPLE
```

Output:

```
{
```

```
"QueryLoggingConfigs": [  
  {  
    "Id": "964ff34e-ae03-4f06-80a2-9683cexample",  
    "HostedZoneId": "Z10X3WQEXAMPLE",  
    "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-east-1:111122223333:log-  
group:/aws/route53/example.com:*"  
  }  
]
```

Untuk informasi selengkapnya, lihat [Mencatat kueri DNS di Panduan](#) Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListQueryLoggingConfigs](#) di Referensi AWS CLI Perintah.

## list-resource-record-sets

Contoh kode berikut menunjukkan cara menggunakan `list-resource-record-sets`.

### AWS CLI

Untuk mencantumkan kumpulan rekaman sumber daya di zona yang dihosting

`list-resource-record-sets` Perintah berikut mencantumkan informasi ringkasan tentang 100 set catatan sumber daya pertama di zona host tertentu. :

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE
```

Jika zona yang dihosting berisi lebih dari 100 kumpulan catatan sumber daya, atau jika Anda ingin mencantumkannya dalam grup yang lebih kecil dari 100, sertakan `--max-items` parameternya. Misalnya, untuk mencantumkan set catatan sumber daya satu per satu, gunakan perintah berikut:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --max-items 1
```

Untuk melihat informasi tentang catatan sumber daya berikutnya yang ditetapkan di zona yang dihosting, ambil nilai `NextToken` dari respons ke perintah sebelumnya, dan sertakan dalam `--starting-token` parameter, misalnya:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --max-items 1  
--starting-token Z3M3LMPEXAMPLE
```

Untuk melihat semua kumpulan catatan sumber daya dari nama tertentu, gunakan `--query` parameter untuk memfilternya. Sebagai contoh:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --query  
"ResourceRecordSets[?Name == 'example.domain.']"
```

- Untuk detail API, lihat [ListResourceRecordSets](#) di Referensi AWS CLI Perintah.

## Route 53 contoh pendaftaran domain menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan pendaftaran domain AWS Command Line Interface with Route 53.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **check-domain-availability**

Contoh kode berikut menunjukkan cara menggunakan `check-domain-availability`.

AWS CLI

Untuk menentukan apakah Anda dapat mendaftarkan nama domain dengan Route 53

`check-domain-availability` Perintah berikut mengembalikan informasi tentang apakah nama `example.com` domain tersedia untuk didaftarkan menggunakan Route 53.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains check-domain-availability \  
  --region us-east-1 \  
  --domain-name example.com
```

#### Output:

```
{  
  "Availability": "UNAVAILABLE"  
}
```

Route 53 mendukung sejumlah besar domain tingkat atas (TLD), seperti .com dan .jp, tetapi kami tidak mendukung semua TLD yang tersedia. Jika Anda memeriksa ketersediaan domain dan Route 53 tidak mendukung TLD, akan `check-domain-availability` menampilkan pesan berikut.

```
An error occurred (UnsupportedTLD) when calling the CheckDomainAvailability  
operation: <top-level domain> tld is not supported.
```

Untuk daftar TLD yang dapat Anda gunakan saat mendaftarkan domain dengan Route 53, lihat [Domain yang Dapat Anda Daftarkan dengan Amazon Route 53](#) di Panduan Pengembang Amazon Route 53. Untuk informasi selengkapnya tentang mendaftarkan domain dengan Amazon Route 53, lihat [Mendaftarkan Domain Baru](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CheckDomainAvailability](#) di Referensi AWS CLI Perintah.

## check-domain-transferability

Contoh kode berikut menunjukkan cara menggunakan `check-domain-transferability`.

### AWS CLI

Untuk menentukan apakah domain dapat ditransfer ke Route 53

`check-domain-transferability` Perintah berikut mengembalikan informasi tentang apakah Anda dapat mentransfer nama domain `example.com` ke Route 53.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains check-domain-transferability \  
  --region us-east-1
```

```
--region us-east-1 \  
--domain-name example.com
```

Output:

```
{  
  "Transferability": {  
    "Transferable": "UNTRANSFERABLE"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mentransfer Pendaftaran Domain ke Amazon Route 53](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CheckDomainTransferability](#) di Referensi AWS CLI Perintah.

## delete-tags-for-domain

Contoh kode berikut menunjukkan cara menggunakan `delete-tags-for-domain`.

### AWS CLI

Untuk menghapus tag untuk domain

`delete-tags-for-domain` Perintah berikut menghapus tiga tag dari domain yang ditentukan. Perhatikan bahwa Anda hanya menentukan kunci tag, bukan nilai tag.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains delete-tags-for-domain \  
--region us-east-1 \  
--domain-name example.com \  
--tags-to-delete accounting-key hr-key engineering-key
```

Perintah ini tidak menghasilkan output.

Untuk mengonfirmasi bahwa tag telah dihapus, Anda dapat menjalankannya [list-tags-for-domain](#). Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Amazon Route 53](#) di Panduan Pengembang Amazon Route 53.



- Untuk detail API, lihat [DeleteTagsForDomain](#) di Referensi AWS CLI Perintah.

## **disable-domain-auto-renew**

Contoh kode berikut menunjukkan cara menggunakan `disable-domain-auto-renew`.

### AWS CLI

Untuk menonaktifkan perpanjangan otomatis domain

`disable-domain-auto-renew` Perintah berikut mengonfigurasi Route 53 agar tidak memperbarui domain secara otomatis `example.com` sebelum pendaftaran domain kedaluwarsa.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains disable-domain-auto-renew \  
  --region us-east-1 \  
  --domain-name example.com
```

Perintah ini tidak menghasilkan output.

Untuk mengonfirmasi bahwa pengaturan telah diubah, Anda dapat menjalankannya [get-domain-detail](#). Jika perpanjangan otomatis dinonaktifkan, nilainya `AutoRenew` adalah `False`. Untuk informasi selengkapnya tentang perpanjangan otomatis, lihat [Memperbarui Pendaftaran untuk Domain < https://docs.aws.amazon.com/Route53/latest/ DeveloperGuide /domain-renew.html](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-renew.html) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [DisableDomainAutoRenew](#) di Referensi AWS CLI Perintah.

## **disable-domain-transfer-lock**

Contoh kode berikut menunjukkan cara menggunakan `disable-domain-transfer-lock`.

### AWS CLI

Untuk menonaktifkan kunci transfer pada domain

`disable-domain-transfer-lock` Perintah berikut menghapus kunci transfer pada domain `example.com` sehingga domain dapat ditransfer ke registrar lain. Perintah ini mengubah `clientTransferProhibited` status.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains disable-domain-transfer-lock \  
  --region us-east-1 \  
  --domain-name example.com
```

Output:

```
{  
  "OperationId": "3f28e0ac-126a-4113-9048-cc930example"  
}
```

Untuk mengonfirmasi bahwa kunci transfer telah diubah, Anda dapat menjalankannya [get-domain-detail](#). Ketika kunci transfer dinonaktifkan, nilai `StatusList` tidak termasuk `clientTransferProhibited`.

Untuk informasi selengkapnya tentang proses transfer, lihat [Mentransfer Domain dari Amazon Route 53 ke Registrar Lain di Panduan](#) Pengembang Amazon Route 53.

- Untuk detail API, lihat [DisableDomainTransferLock](#) di Referensi AWS CLI Perintah.

## **enable-domain-auto-renew**

Contoh kode berikut menunjukkan cara menggunakan `enable-domain-auto-renew`.

### AWS CLI

Untuk mengaktifkan perpanjangan otomatis domain

`enable-domain-auto-renew` Perintah berikut mengonfigurasi Route 53 untuk memperbarui domain secara otomatis `example.com` sebelum pendaftaran domain kedaluwarsa.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains enable-domain-auto-renew \  
  --region us-east-1 \  
  --domain-name example.com
```

Perintah ini tidak menghasilkan output. Untuk mengonfirmasi bahwa pengaturan telah diubah, Anda dapat menjalankannya [get-domain-detail](#). Jika perpanjangan otomatis diaktifkan, nilainya `AutoRenew` adalah `True`.

Untuk informasi selengkapnya tentang perpanjangan otomatis, lihat [Memperbarui Pendaftaran untuk Domain < https://docs.aws.amazon.com/Route53/latest/ DeveloperGuide /domain-renew.html](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-renew.html) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [EnableDomainAutoRenew](#) di Referensi AWS CLI Perintah.

## **enable-domain-transfer-lock**

Contoh kode berikut menunjukkan cara menggunakan `enable-domain-transfer-lock`.

### AWS CLI

Untuk mengaktifkan kunci transfer pada domain

`enable-domain-transfer-lock` Perintah berikut mengunci domain tertentu sehingga tidak dapat ditransfer ke registrar lain. Perintah ini mengubah `clientTransferProhibited` status.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains enable-domain-transfer-lock \  
  --region us-east-1 \  
  --domain-name example.com
```

Output:

```
{  
  "OperationId": "3f28e0ac-126a-4113-9048-cc930example"  
}
```

Untuk mengonfirmasi bahwa kunci transfer telah diubah, Anda dapat menjalankannya [get-domain-detail](#). Ketika kunci transfer diaktifkan, nilai `StatusList` termasuk `clientTransferProhibited`.

Untuk informasi selengkapnya tentang proses transfer, lihat [Mentransfer Domain dari Amazon Route 53 ke Registrar Lain di Panduan](#) Pengembang Amazon Route 53.

- Untuk detail API, lihat [EnableDomainTransferLock](#) di Referensi AWS CLI Perintah.

## get-contact-reachability-status

Contoh kode berikut menunjukkan cara menggunakan `get-contact-reachability-status`.

### AWS CLI

Untuk menentukan apakah kontak pendaftar telah menanggapi email konfirmasi

`get-contact-reachability-status` Perintah berikut mengembalikan informasi tentang apakah kontak pendaftar untuk domain tertentu telah menanggapi email konfirmasi.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains get-contact-reachability-status \
  --region us-east-1 \
  --domain-name example.com
```

Output:

```
{
  "domainName": "example.com",
  "status": "DONE"
}
```

Untuk informasi selengkapnya, lihat [Mengirim Ulang Email Otorisasi dan Konfirmasi](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetContactReachabilityStatus](#) di Referensi AWS CLI Perintah.

## get-domain-detail

Contoh kode berikut menunjukkan cara menggunakan `get-domain-detail`.

### AWS CLI

Untuk mendapatkan informasi rinci tentang domain tertentu

`get-domain-detail` Perintah berikut menampilkan informasi rinci tentang domain yang ditentukan.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains get-domain-detail \  
--region us-east-1 \  
--domain-name example.com
```

**Output:**

```
{  
  "DomainName": "example.com",  
  "Nameservers": [  
    {  
      "Name": "ns-2048.awsdns-64.com",  
      "GlueIps": []  
    },  
    {  
      "Name": "ns-2049.awsdns-65.net",  
      "GlueIps": []  
    },  
    {  
      "Name": "ns-2050.awsdns-66.org",  
      "GlueIps": []  
    },  
    {  
      "Name": "ns-2051.awsdns-67.co.uk",  
      "GlueIps": []  
    }  
  ],  
  "AutoRenew": true,  
  "AdminContact": {  
    "FirstName": "Saanvi",  
    "LastName": "Sarkar",  
    "ContactType": "COMPANY",  
    "OrganizationName": "Example",  
    "AddressLine1": "123 Main Street",  
    "City": "Anytown",  
    "State": "WA",  
    "CountryCode": "US",  
    "ZipCode": "98101",  
    "PhoneNumber": "+1.8005551212",  
    "Email": "ssarkar@example.com",  
    "ExtraParams": []  
  },  
  "RegistrantContact": {  
    "FirstName": "Alejandro",
```

```
    "LastName": "Rosalez",
    "ContactType": "COMPANY",
    "OrganizationName": "Example",
    "AddressLine1": "123 Main Street",
    "City": "Anytown",
    "State": "WA",
    "CountryCode": "US",
    "ZipCode": "98101",
    "PhoneNumber": "+1.8005551212",
    "Email": "arosalez@example.com",
    "ExtraParams": []
  },
  "TechContact": {
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "ContactType": "COMPANY",
    "OrganizationName": "Example",
    "AddressLine1": "123 Main Street",
    "City": "Anytown",
    "State": "WA",
    "CountryCode": "US",
    "ZipCode": "98101",
    "PhoneNumber": "+1.8005551212",
    "Email": "wxiulan@example.com",
    "ExtraParams": []
  },
  "AdminPrivacy": true,
  "RegistrantPrivacy": true,
  "TechPrivacy": true,
  "RegistrarName": "Amazon Registrar, Inc.",
  "WhoIsServer": "whois.registrar.amazon.com",
  "RegistrarUrl": "http://registrar.amazon.com",
  "AbuseContactEmail": "abuse@registrar.amazon.com",
  "AbuseContactPhone": "+1.2062661000",
  "CreationDate": 1444934889.601,
  "ExpirationDate": 1602787689.0,
  "StatusList": [
    "clientTransferProhibited"
  ]
}
```

- Untuk detail API, lihat [GetDomainDetail](#) di Referensi AWS CLI Perintah.

## get-domain-suggestions

Contoh kode berikut menunjukkan cara menggunakan `get-domain-suggestions`.

### AWS CLI

Untuk mendapatkan daftar nama domain yang disarankan

`get-domain-suggestions` Perintah berikut menampilkan daftar nama domain yang disarankan berdasarkan nama domain `example.com`. Responsnya hanya mencakup nama domain yang tersedia. Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains get-domain-suggestions \
  --region us-east-1 \
  --domain-name example.com \
  --suggestion-count 10 \
  --only-available
```

### Output:

```
{
  "SuggestionsList": [
    {
      "DomainName": "egzaampal.com",
      "Availability": "AVAILABLE"
    },
    {
      "DomainName": "examplelaw.com",
      "Availability": "AVAILABLE"
    },
    {
      "DomainName": "examplehouse.net",
      "Availability": "AVAILABLE"
    },
    {
      "DomainName": "homeexample.net",
      "Availability": "AVAILABLE"
    },
    {
      "DomainName": "examplelist.com",
      "Availability": "AVAILABLE"
    },
  ],
}
```

```
{
  "DomainName": "exemplenews.net",
  "Availability": "AVAILABLE"
},
{
  "DomainName": "officeexample.com",
  "Availability": "AVAILABLE"
},
{
  "DomainName": "exampleworld.com",
  "Availability": "AVAILABLE"
},
{
  "DomainName": "exampleart.com",
  "Availability": "AVAILABLE"
}
]
```

- Untuk detail API, lihat [GetDomainSuggestions](#) di Referensi AWS CLI Perintah.

## get-operation-detail

Contoh kode berikut menunjukkan cara menggunakan `get-operation-detail`.

### AWS CLI

Untuk mendapatkan status operasi saat ini

Beberapa operasi pendaftaran domain beroperasi secara asinkron dan mengembalikan respons sebelum selesai. Operasi ini mengembalikan ID operasi yang dapat Anda gunakan untuk mendapatkan status saat ini. `get-operation-detail` Perintah berikut mengembalikan status operasi yang ditentukan.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains get-operation-detail \
  --region us-east-1 \
  --operation-id edbd8d63-7fe7-4343-9bc5-54033example
```

Output:



```
{
  "OperationId": "edbd8d63-7fe7-4343-9bc5-54033example",
  "Status": "SUCCESSFUL",
  "DomainName": "example.com",
  "Type": "DOMAIN_LOCK",
  "SubmittedDate": 1573749367.864
}
```

- Untuk detail API, lihat [GetOperationDetail](#) di Referensi AWS CLI Perintah.

## list-domains

Contoh kode berikut menunjukkan cara menggunakan `list-domains`.

### AWS CLI

Untuk membuat daftar domain yang terdaftar dengan akun saat ini AWS

`list-domains` Perintah berikut mencantumkan informasi ringkasan tentang domain yang terdaftar dengan AWS akun saat ini.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains list-domains
  --region us-east-1
```

Output:

```
{
  "Domains": [
    {
      "DomainName": "example.com",
      "AutoRenew": true,
      "TransferLock": true,
      "Expiry": 1602712345.0
    },
    {
      "DomainName": "example.net",
      "AutoRenew": true,
      "TransferLock": true,
      "Expiry": 1602723456.0
    }
  ]
}
```

```
    },
    {
      "DomainName": "example.org",
      "AutoRenew": true,
      "TransferLock": true,
      "Expiry": 1602734567.0
    }
  ]
}
```

- Untuk detail API, lihat [ListDomains](#) di Referensi AWS CLI Perintah.

## list-operations

Contoh kode berikut menunjukkan cara menggunakan `list-operations`.

### AWS CLI

Untuk mencantumkan status operasi yang mengembalikan ID operasi

Beberapa operasi pendaftaran domain berjalan secara asinkron dan mengembalikan respons sebelum selesai. Operasi ini mengembalikan ID operasi yang dapat Anda gunakan untuk mendapatkan status saat ini. `list-operations` Perintah berikut mencantumkan informasi ringkasan, termasuk status, tentang operasi registrasi domain saat ini.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains list-operations
  --region us-east-1
```

Output:

```
{
  "Operations": [
    {
      "OperationId": "aab9822f-1da0-4bf3-8a15-fd4e0example",
      "Status": "SUCCESSFUL",
      "Type": "DOMAIN_LOCK",
      "SubmittedDate": 1455321739.986
    },
    {
```

```

    "OperationId": "c24379ed-76be-42f8-bdad-9379bexample",
    "Status": "SUCCESSFUL",
    "Type": "UPDATE_NAMESERVER",
    "SubmittedDate": 1468960475.109
  },
  {
    "OperationId": "f47e1297-ef9e-4c2b-ae1e-a5fcbexample",
    "Status": "SUCCESSFUL",
    "Type": "RENEW_DOMAIN",
    "SubmittedDate": 1473561835.943
  },
  {
    "OperationId": "75584f23-b15f-459e-aed7-dc6f5example",
    "Status": "SUCCESSFUL",
    "Type": "UPDATE_DOMAIN_CONTACT",
    "SubmittedDate": 1547501003.41
  }
]
}

```

Outputnya mencakup semua operasi yang mengembalikan ID operasi dan yang telah Anda lakukan di semua domain yang pernah Anda daftarkan menggunakan AWS akun saat ini. Jika Anda hanya ingin mendapatkan operasi yang Anda kirimkan setelah tanggal yang ditentukan, Anda dapat menyertakan `submitted-since` parameter dan menentukan tanggal dalam format Unix dan Waktu Universal Terkoordinasi (UTC). Perintah berikut mendapatkan status semua operasi yang diajukan setelah pukul 12:00 UTC pada 1 Januari 2020.

```
aws route53domains list-operations \
  --submitted-since 1577836800
```

- Untuk detail API, lihat [ListOperations](#) di Referensi AWS CLI Perintah.

## list-tags-for-domain

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-domain`.

### AWS CLI

Untuk mencantumkan tag untuk domain

`list-tags-for-domain` Perintah berikut mencantumkan tag yang saat ini terkait dengan domain yang ditentukan.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains list-tags-for-domain \  
  --region us-east-1 \  
  --domain-name example.com
```

Output:

```
{  
  "TagList": [  
    {  
      "Key": "key1",  
      "Value": "value1"  
    },  
    {  
      "Key": "key2",  
      "Value": "value2"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Amazon Route 53](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListTagsForDomain](#) di Referensi AWS CLI Perintah.

## register-domain

Contoh kode berikut menunjukkan cara menggunakan `register-domain`.

### AWS CLI

Untuk mendaftarkan domain

`register-domain` Perintah berikut mendaftarkan domain, mengambil semua nilai parameter dari file berformat JSON.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains register-domain \  
  --domain-name example.com
```

```
--region us-east-1 \  
--cli-input-json file://register-domain.json
```

Isi dari `register-domain.json`:

```
{  
  "DomainName": "example.com",  
  "DurationInYears": 1,  
  "AutoRenew": true,  
  "AdminContact": {  
    "FirstName": "Martha",  
    "LastName": "Rivera",  
    "ContactType": "PERSON",  
    "OrganizationName": "Example",  
    "AddressLine1": "1 Main Street",  
    "City": "Anytown",  
    "State": "WA",  
    "CountryCode": "US",  
    "ZipCode": "98101",  
    "PhoneNumber": "+1.8005551212",  
    "Email": "mrivera@example.com"  
  },  
  "RegistrantContact": {  
    "FirstName": "Li",  
    "LastName": "Juan",  
    "ContactType": "PERSON",  
    "OrganizationName": "Example",  
    "AddressLine1": "1 Main Street",  
    "City": "Anytown",  
    "State": "WA",  
    "CountryCode": "US",  
    "ZipCode": "98101",  
    "PhoneNumber": "+1.8005551212",  
    "Email": "ljuan@example.com"  
  },  
  "TechContact": {  
    "FirstName": "Mateo",  
    "LastName": "Jackson",  
    "ContactType": "PERSON",  
    "OrganizationName": "Example",  
    "AddressLine1": "1 Main Street",  
    "City": "Anytown",  
    "State": "WA",
```

```
"CountryCode": "US",
"ZipCode": "98101",
"PhoneNumber": "+1.8005551212",
"Email": "mjackson@example.com"
},
"PrivacyProtectAdminContact": true,
"PrivacyProtectRegistrantContact": true,
"PrivacyProtectTechContact": true
}
```

### Output:

```
{
  "OperationId": "b114c44a-9330-47d1-a6e8-a0b11example"
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation-detail`. Untuk informasi lebih lanjut, lihat [get-operation-detail](#).

Untuk informasi lebih lanjut, lihat [Mendaftarkan Domain Baru](#) dalam Panduan Pengembang Amazon Route 53.

Untuk informasi tentang domain tingkat atas (TLD) mana yang memerlukan nilai `ExtraParams` dan nilai validnya, lihat [ExtraParam](#) di Referensi API Amazon Route 53.

- Untuk detail API, lihat [RegisterDomain](#) di Referensi AWS CLI Perintah.

## renew-domain

Contoh kode berikut menunjukkan cara menggunakan `renew-domain`.

### AWS CLI

Untuk memperbarui domain

`renew-domain` Perintah berikut memperbarui domain yang ditentukan selama lima tahun. Untuk mendapatkan nilai `current-expiry-year`, gunakan `get-domain-detail` perintah, dan konversi nilai dari `ExpirationDate` dari format Unix.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains renew-domain \  
  --region us-east-1 \  
  --domain-name example.com \  
  --duration-in-years 5 \  
  --current-expiry-year 2020
```

Output:

```
{  
  "OperationId": "3f28e0ac-126a-4113-9048-cc930example"  
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation-detail`. Untuk informasi lebih lanjut, lihat [get-operation-detail](#).

Registri untuk setiap domain tingkat atas (TLD), seperti .com atau .org, mengontrol jumlah maksimum tahun yang dapat Anda perbarui domain. Untuk mendapatkan periode perpanjangan maksimum untuk domain Anda, lihat bagian “Periode Pendaftaran dan Perpanjangan” untuk TLD Anda di [Domain yang Dapat Anda Daftarkan dengan Amazon Route 53 di Panduan Pengembang Amazon Route 53](#).

Untuk informasi selengkapnya, lihat [Memperbarui Pendaftaran untuk Domain](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [RenewDomain](#) di Referensi AWS CLI Perintah.

## **resend-contact-reachability-email**

Contoh kode berikut menunjukkan cara menggunakan `resend-contact-reachability-email`.

### AWS CLI

Untuk mengirim ulang email konfirmasi ke alamat email saat ini untuk kontak pendaftar

`resend-contact-reachability-email` Perintah berikut mengirimkan ulang email konfirmasi ke alamat email saat ini untuk kontak pendaftar untuk domain `example.com`.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains resend-contact-reachability-email \  
  --region us-east-1
```

```
--region us-east-1 \  
--domain-name example.com
```

#### Output:

```
{  
  "domainName": "example.com",  
  "emailAddress": "moliveira@example.com",  
  "isAlreadyVerified": true  
}
```

Jika nilainya `true`, seperti dalam contoh ini, kontak pendaftar telah mengkonfirmasi bahwa alamat email yang ditentukan dapat dijangkau. `isAlreadyVerified`

Untuk informasi selengkapnya, lihat [Mengirim Ulang Email Otorisasi dan Konfirmasi](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ResendContactReachabilityEmail](#) di Referensi AWS CLI Perintah.

## retrieve-domain-auth-code

Contoh kode berikut menunjukkan cara menggunakan `retrieve-domain-auth-code`.

### AWS CLI

Untuk mendapatkan kode otorisasi untuk domain sehingga Anda dapat mentransfer domain ke registrar lain

`retrieve-domain-auth-code` Perintah berikut mendapatkan kode otorisasi saat ini untuk domain `example.com`. Anda memberikan nilai ini ke registrar domain lain ketika Anda ingin mentransfer domain ke registrar tersebut.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains retrieve-domain-auth-code \  
  --region us-east-1 \  
  --domain-name example.com
```

#### Output:

```
{
```



```
"AuthCode": ")o!v3dJeXampLe"  
}
```

Untuk informasi selengkapnya, lihat [Mentransfer Domain dari Amazon Route 53 ke Registrar Lain di Panduan](#) Pengembang Amazon Route 53.

- Untuk detail API, lihat [RetrieveDomainAuthCode](#) di Referensi AWS CLI Perintah.

## transfer-domain

Contoh kode berikut menunjukkan cara menggunakan `transfer-domain`.

### AWS CLI

Untuk mentransfer domain ke Amazon Route 53

`transfer-domain` Perintah berikut mentransfer domain ke Route 53, dengan parameter yang disediakan oleh file berformat JSON. `C:\temp\transfer-domain.json`

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains transfer-domain \  
  --region us-east-1 \  
  --cli-input-json file://C:\temp\transfer-domain.json
```

Isi dari `transfer-domain.json`:

```
{  
  "DomainName": "example.com",  
  "DurationInYears": 1,  
  "Nameservers": [  
    {  
      "Name": "ns-2048.awsdns-64.com"  
    },  
    {  
      "Name": "ns-2049.awsdns-65.net"  
    },  
    {  
      "Name": "ns-2050.awsdns-66.org"  
    },  
    {  
      "Name": "ns-2051.awsdns-67.co.uk"  
    }  
  ]  
}
```

```
    }  
  ],  
  "AuthCode": ")o!v3dJeXampLe",  
  "AutoRenew": true,  
  "AdminContact": {  
    "FirstName": "Martha",  
    "LastName": "Rivera",  
    "ContactType": "PERSON",  
    "OrganizationName": "Example",  
    "AddressLine1": "1 Main Street",  
    "City": "Anytown",  
    "State": "WA",  
    "CountryCode": "US",  
    "ZipCode": "98101",  
    "PhoneNumber": "+1.8005551212",  
    "Email": "mrivera@example.com"  
  },  
  "RegistrantContact": {  
    "FirstName": "Li",  
    "LastName": "Juan",  
    "ContactType": "PERSON",  
    "OrganizationName": "Example",  
    "AddressLine1": "1 Main Street",  
    "City": "Anytown",  
    "State": "WA",  
    "CountryCode": "US",  
    "ZipCode": "98101",  
    "PhoneNumber": "+1.8005551212",  
    "Email": "ljuan@example.com"  
  },  
  "TechContact": {  
    "FirstName": "Mateo",  
    "LastName": "Jackson",  
    "ContactType": "PERSON",  
    "OrganizationName": "Example",  
    "AddressLine1": "1 Main Street",  
    "City": "Anytown",  
    "State": "WA",  
    "CountryCode": "US",  
    "ZipCode": "98101",  
    "PhoneNumber": "+1.8005551212",  
    "Email": "mjackson@example.com"  
  },  
  "PrivacyProtectAdminContact": true,
```

```
"PrivacyProtectRegistrantContact": true,  
"PrivacyProtectTechContact": true  
}
```

Output:

```
{  
  "OperationId": "b114c44a-9330-47d1-a6e8-a0b11example"  
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation-detail`. Untuk informasi lebih lanjut, lihat [get-operation-detail](#).

Untuk informasi selengkapnya, lihat [Mentransfer Pendaftaran Domain ke Amazon Route 53](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [TransferDomain](#) di Referensi AWS CLI Perintah.

## update-domain-contact-privacy

Contoh kode berikut menunjukkan cara menggunakan `update-domain-contact-privacy`.

### AWS CLI

Untuk memperbarui pengaturan privasi untuk kontak untuk domain

`update-domain-contact-privacy` Perintah berikut mematikan perlindungan privasi untuk kontak administratif untuk domain `example.com`. Perintah ini hanya berjalan di `us-east-1` Wilayah.

Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains update-domain-contact-privacy \  
  --region us-east-1 \  
  --domain-name example.com \  
  --no-admin-privacy
```

Output:

```
{  
  "OperationId": "b3a219e9-d801-4244-b533-b7256example"  
}
```

```
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya `get-operation-detail`. Untuk informasi lebih lanjut, lihat [get-operation-detail](#).

Untuk informasi selengkapnya, lihat [Mengaktifkan atau Menonaktifkan Perlindungan Privasi untuk Informasi Kontak untuk Domain di Panduan](#) Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateDomainContactPrivacy](#) di Referensi AWS CLI Perintah.

## update-domain-contact

Contoh kode berikut menunjukkan cara menggunakan `update-domain-contact`.

### AWS CLI

Untuk memperbarui informasi kontak untuk domain

`update-domain-contact` Perintah berikut memperbarui informasi kontak untuk domain, mendapatkan parameter dari file berformat JSON. `C:\temp\update-domain-contact.json`

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains update-domain-contact \
  --region us-east-1 \
  --cli-input-json file://C:\temp\update-domain-contact.json
```

Isi dari `update-domain-contact.json`:

```
{
  "AdminContact": {
    "AddressLine1": "101 Main Street",
    "AddressLine2": "Suite 1a",
    "City": "Seattle",
    "ContactType": "COMPANY",
    "CountryCode": "US",
    "Email": "w.xiulan@example.com",
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "OrganizationName": "Example",
    "PhoneNumber": "+1.8005551212",
```

```
    "State": "WA",
    "ZipCode": "98101"
  },
  "DomainName": "example.com",
  "RegistrantContact": {
    "AddressLine1": "101 Main Street",
    "AddressLine2": "Suite 1a",
    "City": "Seattle",
    "ContactType": "COMPANY",
    "CountryCode": "US",
    "Email": "w.xiulan@example.com",
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "OrganizationName": "Example",
    "PhoneNumber": "+1.8005551212",
    "State": "WA",
    "ZipCode": "98101"
  },
  "TechContact": {
    "AddressLine1": "101 Main Street",
    "AddressLine2": "Suite 1a",
    "City": "Seattle",
    "ContactType": "COMPANY",
    "CountryCode": "US",
    "Email": "w.xiulan@example.com",
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "OrganizationName": "Example",
    "PhoneNumber": "+1.8005551212",
    "State": "WA",
    "ZipCode": "98101"
  }
}
```

#### Output:

```
{
  "OperationId": "b3a219e9-d801-4244-b533-b7256example"
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya [get-domain-detail](#). Untuk informasi selengkapnya, lihat [Memperbarui Informasi Kontak untuk Domain](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateDomainContact](#) di Referensi AWS CLI Perintah.

## update-domain-nameservers

Contoh kode berikut menunjukkan cara menggunakan `update-domain-nameservers`.

### AWS CLI

Untuk memperbarui server nama untuk domain

`update-domain-nameservers` Perintah berikut memperbarui server nama untuk domain.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains update-domain-nameservers \
  --region us-east-1 \
  --domain-name example.com \
  --nameservers Name=ns-1.awsdns-01.org Name=ns-2.awsdns-02.co.uk
Name=ns-3.awsdns-03.net Name=ns-4.awsdns-04.com
```

Output:

```
{
  "OperationId": "f1691ec4-0e7a-489e-82e0-b19d3example"
}
```

Untuk mengonfirmasi bahwa operasi berhasil, Anda dapat menjalankannya [get-domain-detail](#).

Untuk informasi selengkapnya, lihat [Menambahkan atau Mengubah Server Nama dan Catatan Glue untuk Domain](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateDomainNameservers](#) di Referensi AWS CLI Perintah.

## update-tags-for-domain

Contoh kode berikut menunjukkan cara menggunakan `update-tags-for-domain`.

### AWS CLI

Untuk menambah atau memperbarui tag untuk domain

`update-tags-for-domain` Perintah berikut menambahkan atau memperbarui dua kunci dan nilai yang sesuai untuk domain `example.com`. Untuk memperbarui nilai kunci, cukup sertakan kunci dan nilai baru. Anda dapat menambahkan atau memperbarui tag hanya dalam satu domain pada satu waktu.

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains update-tags-for-domain \  
  --region us-east-1 \  
  --domain-name example.com \  
  --tags-to-update "Key=key1,Value=value1" "Key=key2,Value=value2"
```

Perintah ini tidak menghasilkan output. Untuk mengonfirmasi bahwa tag telah ditambahkan atau diperbarui, Anda dapat menjalankannya [list-tags-for-domain](#).

Untuk informasi selengkapnya, lihat [Menandai Sumber Daya Amazon Route 53](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateTagsForDomain](#) di Referensi AWS CLI Perintah.

## view-billing

Contoh kode berikut menunjukkan cara menggunakan `view-billing`.

### AWS CLI

Untuk mendapatkan informasi penagihan untuk biaya pendaftaran domain untuk akun saat ini AWS

`view-billing` Perintah berikut mengembalikan semua catatan penagihan terkait domain untuk akun saat ini untuk periode dari 1 Januari 2018 (1514764800 dalam waktu Unix) dan tengah malam pada tanggal 31 Desember 2019 (1577836800 dalam waktu Unix).

Perintah ini hanya berjalan di `us-east-1` Wilayah. Jika wilayah default Anda diatur ke `us-east-1`, Anda dapat menghilangkan `region` parameter.

```
aws route53domains view-billing \  
  --region us-east-1 \  
  --start-time 1514764800 \  
  --end-time 1577836800
```

## Output:

```
{
  "BillingRecords": [
    {
      "DomainName": "example.com",
      "Operation": "RENEW_DOMAIN",
      "InvoiceId": "149962827",
      "BillDate": 1536618063.181,
      "Price": 12.0
    },
    {
      "DomainName": "example.com",
      "Operation": "RENEW_DOMAIN",
      "InvoiceId": "290913289",
      "BillDate": 1568162630.884,
      "Price": 12.0
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [ViewBilling](#) di Referensi API Amazon Route 53.

- Untuk detail API, lihat [ViewBilling](#) di Referensi AWS CLI Perintah.

## Contoh Route 53 Resolver menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface dengan Route 53 Resolver.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)



## Tindakan

### **associate-firewall-rule-group**

Contoh kode berikut menunjukkan cara menggunakan `associate-firewall-rule-group`.

#### AWS CLI

Untuk mengaitkan grup aturan firewall dengan VPC

`associate-firewall-rule-group` Contoh berikut mengaitkan grup aturan DNS Firewall dengan Amazon VPC.

```
aws route53resolver associate-firewall-rule-group \  
  --name test-association \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample \  
  --vpc-id vpc-31e92222 \  
  --priority 101
```

Output:

```
{  
  "FirewallRuleGroupAssociation": {  
    "Id": "rslvr-frgassoc-57e8873d7example",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-  
association/rslvr-frgassoc-57e8873d7example",  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "VpcId": "vpc-31e92222",  
    "Name": "test-association",  
    "Priority": 101,  
    "MutationProtection": "DISABLED",  
    "Status": "UPDATING",  
    "StatusMessage": "Creating Firewall Rule Group Association",  
    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:47:48.755768Z",  
    "ModificationTime": "2021-05-25T21:47:48.755768Z"  
  }  
}
```

Untuk selengkapnya, lihat [Mengelola asosiasi antara grup aturan DNS Firewall VPC dan Route 53 Resolver DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [AssociateFirewallRuleGroup](#) di Referensi AWS CLI Perintah.

## associate-resolver-endpoint-ip-address

Contoh kode berikut menunjukkan cara menggunakan `associate-resolver-endpoint-ip-address`.

### AWS CLI

Untuk mengaitkan alamat IP lain dengan titik akhir Resolver

`associate-resolver-endpoint-ip-address` Contoh berikut mengaitkan alamat IP lain dengan titik akhir Resolver masuk. Jika Anda hanya menentukan subnet ID dan menghilangkan alamat IP dari `--ip-address` parameter, Resolver memilih alamat IP untuk Anda dari antara alamat IP yang tersedia di subnet yang ditentukan.

```
aws route53resolver associate-resolver-endpoint-ip-address \
  --resolver-endpoint-id rslvr-in-497098ad5example \
  --ip-address="SubnetId=subnet-12d8exam,Ip=192.0.2.118"
```

Output:

```
{
  "ResolverEndpoint": {
    "Id": "rslvr-in-497098ad5example",
    "CreatorRequestId": "AWSConsole.25.0123456789",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/rslvr-in-497098ad5example",
    "Name": "my-inbound-endpoint",
    "SecurityGroupIds": [
      "sg-05cd7b25d6example"
    ],
    "Direction": "INBOUND",
    "IpAddressCount": 3,
    "HostVPCId": "vpc-304bexam",
    "Status": "UPDATING",
    "StatusMessage": "Updating the Resolver Endpoint",
    "CreationTime": "2020-01-02T23:25:45.538Z",
    "ModificationTime": "2020-01-02T23:25:45.538Z"
  }
}
```

Untuk selengkapnya, lihat [Nilai yang Anda Tentukan Saat Membuat atau Mengedit Titik Akhir Masuk](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [AssociateResolverEndpointIpAddress](#) di Referensi AWS CLI Perintah.

## associate-resolver-rule

Contoh kode berikut menunjukkan cara menggunakan `associate-resolver-rule`.

### AWS CLI

Untuk mengaitkan aturan Resolver dengan VPC

`associate-resolver-rule` Contoh berikut mengaitkan aturan Resolver dengan VPC Amazon. Setelah Anda menjalankan perintah, Resolver mulai meneruskan kueri DNS ke jaringan Anda berdasarkan pengaturan dalam aturan, seperti nama domain kueri yang diteruskan.

```
aws route53resolver associate-resolver-rule \
  --name my-resolver-rule-association \
  --resolver-rule-id rslvr-rr-42b60677c0example \
  --vpc-id vpc-304bexam
```

Output:

```
{
  "ResolverRuleAssociation": {
    "Id": "rslvr-rrassoc-d61cbb2c8bexample",
    "ResolverRuleId": "rslvr-rr-42b60677c0example",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "CREATING",
    "StatusMessage": "[Trace id: 1-5dc5a8fa-ec2cc480d2ef07617example] Creating
the association."
  }
}
```

Untuk informasi selengkapnya, lihat [Meneruskan Kueri DNS Keluar ke Jaringan Anda](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [AssociateResolverRule](#) di Referensi AWS CLI Perintah.

## create-firewall-domain-list

Contoh kode berikut menunjukkan cara menggunakan `create-firewall-domain-list`.

## AWS CLI

Untuk membuat daftar domain Route 53 Resolver DNS Firewall

`create-firewall-domain-list` Contoh berikut membuat daftar domain Route 53 Resolver DNS Firewall, bernama test, di akun Anda. AWS

```
aws route53resolver create-firewall-domain-list \  
  --creator-request-id my-request-id \  
  --name test
```

Output:

```
{  
  "FirewallDomainList": {  
    "Id": "rslvr-fdl-d61cbb2cbexample",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/  
rslvr-fdl-d61cbb2cbexample",  
    "Name": "test",  
    "DomainCount": 0,  
    "Status": "COMPLETE",  
    "StatusMessage": "Created Firewall Domain List",  
    "CreatorRequestId": "my-request-id",  
    "CreationTime": "2021-05-25T15:55:51.115365Z",  
    "ModificationTime": "2021-05-25T15:55:51.115365Z"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengelola daftar domain Anda sendiri](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CreateFirewallDomainList](#) di Referensi AWS CLI Perintah.

## `create-firewall-rule-group`

Contoh kode berikut menunjukkan cara menggunakan `create-firewall-rule-group`.

## AWS CLI

Untuk membuat grup aturan Firewall

`create-firewall-rule-group` Contoh berikut membuat grup aturan DNS Firewall.

```
aws route53resolver create-firewall-rule-group \  
  --creator-request-id my-request-id \  
  --name test
```

### Output:

```
{  
  "FirewallRuleGroup": {  
    "Id": "rslvr-frg-47f93271fexample",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/  
rslvr-frg-47f93271fexample",  
    "Name": "test",  
    "RuleCount": 0,  
    "Status": "COMPLETE",  
    "StatusMessage": "Created Firewall Rule Group",  
    "OwnerId": "123456789012",  
    "CreatorRequestId": "my-request-id",  
    "ShareStatus": "NOT_SHARED",  
    "CreationTime": "2021-05-25T18:59:26.490017Z",  
    "ModificationTime": "2021-05-25T18:59:26.490017Z"  
  }  
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CreateFirewallRuleGroup](#) di Referensi AWS CLI Perintah.

## create-firewall-rule

Contoh kode berikut menunjukkan cara menggunakan `create-firewall-rule`.

### AWS CLI

Untuk membuat aturan firewall

`create-firewall-rule` Contoh berikut membuat aturan firewall dalam aturan DNS Firewall untuk domain yang tercantum dalam daftar domain DNS Firewall.

```
aws route53resolver create-firewall-rule \  
  --name allow-rule \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample \  
  --rule-name allow-rule
```

```
--firewall-domain-list-id rslvr-fdl-9e956e9ffexample \
--priority 101 \
--action ALLOW
```

Output:

```
{
  "FirewallRule": {
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
    "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
    "Name": "allow-rule",
    "Priority": 101,
    "Action": "ALLOW",
    "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
    "CreationTime": "2021-05-25T21:44:00.346093Z",
    "ModificationTime": "2021-05-25T21:44:00.346093Z"
  }
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [CreateFirewallRule](#) di Referensi AWS CLI Perintah.

## create-resolver-endpoint

Contoh kode berikut menunjukkan cara menggunakan `create-resolver-endpoint`.

### AWS CLI

Untuk membuat titik akhir Resolver masuk

`create-resolver-endpoint` Contoh berikut menciptakan titik akhir Resolver inbound. Anda dapat menggunakan perintah yang sama untuk membuat endpoint masuk dan keluar.

```
aws route53resolver create-resolver-endpoint --name my-inbound-endpoint -- creator-request-id 2020-01-01-18:47 -- security-group-ids "sg-f62bexam" --direction INBOUND --ip-address = subnet-ba47exam, IP = 192.0.2.255 = subnet-12d8exam, IP = 192.0.2.254 SubnetId SubnetId
```

Output:

```
{
```

```

    "ResolverEndpoint": {
      "Id": "rslvr-in-f9ab8a03f1example",
      "CreatorRequestId": "2020-01-01-18:47",
      "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-in-f9ab8a03f1example",
      "Name": "my-inbound-endpoint",
      "SecurityGroupIds": [
        "sg-f62bexam"
      ],
      "Direction": "INBOUND",
      "IpAddressCount": 2,
      "HostVPCId": "vpc-304examp",
      "Status": "CREATING",
      "StatusMessage": "[Trace id: 1-5dc1ff84-f3477826e4a190025example] Creating
the Resolver Endpoint",
      "CreationTime": "2020-01-01T23:02:29.583Z",
      "ModificationTime": "2020-01-01T23:02:29.583Z"
    }
  }
}

```

Untuk membuat titik akhir Resolver keluar

`create-resolver-endpoint` Contoh berikut membuat endpoint resolver keluar menggunakan nilai-nilai dalam dokumen berformat JSON. `create-outbound-resolver-endpoint.json`

```

aws route53resolver create-resolver-endpoint \
  --cli-input-json file://c:\temp\create-outbound-resolver-endpoint.json

```

Isi dari `create-outbound-resolver-endpoint.json`:

```

{
  "CreatorRequestId": "2020-01-01-18:47",
  "Direction": "OUTBOUND",
  "IpAddresses": [
    {
      "Ip": "192.0.2.255",
      "SubnetId": "subnet-ba47exam"
    },
    {
      "Ip": "192.0.2.254",
      "SubnetId": "subnet-12d8exam"
    }
  ],
}

```

```
"Name": "my-outbound-endpoint",
"SecurityGroupIds": [ "sg-05cd7b25d6example" ],
"Tags": [
  {
    "Key": "my-key-name",
    "Value": "my-key-value"
  }
]
}
```

Untuk informasi lebih lanjut, lihat [Menyelesaikan permintaan DNS antara VPC dan jaringan Anda](#) di Panduan Developer Amazon Route 53.

- Untuk detail API, lihat [CreateResolverEndpoint](#) di Referensi AWS CLI Perintah.

## create-resolver-rule

Contoh kode berikut menunjukkan cara menggunakan `create-resolver-rule`.

### AWS CLI

Untuk membuat aturan Resolver

`create-resolver-rule` Contoh berikut membuat aturan penerusan Resolver. Aturan menggunakan `rslvr-out-d5e5920e37example` untuk meneruskan kueri DNS ke alamat IP 10.24.8.75 dan 10.24.8.156. `example.com`

```
aws route53resolver create-resolver-rule \
  --creator-request-id 2020-01-02-18:47 \
  --domain-name example.com \
  --name my-rule \
  --resolver-endpoint-id rslvr-out-d5e5920e37example \
  --rule-type FORWARD \
  --target-ips "Ip=10.24.8.75" "Ip=10.24.8.156"
```

Output:

```
{
  "ResolverRule": {
    "Status": "COMPLETE",
    "RuleType": "FORWARD",
    "ResolverEndpointId": "rslvr-out-d5e5920e37example",
```



```

    "Name": "my-rule",
    "DomainName": "example.com.",
    "CreationTime": "2022-05-10T21:35:30.923187Z",
    "TargetIps": [
      {
        "Ip": "10.24.8.75",
        "Port": 53
      },
      {
        "Ip": "10.24.8.156",
        "Port": 53
      }
    ],
    "CreatorRequestId": "2022-05-10-16:33",
    "ModificationTime": "2022-05-10T21:35:30.923187Z",
    "ShareStatus": "NOT_SHARED",
    "Arn": "arn:aws:route53resolver:us-east-1:111117012054:resolver-rule/rslvr-rr-b1e0b905e93611111",
    "OwnerId": "111111111111",
    "Id": "rslvr-rr-rslvr-rr-b1e0b905e93611111",
    "StatusMessage": "[Trace id: 1-22222222-3e56afcc71a3724664f22e24]
Successfully created Resolver Rule."
  }
}

```

- Untuk detail API, lihat [CreateResolverRule](#) di Referensi AWS CLI Perintah.

## delete-firewall-domain-list

Contoh kode berikut menunjukkan cara menggunakan `delete-firewall-domain-list`.

### AWS CLI

Untuk menghapus daftar domain Route 53 Resolver DNS Firewall

`delete-firewall-domain-list` Contoh berikut menghapus daftar domain Route 53 Resolver DNS Firewall, bernama test, di akun Anda. AWS

```
aws route53resolver delete-firewall-domain-list \
  --firewall-domain-list-id rslvr-fdl-9e956e9ffexample
```

Output:

```
{
  "FirewallDomainList": {
    "Id": "rslvr-fdl-9e956e9ffexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/rslvr-fdl-9e956e9ffexample",
    "Name": "test",
    "DomainCount": 6,
    "Status": "DELETING",
    "StatusMessage": "Deleting the Firewall Domain List",
    "CreatorRequestId": "my-request-id",
    "CreationTime": "2021-05-25T15:55:51.115365Z",
    "ModificationTime": "2021-05-25T18:58:05.588024Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola daftar domain Anda sendiri](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [DeleteFirewallDomainList](#) di Referensi AWS CLI Perintah.

## delete-firewall-rule-group

Contoh kode berikut menunjukkan cara menggunakan `delete-firewall-rule-group`.

### AWS CLI

Untuk menghapus grup aturan firewall

`delete-firewall-rule-group` Contoh berikut menghapus grup aturan firewall.

```
aws route53resolver delete-firewall-rule-group \
  --firewall-rule-group-id rslvr-frg-47f93271fexample
```

Output:

```
{
  "FirewallRuleGroup": {
    "Id": "rslvr-frg-47f93271fexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/rslvr-frg-47f93271fexample",
    "Name": "test",
    "RuleCount": 0,
  }
}
```

```

    "Status": "UPDATING",
    "StatusMessage": "Updating Firewall Rule Group",
    "OwnerId": "123456789012",
    "CreatorRequestId": "my-request-id",
    "ShareStatus": "NOT_SHARED",
    "CreationTime": "2021-05-25T18:59:26.490017Z",
    "ModificationTime": "2021-05-25T21:51:53.028688Z"
  }
}

```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [DeleteFirewallRuleGroup](#) di Referensi AWS CLI Perintah.

## delete-firewall-rule

Contoh kode berikut menunjukkan cara menggunakan `delete-firewall-rule`.

### AWS CLI

Untuk menghapus aturan firewall

`delete-firewall-rule` Contoh berikut menghapus aturan firewall tertentu.

```

aws route53resolver delete-firewall-rule \
  --firewall-rule-group-id rslvr-frg-47f93271fexample \
  --firewall-domain-list-id rslvr-fdl-9e956e9ffexample

```

Output:

```

{
  "FirewallRule": {
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
    "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
    "Name": "allow-rule",
    "Priority": 102,
    "Action": "ALLOW",
    "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
    "CreationTime": "2021-05-25T21:44:00.346093Z",
    "ModificationTime": "2021-05-25T21:45:59.611600Z"
  }
}

```

```
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [DeleteFirewallRule](#) di Referensi AWS CLI Perintah.

## delete-resolver-endpoint

Contoh kode berikut menunjukkan cara menggunakan `delete-resolver-endpoint`.

### AWS CLI

Untuk menghapus titik akhir Resolver

`delete-resolver-endpoint` Contoh berikut menghapus titik akhir yang ditentukan.

Penting Jika Anda menghapus titik akhir masuk, kueri DNS dari jaringan Anda tidak lagi diteruskan ke Resolver di VPC yang Anda tentukan di titik akhir. Jika Anda menghapus titik akhir keluar, Resolver berhenti meneruskan kueri DNS dari VPC ke jaringan untuk aturan yang menentukan titik akhir keluar yang dihapus.

```
aws route53resolver delete-resolver-endpoint \  
  --resolver-endpoint-id rslvr-in-497098ad59example
```

Output:

```
{  
  "ResolverEndpoint": {  
    "Id": "rslvr-in-497098ad59example",  
    "CreatorRequestId": "AWSConsole.25.157290example",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/  
rslvr-in-497098ad59example",  
    "Name": "my-inbound-endpoint",  
    "SecurityGroupIds": [  
      "sg-05cd7b25d6example"  
    ],  
    "Direction": "INBOUND",  
    "IpAddressCount": 5,  
    "HostVPCId": "vpc-304bexam",  
    "Status": "DELETING",
```

```

    "StatusMessage": "[Trace id: 1-5dc5b658-811b5be0922bbc382example] Deleting
ResolverEndpoint.",
    "CreationTime": "2020-01-01T23:25:45.538Z",
    "ModificationTime": "2020-01-02T23:25:45.538Z"
  }
}

```

- Untuk detail API, lihat [DeleteResolverEndpoint](#) di Referensi AWS CLI Perintah.

## delete-resolver-rule

Contoh kode berikut menunjukkan cara menggunakan `delete-resolver-rule`.

### AWS CLI

Untuk menghapus aturan Resolver

`delete-resolver-rule` Contoh berikut menghapus aturan yang ditentukan.

Catatan Jika aturan dikaitkan dengan VPC apa pun, Anda harus terlebih dahulu memisahkan aturan dari VPC sebelum Anda dapat menghapusnya.

```

aws route53resolver delete-resolver-rule \
  --resolver-rule-id rslvr-rr-5b3809426bexample

```

### Output:

```

{
  "ResolverRule": {
    "Id": "rslvr-rr-5b3809426bexample",
    "CreatorRequestId": "2020-01-03-18:47",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-rr-5b3809426bexample",
    "DomainName": "zenith.example.com.",
    "Status": "DELETING",
    "StatusMessage": "[Trace id: 1-5dc5e05b-602e67b052cb74f05example] Deleting Resolver Rule.",
    "RuleType": "FORWARD",
    "Name": "my-resolver-rule",
    "TargetIps": [
      {
        "Ip": "192.0.2.50",
        "Port": 53
      }
    ]
  }
}

```

```

    }
  ],
  "ResolverEndpointId": "rslvr-out-d5e5920e3example",
  "OwnerId": "111122223333",
  "ShareStatus": "NOT_SHARED"
}
}

```

- Untuk detail API, lihat [DeleteResolverRule](#) di Referensi AWS CLI Perintah.

## disassociate-firewall-rule-group

Contoh kode berikut menunjukkan cara menggunakan `disassociate-firewall-rule-group`.

### AWS CLI

Untuk memisahkan grup aturan firewall dari VPC

`disassociate-firewall-rule-group` Contoh berikut memisahkan grup aturan DNS Firewall dari VPC Amazon.

```
aws route53resolver disassociate-firewall-rule-group \
  --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example
```

### Output:

```
{
  "FirewallRuleGroupAssociation": {
    "Id": "rslvr-frgassoc-57e8873d7example",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-association/rslvr-frgassoc-57e8873d7example",
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
    "VpcId": "vpc-31e92222",
    "Name": "test-association",
    "Priority": 103,
    "MutationProtection": "DISABLED",
    "Status": "DELETING",
    "StatusMessage": "Deleting the Firewall Rule Group Association",
    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
    "CreationTime": "2021-05-25T21:47:48.755768Z",
    "ModificationTime": "2021-05-25T21:51:02.377887Z"
  }
}
```

```
}

```

Untuk selengkapnya, lihat [Mengelola asosiasi antara grup aturan DNS Firewall VPC dan Route 53 Resolver DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [DisassociateFirewallRuleGroup](#) di Referensi AWS CLI Perintah.

## disassociate-resolver-endpoint-ip-address

Contoh kode berikut menunjukkan cara menggunakan `disassociate-resolver-endpoint-ip-address`.

### AWS CLI

Untuk memisahkan alamat IP dari titik akhir Resolver

`disassociate-resolver-endpoint-ip-address` Contoh berikut menghapus alamat IP dari titik akhir masuk atau keluar Resolver tertentu.

Catatan Endpoint harus memiliki setidaknya dua alamat IP. Jika titik akhir saat ini hanya memiliki dua alamat IP dan Anda ingin mengganti satu alamat dengan alamat lain, Anda harus terlebih dahulu menggunakan [associate-resolver-endpoint-ip-address](#) untuk mengaitkan alamat IP baru. Kemudian Anda dapat memisahkan salah satu alamat IP asli dari titik akhir.

```
aws route53resolver disassociate-resolver-endpoint-ip-address \
  --resolver-endpoint-id rslvr-in-f9ab8a03f1example \
  --ip-address="SubnetId=subnet-12d8a459,Ip=172.31.40.121"
```

### Output:

```
{
  "ResolverEndpoint": {
    "Id": "rslvr-in-f9ab8a03f1example",
    "CreatorRequestId": "2020-01-01-18:47",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/rslvr-in-f9ab8a03f1example",
    "Name": "my-inbound-endpoint",
    "SecurityGroupIds": [
      "sg-f62bexam"
    ],
    "Direction": "INBOUND",
    "IpAddressCount": 3,
```

```

    "HostVPCId": "vpc-304bexam",
    "Status": "UPDATING",
    "StatusMessage": "Updating the Resolver Endpoint",
    "CreationTime": "2020-01-01T23:02:29.583Z",
    "ModificationTime": "2020-01-05T23:02:29.583Z"
  }
}

```

- Untuk detail API, lihat [DisassociateResolverEndpointIpAddress](#) di Referensi AWS CLI Perintah.

## disassociate-resolver-rule

Contoh kode berikut menunjukkan cara menggunakan `disassociate-resolver-rule`.

### AWS CLI

Untuk memisahkan aturan Resolver dari VPC Amazon

`disassociate-resolver-rule` Contoh berikut menghapus hubungan antara aturan Resolver yang ditentukan dan VPC yang ditentukan. Anda dapat memisahkan aturan dari VPC dalam keadaan berikut:

Untuk kueri DNS yang berasal dari VPC ini, Anda ingin Resolver menghentikan penerusan kueri ke jaringan Anda untuk nama domain yang ditentukan dalam aturan. Anda ingin menghapus aturan penerusan. Jika aturan saat ini terkait dengan satu atau beberapa VPC, Anda harus memisahkan aturan dari semua VPC sebelum dapat menghapusnya.

```

aws route53resolver disassociate-resolver-rule \
  --resolver-rule-id rslvr-rr-4955cb98ceexample \
  --vpc-id vpc-304bexam

```

Output:

```

{
  "ResolverRuleAssociation": {
    "Id": "rslvr-rrassoc-322f4e8b9cexample",
    "ResolverRuleId": "rslvr-rr-4955cb98ceexample",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "DELETING",
    "StatusMessage": "[Trace id: 1-5dc5ffa2-a26c38004c1f94006example] Deleting Association"
  }
}

```



```
}  
}
```

- Untuk detail API, lihat [DisassociateResolverRule](#) di Referensi AWS CLI Perintah.

## get-firewall-config

Contoh kode berikut menunjukkan cara menggunakan `get-firewall-config`.

### AWS CLI

Untuk mendapatkan konfigurasi firewall untuk VPC

`get-firewall-config` Contoh berikut mengambil perilaku DNS Firewall untuk VPC yang ditentukan.

```
aws route53resolver get-firewall-config \  
  --resource-id vpc-31e92222
```

Output:

```
{  
  "FirewallConfig": {  
    "Id": "rslvr-fc-86016850cexample",  
    "ResourceId": "vpc-31e92222",  
    "OwnerId": "123456789012",  
    "FirewallFailOpen": "DISABLED"  
  }  
}
```

Untuk informasi selengkapnya, lihat konfigurasi [DNS Firewall VPC](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetFirewallConfig](#) di Referensi AWS CLI Perintah.

## get-firewall-domain-list

Contoh kode berikut menunjukkan cara menggunakan `get-firewall-domain-list`.

### AWS CLI

Untuk mendapatkan daftar domain Route 53 Resolver DNS Firewall

`get-firewall-domain-list` Contoh berikut mengambil daftar domain dengan ID yang Anda tentukan.

```
aws route53resolver get-firewall-domain-list \
  --firewall-domain-list-id rslvr-fdl-42b60677cexample
```

Output:

```
{
  "FirewallDomainList": {
    "Id": "rslvr-fdl-9e956e9ffexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123457689012:firewall-domain-list/rslvr-fdl-42b60677cexample",
    "Name": "test",
    "DomainCount": 0,
    "Status": "COMPLETE",
    "StatusMessage": "Created Firewall Domain List",
    "CreatorRequestId": "my-request-id",
    "CreationTime": "2021-05-25T15:55:51.115365Z",
    "ModificationTime": "2021-05-25T15:55:51.115365Z"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola daftar domain Anda sendiri](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetFirewallDomainList](#) di Referensi AWS CLI Perintah.

## **get-firewall-rule-group-association**

Contoh kode berikut menunjukkan cara menggunakan `get-firewall-rule-group-association`.

AWS CLI

Untuk mendapatkan asosiasi grup aturan firewall

`get-firewall-rule-group-association` Contoh berikut mengambil asosiasi grup aturan firewall.

```
aws route53resolver get-firewall-rule-group-association \
```

```
--firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example
```

### Output:

```
{
  "FirewallRuleGroupAssociation": {
    "Id": "rslvr-frgassoc-57e8873d7example",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-association/rslvr-frgassoc-57e8873d7example",
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
    "VpcId": "vpc-31e92222",
    "Name": "test-association",
    "Priority": 101,
    "MutationProtection": "DISABLED",
    "Status": "COMPLETE",
    "StatusMessage": "Finished rule group association update",
    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
    "CreationTime": "2021-05-25T21:47:48.755768Z",
    "ModificationTime": "2021-05-25T21:47:48.755768Z"
  }
}
```

Untuk selengkapnya, lihat [Mengelola asosiasi antara grup aturan DNS Firewall VPC dan Route 53 Resolver DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetFirewallRuleGroupAssociation](#) di Referensi AWS CLI Perintah.

## get-firewall-rule-group-policy

Contoh kode berikut menunjukkan cara menggunakan `get-firewall-rule-group-policy`.

### AWS CLI

Untuk mendapatkan kebijakan AWS IAM

`get-firewall-rule-group-policy` Contoh berikut mendapatkan kebijakan AWS Identity and Access Management (AWS IAM) untuk berbagi grup aturan yang ditentukan.

```
aws route53resolver get-firewall-rule-group-policy \
  --arn arn:aws:route53resolver:us-west-2:AWS_ACCOUNT_ID:firewall-rule-group/
  rslvr-frg-47f93271fexample
```

**Output:**

```
{
  "FirewallRuleGroupPolicy": "{\\"Version\\":\\"2012-10-17\\",
  \\"Statement\\":[{\\"Sid\\":\\"test\\",\\"Effect\\":\\"Allow\\",\\"Principal\\":{\\"AWS\\":\\"arn:aws:iam::AWS_ACCOUNT_ID:root\\"},\\"Action\\":
  [\\"route53resolver:GetFirewallRuleGroup\\",\\"route53resolver:ListFirewallRuleGroups\\"],\\"Resource\\":\\"arn:aws:route53resolver:us-east-1:AWS_ACCOUNT_ID:firewall-rule-group/rslvr-frg-47f93271fexample\\"}]}"
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetFirewallRuleGroupPolicy](#) di Referensi AWS CLI Perintah.

**get-firewall-rule-group**

Contoh kode berikut menunjukkan cara menggunakan `get-firewall-rule-group`.

**AWS CLI**

Untuk mendapatkan grup aturan Firewall

`get-firewall-rule-group` Contoh berikut mengambil informasi tentang grup aturan DNS Firewall dengan ID yang Anda berikan.

```
aws route53resolver get-firewall-rule-group \
  --firewall-rule-group-id rslvr-frg-47f93271fexample
```

**Output:**

```
{
  "FirewallRuleGroup": {
    "Id": "rslvr-frg-47f93271fexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/rslvr-frg-47f93271fexample",
    "Name": "test",
    "RuleCount": 0,
    "Status": "COMPLETE",
    "StatusMessage": "Created Firewall Rule Group",
```

```

    "OwnerId": "123456789012",
    "CreatorRequestId": "my-request-id",
    "ShareStatus": "NOT_SHARED",
    "CreationTime": "2021-05-25T18:59:26.490017Z",
    "ModificationTime": "2021-05-25T18:59:26.490017Z"
  }
}

```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetFirewallRuleGroup](#) di Referensi AWS CLI Perintah.

## get-resolver-endpoint

Contoh kode berikut menunjukkan cara menggunakan `get-resolver-endpoint`.

### AWS CLI

Untuk mendapatkan informasi tentang titik akhir Resolver

`get-resolver-endpoint` Contoh berikut menampilkan rincian untuk titik akhir yang ditentukan keluar. Anda dapat menggunakan `get-resolver-endpoint` titik akhir masuk dan keluar dengan menentukan ID titik akhir yang berlaku.

```

aws route53resolver get-resolver-endpoint \
  --resolver-endpoint-id rslvr-out-d5e5920e37example

```

### Output:

```

{
  "ResolverEndpoint": {
    "Id": "rslvr-out-d5e5920e37example",
    "CreatorRequestId": "2020-01-01-18:47",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/rslvr-out-d5e5920e37example",
    "Name": "my-outbound-endpoint",
    "SecurityGroupIds": [
      "sg-05cd7b25d6example"
    ],
    "Direction": "OUTBOUND",
  }
}

```

```

    "IpAddressCount": 2,
    "HostVPCId": "vpc-304bexam",
    "Status": "OPERATIONAL",
    "StatusMessage": "This Resolver Endpoint is operational.",
    "CreationTime": "2020-01-01T23:50:50.979Z",
    "ModificationTime": "2020-01-02T23:50:50.979Z"
  }
}

```

Untuk selengkapnya, lihat [Nilai yang Anda Tentukan Saat Membuat atau Mengedit Titik Akhir Masuk](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetResolverEndpoint](#) di Referensi AWS CLI Perintah.

## get-resolver-rule-association

Contoh kode berikut menunjukkan cara menggunakan `get-resolver-rule-association`.

### AWS CLI

Untuk mendapatkan informasi tentang hubungan antara aturan Resolver dan VPC

`get-resolver-rule-association` Contoh berikut menampilkan detail tentang hubungan antara aturan Resolver tertentu dan VPC. Anda mengaitkan aturan resolver dan VPC menggunakan [associate-resolver-rule](#)

```

aws route53resolver get-resolver-rule-association \
  --resolver-rule-association-id rslvr-rrassoc-d61cbb2c8bexample

```

### Output:

```

{
  "ResolverRuleAssociation": {
    "Id": "rslvr-rrassoc-d61cbb2c8bexample",
    "ResolverRuleId": "rslvr-rr-42b60677c0example",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "COMPLETE",
    "StatusMessage": ""
  }
}

```

- Untuk detail API, lihat [GetResolverRuleAssociation](#) di Referensi AWS CLI Perintah.

## get-resolver-rule

Contoh kode berikut menunjukkan cara menggunakan `get-resolver-rule`.

### AWS CLI

Untuk mendapatkan informasi tentang aturan Resolver

`get-resolver-rule` Contoh berikut menampilkan detail tentang aturan Resolver yang ditentukan, seperti nama domain tempat aturan meneruskan kueri DNS dan ID titik akhir resolver keluar yang terkait dengan aturan tersebut.

```
aws route53resolver get-resolver-rule \
  --resolver-rule-id rslvr-rr-42b60677c0example
```

Output:

```
{
  "ResolverRule": {
    "Id": "rslvr-rr-42b60677c0example",
    "CreatorRequestId": "2020-01-01-18:47",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-rr-42b60677c0example",
    "DomainName": "example.com.",
    "Status": "COMPLETE",
    "StatusMessage": "[Trace id: 1-5dc4b177-ff1d9d001a0f80005example]
Successfully created Resolver Rule.",
    "RuleType": "FORWARD",
    "Name": "my-rule",
    "TargetIps": [
      {
        "Ip": "192.0.2.45",
        "Port": 53
      }
    ],
    "ResolverEndpointId": "rslvr-out-d5e5920e37example",
    "OwnerId": "111122223333",
    "ShareStatus": "NOT_SHARED"
  }
}
```

Untuk selengkapnya, lihat [Nilai yang Anda Tentukan Saat Membuat atau Mengedit Aturan](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [GetResolverRule](#) di Referensi AWS CLI Perintah.

## **import-firewall-domains**

Contoh kode berikut menunjukkan cara menggunakan `import-firewall-domains`.

### AWS CLI

Untuk mengimpor domain ke dalam daftar domain

`import-firewall-domains` Contoh berikut mengimpor satu set domain dari file ke dalam daftar domain DNS Firewall yang Anda tentukan.

```
aws route53resolver import-firewall-domains \  
  --firewall-domain-list-id rslvr-fdl-d61cbb2cbexample \  
  --operation REPLACE \  
  --domain-file-url s3://PATH/TO/YOUR/FILE
```

Output:

```
{  
  "Id": "rslvr-fdl-d61cbb2cbexample",  
  "Name": "test",  
  "Status": "IMPORTING",  
  "StatusMessage": "Importing domains from provided file."  
}
```

Untuk informasi selengkapnya, lihat [Mengelola daftar domain Anda sendiri](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ImportFirewallDomains](#) di Referensi AWS CLI Perintah.

## **list-firewall-configs**

Contoh kode berikut menunjukkan cara menggunakan `list-firewall-configs`.

### AWS CLI

Untuk membuat daftar konfigurasi firewall



`list-firewall-configs` Contoh berikut mencantumkan konfigurasi DNS Firewall Anda.

```
aws route53resolver list-firewall-configs
```

Output:

```
{
  "FirewallConfigs": [
    {
      "Id": "rslvr-fc-86016850cexample",
      "ResourceId": "vpc-31e92222",
      "OwnerId": "123456789012",
      "FirewallFailOpen": "DISABLED"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat konfigurasi [DNS Firewall VPC](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListFirewallConfigs](#) di Referensi AWS CLI Perintah.

## `list-firewall-domain-lists`

Contoh kode berikut menunjukkan cara menggunakan `list-firewall-domain-lists`.

AWS CLI

Untuk membuat daftar semua daftar domain Route 53 Resolver DNS Firewall

`list-firewall-domain-lists` Contoh berikut mencantumkan semua daftar domain.

```
aws route53resolver list-firewall-domain-lists
```

Output:

```
{
  "FirewallDomainLists": [
    {
      "Id": "rslvr-fdl-2c46f2ecfexample",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/rslvr-fdl-2c46f2ecfexample",
    }
  ]
}
```

```

    "Name": "AWSManagedDomainsMalwareDomainList",
    "CreatorRequestId": "AWSManagedDomainsMalwareDomainList",
    "ManagedOwnerName": "Route 53 Resolver DNS Firewall"
  },
  {
    "Id": "rslvr-fdl-aa970e9e1example",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-
list/rslvr-fdl-aa970e9e1example",
    "Name": "AWSManagedDomainsBotnetCommandandControl",
    "CreatorRequestId": "AWSManagedDomainsBotnetCommandandControl",
    "ManagedOwnerName": "Route 53 Resolver DNS Firewall"
  },
  {
    "Id": "rslvr-fdl-42b60677cexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789111:firewall-domain-
list/rslvr-fdl-42b60677cexample",
    "Name": "test",
    "CreatorRequestId": "my-request-id"
  }
]
}

```

Untuk informasi selengkapnya, lihat [daftar domain Route 53 Resolver DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListFirewallDomainLists](#) di Referensi AWS CLI Perintah.

## list-firewall-domains

Contoh kode berikut menunjukkan cara menggunakan `list-firewall-domains`.

### AWS CLI

Untuk membuat daftar domain dalam daftar domain

`list-firewall-domains` Contoh berikut mencantumkan domain dalam daftar domain DNS Firewall yang Anda tentukan.

```
aws route53resolver list-firewall-domains \
  --firewall-domain-list-id rslvr-fdl-d61cbb2cbexample
```

Output:

```
{
  "Domains": [
    "test1.com.",
    "test2.com.",
    "test3.com."
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola daftar domain Anda sendiri](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListFirewallDomains](#) di Referensi AWS CLI Perintah.

## list-firewall-rule-group-associations

Contoh kode berikut menunjukkan cara menggunakan `list-firewall-rule-group-associations`.

### AWS CLI

Untuk mencantumkan asosiasi grup aturan DNS Firewall

`list-firewall-rule-group-associations` Contoh berikut mencantumkan asosiasi grup aturan DNS Firewall Anda dengan Amazon VPC.

```
aws route53resolver list-firewall-rule-group-associations
```

Output:

```
{
  "FirewallRuleGroupAssociations": [
    {
      "Id": "rslvr-frgassoc-57e8873d7example",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-association/rslvr-frgassoc-57e8873d7example",
      "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
      "VpcId": "vpc-31e92222",
      "Name": "test-association",
      "Priority": 101,
      "MutationProtection": "DISABLED",
      "Status": "UPDATING",
      "StatusMessage": "Creating Firewall Rule Group Association",
    }
  ]
}
```

```
        "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:47:48.755768Z",
        "ModificationTime": "2021-05-25T21:47:48.755768Z"
    }
]
}
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi antara VPC Anda dan grup aturan Route 53 Resolver DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListFirewallRuleGroupAssociations](#) di Referensi AWS CLI Perintah.

## list-firewall-rule-groups

Contoh kode berikut menunjukkan cara menggunakan `list-firewall-rule-groups`.

### AWS CLI

Untuk mendapatkan daftar grup aturan Firewall

`list-firewall-rule-groups` Contoh berikut mencantumkan grup aturan DNS Firewall Anda.

```
aws route53resolver list-firewall-rule-groups
```

Output:

```
{
  "FirewallRuleGroups": [
    {
      "Id": "rslvr-frg-47f93271fexample",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-
group/rslvr-frg-47f93271fexample",
      "Name": "test",
      "OwnerId": "123456789012",
      "CreatorRequestId": "my-request-id",
      "ShareStatus": "NOT_SHARED"
    }
  ]
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListFirewallRuleGroups](#) di Referensi AWS CLI Perintah.

## list-firewall-rules

Contoh kode berikut menunjukkan cara menggunakan `list-firewall-rules`.

### AWS CLI

Untuk daftar aturan firewall

`list-firewall-rules` Contoh berikut mencantumkan semua aturan DNS Firewall Anda dalam grup aturan firewall.

```
aws route53resolver list-firewall-rules \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample
```

Output:

```
{  
  "FirewallRules": [  
    {  
      "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
      "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",  
      "Name": "allow-rule",  
      "Priority": 101,  
      "Action": "ALLOW",  
      "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",  
      "CreationTime": "2021-05-25T21:44:00.346093Z",  
      "ModificationTime": "2021-05-25T21:44:00.346093Z"  
    }  
  ]  
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListFirewallRules](#) di Referensi AWS CLI Perintah.

## list-resolver-endpoint-ip-addresses

Contoh kode berikut menunjukkan cara menggunakan `list-resolver-endpoint-ip-addresses`.

## AWS CLI

Untuk mencantumkan alamat IP untuk titik akhir masuk atau keluar yang ditentukan

`list-resolver-endpoint-ip-addresses` Contoh berikut mencantumkan informasi tentang alamat IP yang terkait dengan titik akhir `rslvr-in-f9ab8a03f1example` masuk. Anda juga dapat menggunakan `list-resolver-endpoint-ip-addresses` titik akhir keluar dengan menentukan ID titik akhir yang berlaku.

```
aws route53resolver list-resolver-endpoint-ip-addresses \  
--resolver-endpoint-id rslvr-in-f9ab8a03f1example
```

Output:

```
{  
  "MaxResults": 10,  
  "IpAddresses": [  
    {  
      "IpId": "rni-1de60cdbfeexample",  
      "SubnetId": "subnet-ba47exam",  
      "Ip": "192.0.2.44",  
      "Status": "ATTACHED",  
      "StatusMessage": "This IP address is operational.",  
      "CreationTime": "2020-01-03T23:02:29.587Z",  
      "ModificationTime": "2020-01-03T23:03:05.555Z"  
    },  
    {  
      "IpId": "rni-aac7085e38example",  
      "SubnetId": "subnet-12d8exam",  
      "Ip": "192.0.2.45",  
      "Status": "ATTACHED",  
      "StatusMessage": "This IP address is operational.",  
      "CreationTime": "2020-01-03T23:02:29.593Z",  
      "ModificationTime": "2020-01-03T23:02:55.060Z"  
    }  
  ]  
}
```

Untuk informasi selengkapnya tentang nilai dalam output, lihat [Nilai yang Anda Tentukan Saat Membuat atau Mengedit Titik Akhir Masuk](#), dan [Nilai yang Anda Tentukan Saat Membuat atau Mengedit Titik Akhir Keluar](#), keduanya dalam Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [ListResolverEndpointIpAddresses](#) di Referensi AWS CLI Perintah.

## list-resolver-endpoints

Contoh kode berikut menunjukkan cara menggunakan `list-resolver-endpoints`.

### AWS CLI

Untuk mencantumkan titik akhir Resolver di Wilayah AWS

`list-resolver-endpoints` Contoh berikut mencantumkan titik akhir Resolver masuk dan keluar yang ada di akun saat ini.

```
aws route53resolver list-resolver-endpoints
```

Output:

```
{
  "MaxResults": 10,
  "ResolverEndpoints": [
    {
      "Id": "rslvr-in-497098ad59example",
      "CreatorRequestId": "2020-01-01-18:47",
      "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-
endpoint/rslvr-in-497098ad59example",
      "Name": "my-inbound-endpoint",
      "SecurityGroupIds": [
        "sg-05cd7b25d6example"
      ],
      "Direction": "INBOUND",
      "IpAddressCount": 2,
      "HostVPCId": "vpc-304bexam",
      "Status": "OPERATIONAL",
      "StatusMessage": "This Resolver Endpoint is operational.",
      "CreationTime": "2020-01-01T23:25:45.538Z",
      "ModificationTime": "2020-01-01T23:25:45.538Z"
    },
    {
      "Id": "rslvr-out-d5e5920e37example",
      "CreatorRequestId": "2020-01-01-18:48",
      "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-
endpoint/rslvr-out-d5e5920e37example",
```

```

    "Name": "my-outbound-endpoint",
    "SecurityGroupIds": [
      "sg-05cd7b25d6example"
    ],
    "Direction": "OUTBOUND",
    "IpAddressCount": 2,
    "HostVPCId": "vpc-304bexam",
    "Status": "OPERATIONAL",
    "StatusMessage": "This Resolver Endpoint is operational.",
    "CreationTime": "2020-01-01T23:50:50.979Z",
    "ModificationTime": "2020-01-01T23:50:50.979Z"
  }
]
}

```

- Untuk detail API, lihat [ListResolverEndpoints](#) di Referensi AWS CLI Perintah.

## list-resolver-rule-associations

Contoh kode berikut menunjukkan cara menggunakan `list-resolver-rule-associations`.

### AWS CLI

Untuk membuat daftar asosiasi antara aturan Resolver dan VPC

`list-resolver-rule-associations` Contoh berikut mencantumkan asosiasi antara aturan resolver dan VPC di akun saat ini. AWS

```
aws route53resolver list-resolver-rule-associations
```

Output:

```

{
  "MaxResults": 30,
  "ResolverRuleAssociations": [
    {
      "Id": "rslvr-autodefined-assoc-vpc-304bexam-internet-resolver",
      "ResolverRuleId": "rslvr-autodefined-rr-internet-resolver",
      "Name": "System Rule Association",
      "VPCId": "vpc-304bexam",
      "Status": "COMPLETE",
    }
  ]
}

```



```
    "StatusMessage": ""
  },
  {
    "Id": "rslvr-rrassoc-d61cbb2c8bexample",
    "ResolverRuleId": "rslvr-rr-42b60677c0example",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "COMPLETE",
    "StatusMessage": ""
  }
]
```

Untuk informasi selengkapnya, lihat [Cara Route 53 Resolver Memajukan Kueri DNS dari VPC ke Jaringan Anda di Panduan Pengembang Amazon Route 53](#).

- Untuk detail API, lihat [ListResolverRuleAssociations](#) di Referensi AWS CLI Perintah.

## list-resolver-rules

Contoh kode berikut menunjukkan cara menggunakan `list-resolver-rules`.

### AWS CLI

Untuk membuat daftar aturan Resolver

`list-resolver-rules` Contoh berikut mencantumkan semua aturan Resolver di akun saat ini AWS .

```
aws route53resolver list-resolver-rules
```

Output:

```
{
  "MaxResults": 30,
  "ResolverRules": [
    {
      "Id": "rslvr-autodefined-rr-internet-resolver",
      "CreatorRequestId": "",
      "Arn": "arn:aws:route53resolver:us-west-2::autodefined-rule/rslvr-
autodefined-rr-internet-resolver",
```

```

        "DomainName": ".",
        "Status": "COMPLETE",
        "RuleType": "RECURSIVE",
        "Name": "Internet Resolver",
        "OwnerId": "Route 53 Resolver",
        "ShareStatus": "NOT_SHARED"
    },
    {
        "Id": "rslvr-rr-42b60677c0example",
        "CreatorRequestId": "2020-01-01-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0bc4e299",
        "DomainName": "example.com.",
        "Status": "COMPLETE",
        "StatusMessage": "[Trace id: 1-5dc4b177-ff1d9d001a0f80005example]
Successfully created Resolver Rule.",
        "RuleType": "FORWARD",
        "Name": "my-rule",
        "TargetIps": [
            {
                "Ip": "192.0.2.45",
                "Port": 53
            }
        ],
        "ResolverEndpointId": "rslvr-out-d5e5920e37example",
        "OwnerId": "111122223333",
        "ShareStatus": "NOT_SHARED"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Cara Route 53 Resolver Memajukan Kueri DNS dari VPC ke Jaringan Anda di Panduan Pengembang Amazon Route 53](#).

- Untuk detail API, lihat [ListResolverRules](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag untuk sumber daya Resolver

`list-tags-for-resource` Contoh berikut mencantumkan tag yang ditetapkan untuk aturan Resolver tertentu.

```
aws route53resolver list-tags-for-resource \
  --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
  rslvr-rr-42b60677c0example"
```

Output:

```
{
  "Tags": [
    {
      "Key": "my-key-1",
      "Value": "my-value-1"
    },
    {
      "Key": "my-key-2",
      "Value": "my-value-2"
    }
  ]
}
```

Untuk informasi tentang penggunaan tag untuk alokasi biaya, lihat [Menggunakan Tag Alokasi Biaya](#) di Panduan Pengguna AWS Billing and Cost Management.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## `put-firewall-rule-group-policy`

Contoh kode berikut menunjukkan cara menggunakan `put-firewall-rule-group-policy`.

### AWS CLI

Untuk melampirkan kebijakan AWS IAM untuk membagikan kebijakan grup aturan Firewall

`put-firewall-rule-group-policy` Contoh berikut melampirkan kebijakan AWS Identity and Access Management (AWS IAM) untuk berbagi grup aturan.

```
aws route53resolver put-firewall-rule-group-policy \
  --firewall-rule-group-policy "{\"Version\":\"2012-10-17\",
  \"Statement\": [{\"Sid\":\"test\", \"Effect\":\"Allow\", \"Principal
```

```
\" : { \"AWS\" : { \"arn:aws:iam::AWS_ACCOUNT_ID:root\" }, \"Action\" :
[ \"route53resolver:GetFirewallRuleGroup\" , \"route53resolver:ListFirewallRuleGroups
\" ], \"Resource\" : { \"arn:aws:route53resolver:us-east-1:AWS_ACCOUNT_ID:firewall-rule-
group/rslvr-frg-47f93271fexample\" } ] }
```

Output:

```
{
  \"ReturnValue\": true
}
```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [PutFirewallRuleGroupPolicy](#) di Referensi AWS CLI Perintah.

## put-resolver-rule-policy

Contoh kode berikut menunjukkan cara menggunakan `put-resolver-rule-policy`.

### AWS CLI

Untuk berbagi aturan Resolver dengan akun lain AWS

`put-resolver-rule-policy` Contoh berikut menentukan aturan Resolver yang ingin Anda bagikan dengan AWS akun lain, akun yang ingin Anda bagikan aturannya, dan operasi terkait aturan yang Anda inginkan agar akun dapat dilakukan sesuai aturan.

Catatan Anda harus menjalankan perintah ini menggunakan kredensi dari akun yang sama yang membuat aturan.

```
aws route53resolver put-resolver-rule-policy \
  --region us-east-1 \
  --arn "arn:aws:route53resolver:us-east-1:111122223333:resolver-rule/rslvr-
rr-42b60677c0example" \
  --resolver-rule-policy "{ \"Version\" : \"2012-10-17\" , \
    \"Statement\" : [ { \
      \"Effect\" : \"Allow\" , \
      \"Principal\" : { \"AWS\" : \"444455556666\" } , \
      \"Action\" : [ \
        \"route53resolver:GetResolverRule\" , \
```

```

    \"route53resolver:AssociateResolverRule\", \
    \"route53resolver:DisassociateResolverRule\", \
    \"route53resolver:ListResolverRules\", \
    \"route53resolver:ListResolverRuleAssociations\" ], \
    \"Resource\" : [ \"arn:aws:route53resolver:us-east-1:111122223333:resolver-
rule/rslvr-rr-42b60677c0example\" ] } ] }"

```

Output:

```

{
  "ReturnValue": true
}

```

Setelah menjalankan `put-resolver-rule-policy`, Anda dapat menjalankan dua perintah Resource Access Manager (RAM) berikut. Anda harus menggunakan akun yang ingin Anda bagikan aturannya:

`get-resource-share-invitations` mengembalikan nilai `resourceShareInvitationArn`. Anda memerlukan nilai ini untuk menerima undangan untuk menggunakan aturan bersama.  
`accept-resource-share-invitation` menerima undangan untuk menggunakan aturan bersama.

Untuk informasi selengkapnya, lihat dokumentasi berikut ini:

[get-resource-share-invitationsaccept-resource-share-invitations](#) [Berbagi Aturan Penerusan dengan AWS Akun Lain dan Menggunakan Aturan Bersama](#) di Panduan Pengembang Amazon Route 53

- Untuk detail API, lihat [PutResolverRulePolicy](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk mengaitkan tag dengan sumber daya Resolver

`tag-resource` Contoh berikut mengaitkan dua pasangan kunci tag/nilai dengan aturan Resolver yang ditentukan.

```
aws route53resolver tag-resource \
```

```
--resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0example" \
--tags "Key=my-key-1,Value=my-value-1" "Key=my-key-2,Value=my-value-2"
```

Perintah ini tidak menghasilkan output.

Untuk informasi tentang penggunaan tag untuk alokasi biaya, lihat [Menggunakan Tag Alokasi Biaya](#) di Panduan Pengguna AWS Billing and Cost Management.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya Resolver

`untag-resource` Contoh berikut menghapus dua tag dari aturan Resolver yang ditentukan.

```
aws route53resolver untag-resource \
--resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0example" \
--tag-keys my-key-1 my-key-2
```

Perintah ini tidak menghasilkan output. Untuk mengonfirmasi bahwa tag telah dihapus, Anda dapat menggunakan [list-tags-for-resource](#).

Untuk informasi tentang penggunaan tag untuk alokasi biaya, lihat [Menggunakan Tag Alokasi Biaya](#) di Panduan Pengguna AWS Billing and Cost Management.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-firewall-config

Contoh kode berikut menunjukkan cara menggunakan `update-firewall-config`.

### AWS CLI

Untuk memperbaiki konfigurasi firewall

`update-firewall-config` Contoh berikut memperbarui konfigurasi DNS Firewall.

```
aws route53resolver update-firewall-config \  
  --resource-id vpc-31e92222 \  
  --firewall-fail-open DISABLED
```

Output:

```
{  
  "FirewallConfig": {  
    "Id": "rslvr-fc-86016850cexample",  
    "ResourceId": "vpc-31e92222",  
    "OwnerId": "123456789012",  
    "FirewallFailOpen": "DISABLED"  
  }  
}
```

Untuk informasi selengkapnya, lihat konfigurasi [DNS Firewall VPC](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateFirewallConfig](#) di Referensi AWS CLI Perintah.

## update-firewall-domains

Contoh kode berikut menunjukkan cara menggunakan `update-firewall-domains`.

AWS CLI

Untuk memperbarui daftar domain

`update-firewall-domains` Contoh berikut menambahkan domain ke daftar domain dengan ID yang Anda berikan.

```
aws route53resolver update-firewall-domains \  
  --firewall-domain-list-id rslvr-fdl-42b60677cexampleb \  
  --operation ADD \  
  --domains test1.com test2.com test3.com
```

Output:

```
{
```

```
"Id": "rslvr-fdl-42b60677cexample",
"Name": "test",
"Status": "UPDATING",
"StatusMessage": "Updating the Firewall Domain List"
}
```

Untuk informasi selengkapnya, lihat [Mengelola daftar domain Anda sendiri](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateFirewallDomains](#) di Referensi AWS CLI Perintah.

## update-firewall-rule-group-association

Contoh kode berikut menunjukkan cara menggunakan `update-firewall-rule-group-association`.

### AWS CLI

Untuk memperbarui asosiasi grup aturan firewall

`update-firewall-rule-group-association` Contoh berikut memperbarui asosiasi grup aturan firewall.

```
aws route53resolver update-firewall-rule-group-association \
  --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example \
  --priority 103
```

Output:

```
{
  "FirewallRuleGroupAssociation": {
    "Id": "rslvr-frgassoc-57e8873d7example",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-association/rslvr-frgassoc-57e8873d7example",
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
    "VpcId": "vpc-31e92222",
    "Name": "test-association",
    "Priority": 103,
    "MutationProtection": "DISABLED",
    "Status": "UPDATING",
    "StatusMessage": "Updating the Firewall Rule Group Association Attributes",
```



```

    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
    "CreationTime": "2021-05-25T21:47:48.755768Z",
    "ModificationTime": "2021-05-25T21:50:09.272569Z"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi antara VPC Anda dan grup aturan Route 53 Resolver DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateFirewallRuleGroupAssociation](#) di Referensi AWS CLI Perintah.

## update-firewall-rule

Contoh kode berikut menunjukkan cara menggunakan `update-firewall-rule`.

### AWS CLI

Untuk memperbarui aturan firewall

`update-firewall-rule` Contoh berikut memperbarui aturan firewall dengan parameter yang Anda tentukan.

```

aws route53resolver update-firewall-rule \
  --firewall-rule-group-id rslvr-frg-47f93271fexample \
  --firewall-domain-list-id rslvr-fdl-9e956e9ffexample \
  --priority 102

```

Output:

```

{
  "FirewallRule": {
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
    "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
    "Name": "allow-rule",
    "Priority": 102,
    "Action": "ALLOW",
    "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
    "CreationTime": "2021-05-25T21:44:00.346093Z",
    "ModificationTime": "2021-05-25T21:45:59.611600Z"
  }
}

```

Untuk selengkapnya, lihat [Mengelola grup aturan dan aturan di DNS Firewall](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateFirewallRule](#) di Referensi AWS CLI Perintah.

## update-resolver-endpoint

Contoh kode berikut menunjukkan cara menggunakan `update-resolver-endpoint`.

### AWS CLI

Untuk memperbarui nama titik akhir Resolver

`update-resolver-endpoint` Contoh berikut memperbarui nama titik akhir Resolver. Memperbarui nilai lain tidak didukung.

```
aws route53resolver update-resolver-endpoint \  
  --resolver-endpoint-id rslvr-in-b5d45e32bdc445f09 \  
  --name my-renamed-inbound-endpoint
```

Output:

```
{  
  "ResolverEndpoint": {  
    "Id": "rslvr-in-b5d45e32bdexample",  
    "CreatorRequestId": "2020-01-02-18:48",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/  
rslvr-in-b5d45e32bdexample",  
    "Name": "my-renamed-inbound-endpoint",  
    "SecurityGroupIds": [  
      "sg-f62bexam"  
    ],  
    "Direction": "INBOUND",  
    "IpAddressCount": 2,  
    "HostVPCId": "vpc-304bexam",  
    "Status": "OPERATIONAL",  
    "StatusMessage": "This Resolver Endpoint is operational.",  
    "CreationTime": "2020-01-01T18:33:59.265Z",  
    "ModificationTime": "2020-01-08T18:33:59.265Z"  
  }  
}
```

- Untuk detail API, lihat [UpdateResolverEndpoint](#) di Referensi AWS CLI Perintah.

## update-resolver-rule

Contoh kode berikut menunjukkan cara menggunakan `update-resolver-rule`.

### AWS CLI

Contoh 1: Untuk memperbarui pengaturan titik akhir Resolver

`update-resolver-rule` Contoh berikut memperbarui nama aturan, alamat IP di jaringan lokal tempat kueri DNS diteruskan, dan ID titik akhir Resolver keluar yang Anda gunakan untuk meneruskan kueri ke jaringan Anda.

Catatan Nilai yang `TargetIps` ada untuk ditimpa, jadi Anda harus menentukan semua alamat IP yang ingin Anda miliki aturan setelah pembaruan.

```
aws route53resolver update-resolver-rule \  
  --resolver-rule-id rslvr-rr-1247fa64f3example \  
  --config Name="my-2nd-rule",TargetIps=[{Ip=192.0.2.45,Port=53},  
{Ip=192.0.2.46,Port=53}],ResolverEndpointId=rslvr-out-7b89ed0d25example
```

Output:

```
{  
  "ResolverRule": {  
    "Id": "rslvr-rr-1247fa64f3example",  
    "CreatorRequestId": "2020-01-02-18:47",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-  
rr-1247fa64f3example",  
    "DomainName": "www.example.com.",  
    "Status": "COMPLETE",  
    "StatusMessage": "[Trace id: 1-5dcc90b9-8a8ee860aba1ebd89example]  
Successfully updated Resolver Rule.",  
    "RuleType": "FORWARD",  
    "Name": "my-2nd-rule",  
    "TargetIps": [  
      {  
        "Ip": "192.0.2.45",  
        "Port": 53  
      },  
    ],  
  },  
}
```

```

    {
      "Ip": "192.0.2.46",
      "Port": 53
    }
  ],
  "ResolverEndpointId": "rslvr-out-7b89ed0d25example",
  "OwnerId": "111122223333",
  "ShareStatus": "NOT_SHARED"
}
}

```

Contoh 2: Untuk memperbarui pengaturan Titik akhir Resolver menggunakan file untuk pengaturan ``config``

Anda juga dapat menyertakan config pengaturan dalam file JSON dan kemudian menentukan file itu saat Anda menelepon `update-resolver-rule`.

```

aws route53resolver update-resolver-rule \
  --resolver-rule-id rslvr-rr-1247fa64f3example \
  --config file://c:\temp\update-resolver-rule.json

```

Isi dari `update-resolver-rule.json`.

```

{
  "Name": "my-2nd-rule",
  "TargetIps": [
    {
      "Ip": "192.0.2.45",
      "Port": 53
    },
    {
      "Ip": "192.0.2.46",
      "Port": 53
    }
  ],
  "ResolverEndpointId": "rslvr-out-7b89ed0d25example"
}

```

Untuk selengkapnya, lihat [Nilai yang Anda Tentukan Saat Membuat atau Mengedit Aturan](#) di Panduan Pengembang Amazon Route 53.

- Untuk detail API, lihat [UpdateResolverRule](#) di Referensi AWS CLI Perintah.

## Contoh Amazon S3 menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon S3. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **abort-multipart-upload**

Contoh kode berikut menunjukkan cara menggunakan `abort-multipart-upload`.

AWS CLI

Untuk membatalkan unggahan multipart yang ditentukan

`abort-multipart-upload` Perintah berikut membatalkan unggahan multibagian untuk kunci `multipart/01` di bucket `my-bucket`

```
aws s3api abort-multipart-upload \
  --bucket my-bucket \
  --key multipart/01 \
  --upload-id
dfRtDYU0WCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZ1jF.Yxwh6XG7WfS2vC4to6HiV6Yj1x.cph0gtNBtJ8P3URCS
```

ID unggahan yang diperlukan oleh perintah ini adalah output oleh `create-multipart-upload` dan juga dapat diambil dengan `list-multipart-uploads`.

- Untuk detail API, lihat [AbortMultipartUpload](#) di Referensi AWS CLI Perintah.

## complete-multipart-upload

Contoh kode berikut menunjukkan cara menggunakan `complete-multipart-upload`.

### AWS CLI

Perintah berikut menyelesaikan unggahan multipart untuk kunci `multipart/01` di bucket: `my-bucket`

```
aws s3api complete-multipart-upload --multipart-upload file://
mpustruct --bucket my-bucket --key 'multipart/01' --upload-id
dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3URCS
```

ID unggahan yang diperlukan oleh perintah ini adalah output oleh `create-multipart-upload` dan juga dapat diambil dengan `list-multipart-uploads`.

Opsi unggahan multibagian dalam perintah di atas mengambil struktur JSON yang menjelaskan bagian-bagian dari unggahan multibagian yang harus dipasang kembali ke dalam file lengkap. Dalam contoh ini, `file://` awalan digunakan untuk memuat struktur JSON dari file di folder lokal bernama `mpustruct`

`mpustruct`:

```
{
  "Parts": [
    {
      "ETag": "e868e0f4719e394144ef36531ee6824c",
      "PartNumber": 1
    },
    {
      "ETag": "6bb2b12753d66fe86da4998aa33fffb0",
      "PartNumber": 2
    },
    {
      "ETag": "d0a0112e841abec9c9ec83406f0159c8",
      "PartNumber": 3
    }
  ]
}
```

Nilai ETag untuk setiap bagian yang diunggah adalah output setiap kali Anda mengunggah bagian menggunakan `upload-part` perintah dan juga dapat diambil dengan memanggil `list-parts` atau dihitung dengan mengambil checksum MD5 dari setiap bagian.

Output:

```
{
  "ETag": "\"3944a9f7a4faab7f78788ff6210f63f0-3\"",
  "Bucket": "my-bucket",
  "Location": "https://my-bucket.s3.amazonaws.com/multipart%2F01",
  "Key": "multipart/01"
}
```

- Untuk detail API, lihat [CompleteMultipartUpload](#) di Referensi AWS CLI Perintah.

## copy-object

Contoh kode berikut menunjukkan cara menggunakan `copy-object`.

AWS CLI

Perintah berikut menyalin objek dari `bucket-1` ke `bucket-2`:

```
aws s3api copy-object --copy-source bucket-1/test.txt --key test.txt --bucket
bucket-2
```

Output:

```
{
  "CopyObjectResult": {
    "LastModified": "2015-11-10T01:07:25.000Z",
    "ETag": "\"589c8b79c230a6ecd5a7e1d040a9a030\""
  },
  "VersionId": "YdnYvTCVDqRRFA.NFJjy36p0hxifM1kA"
}
```

- Untuk detail API, lihat [CopyObject](#) di Referensi AWS CLI Perintah.

## cp

Contoh kode berikut menunjukkan cara menggunakan `cp`.

## AWS CLI

### Contoh 1: Menyalin file lokal ke S3

cpPerintah berikut menyalin satu file ke bucket dan kunci tertentu:

```
aws s3 cp test.txt s3://mybucket/test2.txt
```

Output:

```
upload: test.txt to s3://mybucket/test2.txt
```

### Contoh 2: Menyalin file lokal ke S3 dengan tanggal kedaluwarsa

cpPerintah berikut menyalin satu file ke bucket dan kunci tertentu yang kedaluwarsa pada stempel waktu ISO 8601 yang ditentukan:

```
aws s3 cp test.txt s3://mybucket/test2.txt \  
--expires 2014-10-01T20:30:00Z
```

Output:

```
upload: test.txt to s3://mybucket/test2.txt
```

### Contoh 3: Menyalin file dari S3 ke S3

cpPerintah berikut menyalin objek s3 tunggal ke bucket dan kunci tertentu:

```
aws s3 cp s3://mybucket/test.txt s3://mybucket/test2.txt
```

Output:

```
copy: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

### Contoh 4: Menyalin objek S3 ke file lokal

cpPerintah berikut menyalin satu objek ke file tertentu secara lokal:

```
aws s3 cp s3://mybucket/test.txt test2.txt
```



**Output:**

```
download: s3://mybucket/test.txt to test2.txt
```

**Contoh 5: Menyalin objek S3 dari satu ember ke ember lainnya**

Perintah berikut menyalin satu objek ke bucket tertentu sambil mempertahankan nama aslinya:

```
aws s3 cp s3://mybucket/test.txt s3://mybucket2/
```

**Output:**

```
copy: s3://mybucket/test.txt to s3://mybucket2/test.txt
```

**Contoh 6: Menyalin objek S3 secara rekursif ke direktori lokal**

Ketika diteruskan dengan parameter `--recursive`, `cp` perintah berikut secara rekursif menyalin semua objek di bawah awalan dan ember yang ditentukan ke direktori tertentu. Dalam contoh ini, ember `mybucket` memiliki objek `test1.txt` dan `test2.txt`:

```
aws s3 cp s3://mybucket . \  
--recursive
```

**Output:**

```
download: s3://mybucket/test1.txt to test1.txt  
download: s3://mybucket/test2.txt to test2.txt
```

**Contoh 7: Menyalin file lokal secara rekursif ke S3**

Ketika diteruskan dengan parameter `--recursive`, `cp` perintah berikut secara rekursif menyalin semua file di bawah direktori tertentu ke bucket dan awalan tertentu sementara mengecualikan beberapa file dengan menggunakan parameter `--exclude`. Dalam contoh ini, direktori `myDir` memiliki file `test1.txt` dan `test2.jpg`:

```
aws s3 cp myDir s3://mybucket/ \  
--recursive \  
--exclude "*.jpg"
```

**Output:**

```
upload: myDir/test1.txt to s3://mybucket/test1.txt
```

**Contoh 8: Menyalin objek S3 secara rekursif ke ember lain**

Ketika diteruskan dengan parameter `--recursive`, `cp` perintah berikut secara rekursif menyalin semua objek di bawah bucket tertentu ke bucket lain sambil mengecualikan beberapa objek dengan menggunakan parameter. `--exclude` Dalam contoh ini, ember `mybucket` memiliki objek `test1.txt` dan `another/test1.txt`:

```
aws s3 cp s3://mybucket/ s3://mybucket2/ \  
  --recursive \  
  --exclude "another/*"
```

**Output:**

```
copy: s3://mybucket/test1.txt to s3://mybucket2/test1.txt
```

Anda dapat menggabungkan `--exclude` dan `--include` opsi untuk menyalin hanya objek yang cocok dengan pola, tidak termasuk semua yang lain:

```
aws s3 cp s3://mybucket/logs/ s3://mybucket2/logs/ \  
  --recursive \  
  --exclude "*" \  
  --include "*.log"
```

**Output:**

```
copy: s3://mybucket/logs/test/test.log to s3://mybucket2/logs/test/test.log  
copy: s3://mybucket/logs/test3.log to s3://mybucket2/logs/test3.log
```

**Contoh 9: Mengatur Daftar Kontrol Akses (ACL) saat menyalin objek S3**

`cp` Perintah berikut menyalin satu objek ke bucket dan kunci tertentu saat menyetel ACL ke `public-read-write`:

```
aws s3 cp s3://mybucket/test.txt s3://mybucket/test2.txt \  
  --acl public-read-write
```

```
--acl public-read-write
```

Output:

```
copy: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

Perhatikan bahwa jika Anda menggunakan `--acl` opsi, pastikan bahwa setiap kebijakan IAM terkait menyertakan `"s3:PutObjectAcl"` tindakan:

```
aws iam get-user-policy \  
  --user-name myuser \  
  --policy-name mypolicy
```

Output:

```
{  
  "UserName": "myuser",  
  "PolicyName": "mypolicy",  
  "PolicyDocument": {  
    "Version": "2012-10-17",  
    "Statement": [  
      {  
        "Action": [  
          "s3:PutObject",  
          "s3:PutObjectAcl"  
        ],  
        "Resource": [  
          "arn:aws:s3:::mybucket/*"  
        ],  
        "Effect": "Allow",  
        "Sid": "Stmt1234567891234"  
      }  
    ]  
  }  
}
```

Contoh 10: Memberikan izin untuk objek S3

Perintah berikut menggambarkan penggunaan `--grants` opsi untuk memberikan akses baca ke semua pengguna yang diidentifikasi oleh URI dan kontrol penuh untuk pengguna tertentu yang diidentifikasi oleh ID Canonical mereka:

```
aws s3 cp file.txt s3://mybucket/ --grants read=uri=http://acs.amazonaws.com/groups/global/AllUsers
full=id=79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

Output:

```
upload: file.txt to s3://mybucket/file.txt
```

Contoh 11: Mengunggah aliran file lokal ke S3

PowerShell dapat mengubah pengkodean atau menambahkan CRLF ke input pipa.

cpPerintah berikut mengunggah aliran file lokal dari input standar ke bucket dan kunci tertentu:

```
aws s3 cp - s3://mybucket/stream.txt
```

Contoh 12: Mengunggah aliran file lokal yang lebih besar dari 50GB ke S3

cpPerintah berikut mengunggah aliran file lokal 51GB dari input standar ke bucket dan kunci tertentu. `--expected-size`Opsi harus disediakan, atau unggahan mungkin gagal ketika mencapai batas bagian default 10.000:

```
aws s3 cp - s3://mybucket/stream.txt --expected-size 54760833024
```

Contoh 13: Mengunduh objek S3 sebagai aliran file lokal

PowerShell dapat mengubah pengkodean atau menambahkan CRLF ke output pipa atau dialihkan.

cpPerintah berikut mengunduh objek S3 secara lokal sebagai aliran ke output standar. Mengunduh sebagai aliran saat ini tidak kompatibel dengan `--recursive` parameter:

```
aws s3 cp s3://mybucket/stream.txt -
```

Contoh 14: Mengunggah ke titik akses S3

cpPerintah berikut mengunggah satu file (`mydoc.txt`) ke titik akses (`myaccesspoint`) di key (`mykey`):

```
aws s3 cp mydoc.txt s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/mykey
```

**Output:**

```
upload: mydoc.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/mykey
```

**Contoh 15: Mengunduh dari titik akses S3**

cpPerintah berikut mengunduh satu objek (mykey) dari titik akses (myaccesspoint) ke file lokal (mydoc.txt):

```
aws s3 cp s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
mydoc.txt
```

**Output:**

```
download: s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey to
mydoc.txt
```

- Untuk detail API, lihat [Cp](#) di Referensi AWS CLI Perintah.

**create-bucket**

Contoh kode berikut menunjukkan cara menggunakancreate-bucket.

**AWS CLI**

Contoh 1: Untuk membuat ember

create-bucketContoh berikut membuat bucket bernama my-bucket:

```
aws s3api create-bucket \
  --bucket my-bucket \
  --region us-east-1
```

**Output:**

```
{
  "Location": "/my-bucket"
}
```

Lihat informasi yang lebih lengkap di [Membuat bucket](#) dalam Panduan Pengguna Amazon S3.

Contoh 2: Untuk membuat ember dengan pemilik diberlakukan

`create-bucket` Contoh berikut membuat bucket bernama yang menggunakan setelan `my-bucket` yang diterapkan pemilik bucket untuk Kepemilikan Objek S3.

```
aws s3api create-bucket \
  --bucket my-bucket \
  --region us-east-1 \
  --object-ownership BucketOwnerEnforced
```

Output:

```
{
  "Location": "/my-bucket"
}
```

Untuk informasi selengkapnya, lihat [Mengontrol kepemilikan objek dan menonaktifkan ACL di Panduan Pengguna Amazon S3](#).

Contoh 3: Untuk membuat bucket di luar wilayah ``us-east-1``

`create-bucket` Contoh berikut membuat bucket bernama `my-bucket` di `eu-west-1` wilayah tersebut. Daerah di luar `us-east-1` memerlukan yang sesuai `LocationConstraint` untuk ditentukan untuk membuat ember di wilayah yang diinginkan.

```
aws s3api create-bucket \
  --bucket my-bucket \
  --region eu-west-1 \
  --create-bucket-configuration LocationConstraint=eu-west-1
```

Output:

```
{
```

```
"Location": "http://my-bucket.s3.amazonaws.com/"
}
```

Lihat informasi yang lebih lengkap di [Membuat bucket](#) dalam Panduan Pengguna Amazon S3.

- Untuk detail API, lihat [CreateBucket](#) di Referensi AWS CLI Perintah.

## **create-multipart-upload**

Contoh kode berikut menunjukkan cara menggunakan `create-multipart-upload`.

### AWS CLI

Perintah berikut membuat unggahan multipart di bucket `my-bucket` dengan kunci `multipart/01`:

```
aws s3api create-multipart-upload --bucket my-bucket --key 'multipart/01'
```

Output:

```
{
  "Bucket": "my-bucket",
  "UploadId":
  "dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZ1jF.Yxwh6XG7WfS2vC4to6HiV6Yj1x.cph0gtNBtJ8P3URC
  "Key": "multipart/01"
}
```

File yang sudah selesai akan diberi nama `01` dalam folder yang disebut `multipart` dalam ember `my-bucket`. Simpan ID unggahan, kunci, dan nama bucket untuk digunakan dengan `upload-part` perintah.

- Untuk detail API, lihat [CreateMultipartUpload](#) di Referensi AWS CLI Perintah.

## **delete-bucket-analytics-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-analytics-configuration`.

### AWS CLI

Untuk menghapus konfigurasi analitik untuk bucket

`delete-bucket-analytics-configuration` Contoh berikut menghapus konfigurasi analitik untuk bucket dan ID yang ditentukan.

```
aws s3api delete-bucket-analytics-configuration \  
  --bucket my-bucket \  
  --id 1
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBucketAnalyticsConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-bucket-cors**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-cors`.

### AWS CLI

Perintah berikut menghapus konfigurasi Cross-Origin Resource Sharing dari bucket bernama: `my-bucket`

```
aws s3api delete-bucket-cors --bucket my-bucket
```

- Untuk detail API, lihat [DeleteBucketCors](#) di Referensi AWS CLI Perintah.

## **delete-bucket-encryption**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-encryption`.

### AWS CLI

Untuk menghapus konfigurasi enkripsi sisi server dari bucket

`delete-bucket-encryption` Contoh berikut menghapus konfigurasi enkripsi sisi server dari bucket yang ditentukan.

```
aws s3api delete-bucket-encryption \  
  --bucket my-bucket
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBucketEncryption](#) di Referensi AWS CLI Perintah.



## delete-bucket-intelligent-tiering-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-intelligent-tiering-configuration`.

### AWS CLI

Untuk menghapus konfigurasi S3 Intelligent-Tiering pada bucket

`delete-bucket-intelligent-tiering-configuration` Contoh berikut menghapus konfigurasi S3 Intelligent-Tiering, bernama `ExampleConfig`, pada bucket.

```
aws s3api delete-bucket-intelligent-tiering-configuration \  
  --bucket DOC-EXAMPLE-BUCKET \  
  --id ExampleConfig
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menggunakan S3 Intelligent-Tiering di Panduan Pengguna Amazon S3](#).

- Untuk detail API, lihat [DeleteBucketIntelligentTieringConfiguration](#) di Referensi AWS CLI Perintah.

## delete-bucket-inventory-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-inventory-configuration`.

### AWS CLI

Untuk menghapus konfigurasi inventaris bucket

`delete-bucket-inventory-configuration` Contoh berikut menghapus konfigurasi inventaris dengan ID 1 untuk bucket yang ditentukan.

```
aws s3api delete-bucket-inventory-configuration \  
  --bucket my-bucket \  
  --id 1
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBucketInventoryConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-bucket-lifecycle**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-lifecycle`.

### AWS CLI

Perintah berikut menghapus konfigurasi siklus hidup dari bucket bernama: `my-bucket`

```
aws s3api delete-bucket-lifecycle --bucket my-bucket
```

- Untuk detail API, lihat [DeleteBucketLifecycle](#) di Referensi AWS CLI Perintah.

## **delete-bucket-metrics-configuration**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-metrics-configuration`.

### AWS CLI

Untuk menghapus konfigurasi metrik untuk bucket

`delete-bucket-metrics-configuration` Contoh berikut menghapus konfigurasi metrik untuk bucket dan ID yang ditentukan.

```
aws s3api delete-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteBucketMetricsConfiguration](#) di Referensi AWS CLI Perintah.

## **delete-bucket-ownership-controls**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-ownership-controls`.

### AWS CLI

Untuk menghapus pengaturan kepemilikan bucket dari bucket

`delete-bucket-ownership-controls` Contoh berikut menghapus pengaturan kepemilikan bucket dari bucket.

```
aws s3api delete-bucket-ownership-controls \  
  --bucket DOC-EXAMPLE-BUCKET
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyetel Kepemilikan Objek pada bucket yang ada](#) di Panduan Pengguna Amazon S3.

- Untuk detail API, lihat [DeleteBucketOwnershipControls](#) di Referensi AWS CLI Perintah.

## **delete-bucket-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-policy`.

AWS CLI

Perintah berikut menghapus kebijakan bucket dari bucket bernama `my-bucket`:

```
aws s3api delete-bucket-policy --bucket my-bucket
```

- Untuk detail API, lihat [DeleteBucketPolicy](#) di Referensi AWS CLI Perintah.

## **delete-bucket-replication**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-replication`.

AWS CLI

Perintah berikut menghapus konfigurasi replikasi dari bucket bernama: `my-bucket`

```
aws s3api delete-bucket-replication --bucket my-bucket
```

- Untuk detail API, lihat [DeleteBucketReplication](#) di Referensi AWS CLI Perintah.

## **delete-bucket-tagging**

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-tagging`.

## AWS CLI

Perintah berikut menghapus konfigurasi penandaan dari bucket bernama: my-bucket

```
aws s3api delete-bucket-tagging --bucket my-bucket
```

- Untuk detail API, lihat [DeleteBucketTagging](#) di Referensi AWS CLI Perintah.

## delete-bucket-website

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket-website`.

## AWS CLI

Perintah berikut menghapus konfigurasi situs web dari bucket bernama my-bucket:

```
aws s3api delete-bucket-website --bucket my-bucket
```

- Untuk detail API, lihat [DeleteBucketWebsite](#) di Referensi AWS CLI Perintah.

## delete-bucket

Contoh kode berikut menunjukkan cara menggunakan `delete-bucket`.

## AWS CLI

Perintah berikut menghapus bucket bernama my-bucket:

```
aws s3api delete-bucket --bucket my-bucket --region us-east-1
```

- Untuk detail API, lihat [DeleteBucket](#) di Referensi AWS CLI Perintah.

## delete-object-tagging

Contoh kode berikut menunjukkan cara menggunakan `delete-object-tagging`.

## AWS CLI

Untuk menghapus set tag dari suatu objek

`delete-object-tagging` Contoh berikut menghapus tag dengan kunci yang ditentukan dari objek `doc1.rtf`.

```
aws s3api delete-object-tagging \  
  --bucket my-bucket \  
  --key doc1.rtf
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteObjectTagging](#) di Referensi AWS CLI Perintah.

## **delete-object**

Contoh kode berikut menunjukkan cara menggunakan `delete-object`.

### AWS CLI

Perintah berikut menghapus objek bernama `test.txt` dari bucket bernama `my-bucket`:

```
aws s3api delete-object --bucket my-bucket --key test.txt
```

Jika pembuatan versi bucket diaktifkan, output akan berisi ID versi penanda hapus:

```
{  
  "VersionId": "9_gKg5vG56F.TTEUdwkxGpJ3tND1W1Gq",  
  "DeleteMarker": true  
}
```

Untuk informasi selengkapnya tentang menghapus objek, lihat [Menghapus Objek](#) di Panduan Pengembang Amazon S3.

- Untuk detail API, lihat [DeleteObject](#) di Referensi AWS CLI Perintah.

## **delete-objects**

Contoh kode berikut menunjukkan cara menggunakan `delete-objects`.

### AWS CLI

Perintah berikut menghapus objek dari bucket bernama `my-bucket`:

```
aws s3api delete-objects --bucket my-bucket --delete file://delete.json
```

`delete.json` adalah dokumen JSON di direktori saat ini yang menentukan objek yang akan dihapus:

```
{
  "Objects": [
    {
      "Key": "test1.txt"
    }
  ],
  "Quiet": false
}
```

Output:

```
{
  "Deleted": [
    {
      "DeleteMarkerVersionId": "mYAT5Mc6F7aeUL8SS7FAAqUP01koHwzU",
      "Key": "test1.txt",
      "DeleteMarker": true
    }
  ]
}
```

- Untuk detail API, lihat [DeleteObjects](#) di Referensi AWS CLI Perintah.

## **delete-public-access-block**

Contoh kode berikut menunjukkan cara menggunakan `delete-public-access-block`.

### AWS CLI

Untuk menghapus konfigurasi blokir akses publik untuk bucket

`delete-public-access-block` Contoh berikut menghapus konfigurasi blokir akses publik pada bucket yang ditentukan.

```
aws s3api delete-public-access-block \
  --bucket my-bucket
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePublicAccessBlock](#) di Referensi AWS CLI Perintah.

## get-bucket-accelerate-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-accelerate-configuration`.

### AWS CLI

Untuk mengambil konfigurasi percepatan bucket

`get-bucket-accelerate-configuration` Contoh berikut mengambil konfigurasi percepatan untuk bucket yang ditentukan.

```
aws s3api get-bucket-accelerate-configuration \  
  --bucket my-bucket
```

Output:

```
{  
  "Status": "Enabled"  
}
```

- Untuk detail API, lihat [GetBucketAccelerateConfiguration](#) di Referensi AWS CLI Perintah.

## get-bucket-acl

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-acl`.

### AWS CLI

Perintah berikut mengambil daftar kontrol akses untuk bucket bernama `my-bucket`:

```
aws s3api get-bucket-acl --bucket my-bucket
```

Output:

```
{  
  "Owner": {
```

```

    "DisplayName": "my-username",
    "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
  },
  "Grants": [
    {
      "Grantee": {
        "DisplayName": "my-username",
        "ID":
"7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
      },
      "Permission": "FULL_CONTROL"
    }
  ]
}

```

- Untuk detail API, lihat [GetBucketAcl](#) di Referensi AWS CLI Perintah.

## get-bucket-analytics-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-analytics-configuration`.

### AWS CLI

Untuk mengambil konfigurasi analitik untuk bucket dengan ID tertentu

`get-bucket-analytics-configuration` Contoh berikut menampilkan konfigurasi analitik untuk bucket dan ID yang ditentukan.

```

aws s3api get-bucket-analytics-configuration \
  --bucket my-bucket \
  --id 1

```

Output:

```

{
  "AnalyticsConfiguration": {
    "StorageClassAnalysis": {},
    "Id": "1"
  }
}

```

- Untuk detail API, lihat [GetBucketAnalyticsConfiguration](#) di Referensi AWS CLI Perintah.



## get-bucket-cors

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-cors`.

### AWS CLI

Perintah berikut mengambil konfigurasi Cross-Origin Resource Sharing untuk bucket bernama: `my-bucket`

```
aws s3api get-bucket-cors --bucket my-bucket
```

### Output:

```
{
  "CORSRules": [
    {
      "AllowedHeaders": [
        "*"
      ],
      "ExposeHeaders": [
        "x-amz-server-side-encryption"
      ],
      "AllowedMethods": [
        "PUT",
        "POST",
        "DELETE"
      ],
      "MaxAgeSeconds": 3000,
      "AllowedOrigins": [
        "http://www.example.com"
      ]
    },
    {
      "AllowedHeaders": [
        "Authorization"
      ],
      "MaxAgeSeconds": 3000,
      "AllowedMethods": [
        "GET"
      ],
      "AllowedOrigins": [
        "*"
      ]
    }
  ]
}
```

```

    }
  ]
}

```

- Untuk detail API, lihat [GetBucketCors](#) di Referensi AWS CLI Perintah.

## get-bucket-encryption

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-encryption`.

### AWS CLI

Untuk mengambil konfigurasi enkripsi sisi server untuk bucket

`get-bucket-encryption` Contoh berikut mengambil konfigurasi enkripsi sisi server untuk bucket. `my-bucket`

```

aws s3api get-bucket-encryption \
  --bucket my-bucket

```

Output:

```

{
  "ServerSideEncryptionConfiguration": {
    "Rules": [
      {
        "ApplyServerSideEncryptionByDefault": {
          "SSEAlgorithm": "AES256"
        }
      }
    ]
  }
}

```

- Untuk detail API, lihat [GetBucketEncryption](#) di Referensi AWS CLI Perintah.

## get-bucket-intelligent-tiering-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-intelligent-tiering-configuration`.

## AWS CLI

Untuk mengambil konfigurasi S3 Intelligent-Tiering pada bucket

`get-bucket-intelligent-tiering-configuration` Contoh berikut mengambil konfigurasi S3 Intelligent-Tiering, bernama, pada bucket. ExampleConfig

```
aws s3api get-bucket-intelligent-tiering-configuration \
  --bucket DOC-EXAMPLE-BUCKET \
  --id ExampleConfig
```

Output:

```
{
  "IntelligentTieringConfiguration": {
    "Id": "ExampleConfig2",
    "Filter": {
      "Prefix": "images"
    },
    "Status": "Enabled",
    "Tierings": [
      {
        "Days": 90,
        "AccessTier": "ARCHIVE_ACCESS"
      },
      {
        "Days": 180,
        "AccessTier": "DEEP_ARCHIVE_ACCESS"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Menggunakan S3 Intelligent-Tiering di Panduan Pengguna Amazon S3](#).

- Untuk detail API, lihat [GetBucketIntelligentTieringConfiguration](#) di Referensi AWS CLI Perintah.

### **get-bucket-inventory-configuration**

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-inventory-configuration`.

## AWS CLI

Untuk mengambil konfigurasi inventaris untuk bucket

`get-bucket-inventory-configuration` Contoh berikut mengambil konfigurasi inventaris untuk bucket yang ditentukan dengan ID1.

```
aws s3api get-bucket-inventory-configuration \  
  --bucket my-bucket \  
  --id 1
```

Output:

```
{  
  "InventoryConfiguration": {  
    "IsEnabled": true,  
    "Destination": {  
      "S3BucketDestination": {  
        "Format": "ORC",  
        "Bucket": "arn:aws:s3:::my-bucket",  
        "AccountId": "123456789012"  
      }  
    },  
    "IncludedObjectVersions": "Current",  
    "Id": "1",  
    "Schedule": {  
      "Frequency": "Weekly"  
    }  
  }  
}
```

- Untuk detail API, lihat [GetBucketInventoryConfiguration](#) di Referensi AWS CLI Perintah.

## **get-bucket-lifecycle-configuration**

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-lifecycle-configuration`.

## AWS CLI

Perintah berikut mengambil konfigurasi siklus hidup untuk bucket bernama: `my-bucket`

```
aws s3api get-bucket-lifecycle-configuration --bucket my-bucket
```

**Output:**

```
{
  "Rules": [
    {
      "ID": "Move rotated logs to Glacier",
      "Prefix": "rotated/",
      "Status": "Enabled",
      "Transitions": [
        {
          "Date": "2015-11-10T00:00:00.000Z",
          "StorageClass": "GLACIER"
        }
      ]
    },
    {
      "Status": "Enabled",
      "Prefix": "",
      "NoncurrentVersionTransitions": [
        {
          "NoncurrentDays": 0,
          "StorageClass": "GLACIER"
        }
      ],
      "ID": "Move old versions to Glacier"
    }
  ]
}
```

- Untuk detail API, lihat [GetBucketLifecycleConfiguration](#) di Referensi AWS CLI Perintah.

**get-bucket-lifecycle**

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-lifecycle`.

**AWS CLI**

Perintah berikut mengambil konfigurasi siklus hidup untuk bucket bernama: `my-bucket`

```
aws s3api get-bucket-lifecycle --bucket my-bucket
```

**Output:**

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (objects in logs/2015/)",
      "Prefix": "logs/2015/",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    },
    {
      "Expiration": {
        "Date": "2016-01-01T00:00:00.000Z"
      },
      "ID": "Delete 2014 logs in 2016.",
      "Prefix": "logs/2014/",
      "Status": "Enabled"
    }
  ]
}
```

- Untuk detail API, lihat [GetBucketLifecycle](#) di Referensi AWS CLI Perintah.

## get-bucket-location

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-location`.

### AWS CLI

Perintah berikut mengambil batasan lokasi untuk bucket bernama `my-bucket`, jika ada kendala:

```
aws s3api get-bucket-location --bucket my-bucket
```

Output:

```
{
  "LocationConstraint": "us-west-2"
}
```

- Untuk detail API, lihat [GetBucketLocation](#) di Referensi AWS CLI Perintah.

## get-bucket-logging

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-logging`.

### AWS CLI

Untuk mengambil status logging untuk bucket

`get-bucket-logging` Contoh berikut mengambil status logging untuk bucket yang ditentukan.

```
aws s3api get-bucket-logging \  
  --bucket my-bucket
```

Output:

```
{  
  "LoggingEnabled": {  
    "TargetPrefix": "",  
    "TargetBucket": "my-bucket-logs"  
  }  
}
```

- Untuk detail API, lihat [GetBucketLogging](#) di Referensi AWS CLI Perintah.

## get-bucket-metrics-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-metrics-configuration`.

### AWS CLI

Untuk mengambil konfigurasi metrik untuk bucket dengan ID tertentu

`get-bucket-metrics-configuration` Contoh berikut menampilkan konfigurasi metrik untuk bucket dan ID yang ditentukan.

```
aws s3api get-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Output:

```
{
```

```
"MetricsConfiguration": {
  "Filter": {
    "Prefix": "logs"
  },
  "Id": "123"
}
```

- Untuk detail API, lihat [GetBucketMetricsConfiguration](#) di Referensi AWS CLI Perintah.

## get-bucket-notification-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-notification-configuration`.

### AWS CLI

Perintah berikut mengambil konfigurasi notifikasi untuk bucket bernama `my-bucket`:

```
aws s3api get-bucket-notification-configuration --bucket my-bucket
```

Output:

```
{
  "TopicConfigurations": [
    {
      "Id": "YmQzMmEwM2EjZWVlI0NGItNzVtZjI1MC00ZjgyLWZDBiZWw1",
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-notification-topic",
      "Events": [
        "s3:ObjectCreated:*"
      ]
    }
  ]
}
```

- Untuk detail API, lihat [GetBucketNotificationConfiguration](#) di Referensi AWS CLI Perintah.

## get-bucket-notification

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-notification`.



## AWS CLI

Perintah berikut mengambil konfigurasi notifikasi untuk bucket bernama my-bucket:

```
aws s3api get-bucket-notification --bucket my-bucket
```

Output:

```
{
  "TopicConfiguration": {
    "Topic": "arn:aws:sns:us-west-2:123456789012:my-notification-topic",
    "Id": "YmQzMmEwM2EjZWVlI0NGItNzVtZjI1MC00ZjgyLWZDBiZWw1",
    "Event": "s3:ObjectCreated:*",
    "Events": [
      "s3:ObjectCreated:*"
    ]
  }
}
```

- Untuk detail API, lihat [GetBucketNotification](#) di Referensi AWS CLI Perintah.

## get-bucket-ownership-controls

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-ownership-controls`.

## AWS CLI

Untuk mengambil pengaturan kepemilikan bucket dari bucket

`get-bucket-ownership-controls` Contoh berikut mengambil pengaturan kepemilikan bucket dari bucket.

```
aws s3api get-bucket-ownership-controls \
  --bucket DOC-EXAMPLE-BUCKET
```

Output:

```
{
  "OwnershipControls": {
    "Rules": [
      {
```

```

    "ObjectOwnership": "BucketOwnerEnforced"
  }
]
}
}

```

Untuk informasi selengkapnya, lihat [Melihat setelan Kepemilikan Objek untuk bucket S3](#) di Panduan Pengguna Amazon S3.

- Untuk detail API, lihat [GetBucketOwnershipControls](#) di Referensi AWS CLI Perintah.

## get-bucket-policy-status

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-policy-status`.

### AWS CLI

Untuk mengambil status kebijakan untuk bucket yang menunjukkan apakah bucket bersifat publik

`get-bucket-policy-status` Contoh berikut mengambil status kebijakan untuk bucket `my-bucket`.

```
aws s3api get-bucket-policy-status \
  --bucket my-bucket
```

Output:

```
{
  "PolicyStatus": {
    "IsPublic": false
  }
}
```

- Untuk detail API, lihat [GetBucketPolicyStatus](#) di Referensi AWS CLI Perintah.

## get-bucket-policy

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-policy`.

### AWS CLI

Perintah berikut mengambil kebijakan bucket untuk bucket bernama `my-bucket`:

```
aws s3api get-bucket-policy --bucket my-bucket
```

Output:

```
{
  "Policy": "{\"Version\":\"2008-10-17\",\"Statement\": [{\"Sid\":\"\",\"Effect\": \"Allow\",\"Principal\":\"*\",\"Action\":\"s3:GetObject\",\"Resource\": \"arn:aws:s3:::my-bucket/*\"}, {\"Sid\":\"\",\"Effect\":\"Deny\",\"Principal\":\"*\", \"Action\":\"s3:GetObject\",\"Resource\":\"arn:aws:s3:::my-bucket/secret/*\"}]}"
}
```

Dapatkan dan letakkan kebijakan bucket Contoh berikut menunjukkan bagaimana Anda dapat mengunduh kebijakan bucket Amazon S3, membuat modifikasi pada file, dan kemudian put-bucket-policy menggunakannya untuk menerapkan kebijakan bucket yang dimodifikasi. Untuk mengunduh kebijakan bucket ke file, Anda dapat menjalankan:

```
aws s3api get-bucket-policy --bucket mybucket --query Policy --output text > policy.json
```

Anda kemudian dapat memodifikasi `policy.json` file sesuai kebutuhan. Terakhir, Anda dapat menerapkan kebijakan yang dimodifikasi ini kembali ke bucket S3 dengan menjalankan:

`policy.json` berkas sesuai kebutuhan. Terakhir, Anda dapat menerapkan kebijakan yang dimodifikasi ini kembali ke bucket S3 dengan menjalankan:

berkas sesuai kebutuhan. Terakhir, Anda dapat menerapkan kebijakan yang dimodifikasi ini kembali ke bucket S3 dengan menjalankan:

```
aws s3api put-bucket-policy --bucket mybucket --policy file://policy.json
```

- Untuk detail API, lihat [GetBucketPolicy](#) di Referensi AWS CLI Perintah.

## get-bucket-replication

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-replication`.

### AWS CLI

Perintah berikut mengambil konfigurasi replikasi untuk bucket bernama: `my-bucket`

```
aws s3api get-bucket-replication --bucket my-bucket
```

Output:

```
{
  "ReplicationConfiguration": {
    "Rules": [
      {
        "Status": "Enabled",
        "Prefix": "",
        "Destination": {
          "Bucket": "arn:aws:s3:::my-bucket-backup",
          "StorageClass": "STANDARD"
        },
        "ID": "ZmUwNzE4ZmQ4tMjVhOS00MTlkLOGI4NDkzZTIWJjNTUtYTA1"
      }
    ],
    "Role": "arn:aws:iam::123456789012:role/s3-replication-role"
  }
}
```

- Untuk detail API, lihat [GetBucketReplication](#) di Referensi AWS CLI Perintah.

## get-bucket-request-payment

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-request-payment`.

AWS CLI

Untuk mengambil konfigurasi pembayaran permintaan untuk bucket

`get-bucket-request-payment` Contoh berikut mengambil konfigurasi requester pay untuk bucket yang ditentukan.

```
aws s3api get-bucket-request-payment \
  --bucket my-bucket
```

Output:

```
{
  "Payer": "BucketOwner"
}
```

```
}
```

- Untuk detail API, lihat [GetBucketRequestPayment](#) di Referensi AWS CLI Perintah.

## get-bucket-tagging

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-tagging`.

### AWS CLI

Perintah berikut mengambil konfigurasi penandaan untuk bucket bernama: `my-bucket`

```
aws s3api get-bucket-tagging --bucket my-bucket
```

Output:

```
{
  "TagSet": [
    {
      "Value": "marketing",
      "Key": "organization"
    }
  ]
}
```

- Untuk detail API, lihat [GetBucketTagging](#) di Referensi AWS CLI Perintah.

## get-bucket-versioning

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-versioning`.

### AWS CLI

Perintah berikut mengambil konfigurasi pembuatan versi untuk bucket bernama: `my-bucket`

```
aws s3api get-bucket-versioning --bucket my-bucket
```

Output:

```
{
```

```
"Status": "Enabled"
}
```

- Untuk detail API, lihat [GetBucketVersioning](#) di Referensi AWS CLI Perintah.

## get-bucket-website

Contoh kode berikut menunjukkan cara menggunakan `get-bucket-website`.

### AWS CLI

Perintah berikut mengambil konfigurasi situs web statis untuk bucket bernama `my-bucket`:

```
aws s3api get-bucket-website --bucket my-bucket
```

Output:

```
{
  "IndexDocument": {
    "Suffix": "index.html"
  },
  "ErrorDocument": {
    "Key": "error.html"
  }
}
```

- Untuk detail API, lihat [GetBucketWebsite](#) di Referensi AWS CLI Perintah.

## get-object-acl

Contoh kode berikut menunjukkan cara menggunakan `get-object-acl`.

### AWS CLI

Perintah berikut mengambil daftar kontrol akses untuk objek dalam bucket bernama `my-bucket`:

```
aws s3api get-object-acl --bucket my-bucket --key index.html
```

Output:

```
{
  "Owner": {
    "DisplayName": "my-username",
    "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
  },
  "Grants": [
    {
      "Grantee": {
        "DisplayName": "my-username",
        "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
      },
      "Permission": "FULL_CONTROL"
    },
    {
      "Grantee": {
        "URI": "http://acs.amazonaws.com/groups/global/AllUsers"
      },
      "Permission": "READ"
    }
  ]
}
```

- Untuk detail API, lihat [GetObjectAcl](#) di Referensi AWS CLI Perintah.

## get-object-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-object-attributes`.

### AWS CLI

Untuk mengambil metadata dari objek tanpa mengembalikan objek itu sendiri

`get-object-attributes` Contoh berikut mengambil metadata dari objek. `doc1.rtf`

```
aws s3api get-object-attributes \
  --bucket my-bucket \
  --key doc1.rtf \
  --object-attributes "StorageClass" "ETag" "ObjectSize"
```

Output:

```
{
  "LastModified": "2022-03-15T19:37:31+00:00",
  "VersionId": "IuCPjXTDzHNf1dAuitVBIKJpF2p1fg4P",
  "ETag": "b662d79adeb7c8d787ea7eafb9ef6207",
  "StorageClass": "STANDARD",
  "ObjectSize": 405
}
```

Untuk informasi selengkapnya, lihat [GetObjectAttributes](#) di Referensi API Amazon S3.

- Untuk detail API, lihat [GetObjectAttributes](#) di Referensi AWS CLI Perintah.

## get-object-legal-hold

Contoh kode berikut menunjukkan cara menggunakan `get-object-legal-hold`.

### AWS CLI

Mengambil status Legal Hold dari suatu objek

`get-object-legal-hold` Contoh berikut mengambil status Penahanan Hukum untuk objek yang ditentukan.

```
aws s3api get-object-legal-hold \
  --bucket my-bucket-with-object-lock \
  --key doc1.rtf
```

Output:

```
{
  "LegalHold": {
    "Status": "ON"
  }
}
```

- Untuk detail API, lihat [GetObjectLegalHold](#) di Referensi AWS CLI Perintah.

## get-object-lock-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-object-lock-configuration`.



## AWS CLI

Untuk mengambil konfigurasi kunci objek untuk bucket

`get-object-lock-configuration` Contoh berikut mengambil konfigurasi kunci objek untuk bucket yang ditentukan.

```
aws s3api get-object-lock-configuration \  
  --bucket my-bucket-with-object-lock
```

Output:

```
{  
  "ObjectLockConfiguration": {  
    "ObjectLockEnabled": "Enabled",  
    "Rule": {  
      "DefaultRetention": {  
        "Mode": "COMPLIANCE",  
        "Days": 50  
      }  
    }  
  }  
}
```

- Untuk detail API, lihat [GetObjectLockConfiguration](#) di Referensi AWS CLI Perintah.

## **get-object-retention**

Contoh kode berikut menunjukkan cara menggunakan `get-object-retention`.

## AWS CLI

Untuk mengambil konfigurasi retensi objek untuk objek

`get-object-retention` Contoh berikut mengambil konfigurasi retensi objek untuk objek tertentu.

```
aws s3api get-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf
```

**Output:**

```
{
  "Retention": {
    "Mode": "GOVERNANCE",
    "RetainUntilDate": "2025-01-01T00:00:00.000Z"
  }
}
```

- Untuk detail API, lihat [GetObjectRetention](#) di Referensi AWS CLI Perintah.

**get-object-tagging**

Contoh kode berikut menunjukkan cara menggunakan `get-object-tagging`.

**AWS CLI**

Untuk mengambil tag yang dilampirkan ke objek

`get-object-tagging` Contoh berikut mengambil nilai-nilai untuk kunci tertentu dari objek tertentu.

```
aws s3api get-object-tagging \
  --bucket my-bucket \
  --key doc1.rtf
```

**Output:**

```
{
  "TagSet": [
    {
      "Value": "confidential",
      "Key": "designation"
    }
  ]
}
```

`get-object-tagging` Contoh berikut mencoba untuk mengambil set tag objek `doc2.rtf`, yang tidak memiliki tag.

```
aws s3api get-object-tagging \
```

```
--bucket my-bucket \  
--key doc2.rtf
```

Output:

```
{  
  "TagSet": []  
}
```

`get-object-tagging` Contoh berikut mengambil set tag objek `doc3.rtf`, yang memiliki beberapa tag.

```
aws s3api get-object-tagging \  
--bucket my-bucket \  
--key doc3.rtf
```

Output:

```
{  
  "TagSet": [  
    {  
      "Value": "confidential",  
      "Key": "designation"  
    },  
    {  
      "Value": "finance",  
      "Key": "department"  
    },  
    {  
      "Value": "payroll",  
      "Key": "team"  
    }  
  ]  
}
```

- Untuk detail API, lihat [GetObjectTagging](#) di Referensi AWS CLI Perintah.

## get-object-torrent

Contoh kode berikut menunjukkan cara menggunakan `get-object-torrent`.

## AWS CLI

Perintah berikut membuat torrent untuk objek dalam bucket bernama `my-bucket`:

```
aws s3api get-object-torrent --bucket my-bucket --key large-video-file.mp4 large-video-file.torrent
```

File torrent disimpan secara lokal di folder saat ini. Perhatikan bahwa output filename (`large-video-file.torrent`) ditentukan tanpa nama pilihan dan harus menjadi argumen terakhir dalam perintah.

- Untuk detail API, lihat [GetObjectTorrent](#) di Referensi AWS CLI Perintah.

## get-object

Contoh kode berikut menunjukkan cara menggunakan `get-object`.

## AWS CLI

Contoh berikut menggunakan `get-object` perintah untuk mengunduh objek dari Amazon S3:

```
aws s3api get-object --bucket text-content --key dir/my_images.tar.bz2 my_images.tar.bz2
```

Perhatikan bahwa parameter outfile ditentukan tanpa nama opsi seperti “`--outfile`”. Nama file output harus menjadi parameter terakhir dalam perintah.

Contoh di bawah ini menunjukkan penggunaan `--range` untuk men-download rentang byte tertentu dari sebuah objek. Perhatikan rentang byte perlu diawali dengan “`byte =`”:

```
aws s3api get-object --bucket text-content --key dir/my_data --range bytes=8888-9999 my_data_range
```

Untuk informasi selengkapnya tentang mengambil objek, lihat Mendapatkan Objek di Panduan Pengembang Amazon S3.

- Untuk detail API, lihat [GetObject](#) di Referensi AWS CLI Perintah.

## get-public-access-block

Contoh kode berikut menunjukkan cara menggunakan `get-public-access-block`.

## AWS CLI

Untuk menyetel atau memodifikasi konfigurasi blokir akses publik untuk bucket

`get-public-access-block` Contoh berikut menampilkan konfigurasi blokir akses publik untuk bucket yang ditentukan.

```
aws s3api get-public-access-block \  
  --bucket my-bucket
```

Output:

```
{  
  "PublicAccessBlockConfiguration": {  
    "IgnorePublicAcls": true,  
    "BlockPublicPolicy": true,  
    "BlockPublicAcls": true,  
    "RestrictPublicBuckets": true  
  }  
}
```

- Untuk detail API, lihat [GetPublicAccessBlock](#) di Referensi AWS CLI Perintah.

## head-bucket

Contoh kode berikut menunjukkan cara menggunakan `head-bucket`.

### AWS CLI

Perintah berikut memverifikasi akses ke bucket bernama `my-bucket`:

```
aws s3api head-bucket --bucket my-bucket
```

Jika bucket ada dan Anda memiliki akses ke sana, tidak ada output yang dikembalikan. Jika tidak, pesan kesalahan akan ditampilkan. Sebagai contoh:

```
A client error (404) occurred when calling the HeadBucket operation: Not Found
```

- Untuk detail API, lihat [HeadBucket](#) di Referensi AWS CLI Perintah.

## head-object

Contoh kode berikut menunjukkan cara menggunakan `head-object`.

### AWS CLI

Perintah berikut mengambil metadata untuk objek dalam bucket bernama: `my-bucket`

```
aws s3api head-object --bucket my-bucket --key index.html
```

Output:

```
{
  "AcceptRanges": "bytes",
  "ContentType": "text/html",
  "LastModified": "Thu, 16 Apr 2015 18:19:14 GMT",
  "ContentLength": 77,
  "VersionId": "null",
  "ETag": "\"30a6ec7e1a9ad79c203d05a589c8b400\"",
  "Metadata": {}
}
```

- Untuk detail API, lihat [HeadObject](#) di Referensi AWS CLI Perintah.

## list-bucket-analytics-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-bucket-analytics-configurations`.

### AWS CLI

Untuk mengambil daftar konfigurasi analitik untuk bucket

Berikut ini akan `list-bucket-analytics-configurations` mengambil daftar konfigurasi analitik untuk bucket yang ditentukan.

```
aws s3api list-bucket-analytics-configurations \
  --bucket my-bucket
```

Output:

```
{
  "AnalyticsConfigurationList": [
    {
      "StorageClassAnalysis": {},
      "Id": "1"
    }
  ],
  "IsTruncated": false
}
```

- Untuk detail API, lihat [ListBucketAnalyticsConfigurations](#) di Referensi AWS CLI Perintah.

## list-bucket-intelligent-tiering-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-bucket-intelligent-tiering-configurations`.

### AWS CLI

Untuk mengambil semua konfigurasi S3 Intelligent-Tiering pada bucket

`list-bucket-intelligent-tiering-configurations` Contoh berikut mengambil semua konfigurasi S3 Intelligent-Tiering pada bucket.

```
aws s3api list-bucket-intelligent-tiering-configurations \
  --bucket DOC-EXAMPLE-BUCKET
```

### Output:

```
{
  "IsTruncated": false,
  "IntelligentTieringConfigurationList": [
    {
      "Id": "ExampleConfig",
      "Filter": {
        "Prefix": "images"
      },
      "Status": "Enabled",
      "Tierings": [
        {
          "Days": 90,
          "AccessTier": "ARCHIVE_ACCESS"
        }
      ]
    }
  ]
}
```

```
    },
    {
      "Days": 180,
      "AccessTier": "DEEP_ARCHIVE_ACCESS"
    }
  ]
},
{
  "Id": "ExampleConfig2",
  "Status": "Disabled",
  "Tierings": [
    {
      "Days": 730,
      "AccessTier": "ARCHIVE_ACCESS"
    }
  ]
},
{
  "Id": "ExampleConfig3",
  "Filter": {
    "Tag": {
      "Key": "documents",
      "Value": "taxes"
    }
  },
  "Status": "Enabled",
  "Tierings": [
    {
      "Days": 90,
      "AccessTier": "ARCHIVE_ACCESS"
    },
    {
      "Days": 365,
      "AccessTier": "DEEP_ARCHIVE_ACCESS"
    }
  ]
}
]
```

Untuk informasi selengkapnya, lihat [Menggunakan S3 Intelligent-Tiering di](#) Panduan Pengguna Amazon S3.

- Untuk detail API, lihat [ListBucketIntelligentTieringConfigurations](#) di Referensi AWS CLI Perintah.



## list-bucket-inventory-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-bucket-inventory-configurations`.

### AWS CLI

Untuk mengambil daftar konfigurasi inventaris untuk bucket

`list-bucket-inventory-configurations` Contoh berikut mencantumkan konfigurasi inventaris untuk bucket yang ditentukan.

```
aws s3api list-bucket-inventory-configurations \
  --bucket my-bucket
```

### Output:

```
{
  "InventoryConfigurationList": [
    {
      "IsEnabled": true,
      "Destination": {
        "S3BucketDestination": {
          "Format": "ORC",
          "Bucket": "arn:aws:s3:::my-bucket",
          "AccountId": "123456789012"
        }
      },
      "IncludedObjectVersions": "Current",
      "Id": "1",
      "Schedule": {
        "Frequency": "Weekly"
      }
    },
    {
      "IsEnabled": true,
      "Destination": {
        "S3BucketDestination": {
          "Format": "CSV",
          "Bucket": "arn:aws:s3:::my-bucket",
          "AccountId": "123456789012"
        }
      }
    }
  ],
}
```

```
        "IncludedObjectVersions": "Current",
        "Id": "2",
        "Schedule": {
            "Frequency": "Daily"
        }
    },
    "IsTruncated": false
}
```

- Untuk detail API, lihat [ListBucketInventoryConfigurations](#) di Referensi AWS CLI Perintah.

## list-bucket-metrics-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-bucket-metrics-configurations`.

### AWS CLI

Untuk mengambil daftar konfigurasi metrik untuk bucket

`list-bucket-metrics-configurations` Contoh berikut mengambil daftar konfigurasi metrik untuk bucket yang ditentukan.

```
aws s3api list-bucket-metrics-configurations \
  --bucket my-bucket
```

Output:

```
{
  "IsTruncated": false,
  "MetricsConfigurationList": [
    {
      "Filter": {
        "Prefix": "logs"
      },
      "Id": "123"
    },
    {
      "Filter": {
        "Prefix": "tmp"
      },
      "Id": "234"
    }
  ]
}
```

```
    }  
  ]  
}
```

- Untuk detail API, lihat [ListBucketMetricsConfigurations](#) di Referensi AWS CLI Perintah.

## list-buckets

Contoh kode berikut menunjukkan cara menggunakan `list-buckets`.

### AWS CLI

Perintah berikut menggunakan `list-buckets` perintah untuk menampilkan nama semua bucket Amazon S3 Anda (di semua wilayah):

```
aws s3api list-buckets --query "Buckets[].Name"
```

Opsi kueri menyaring output dari `list-buckets` down ke hanya nama bucket.

Untuk informasi selengkapnya tentang bucket, lihat [Bekerja dengan Bucket Amazon S3](#) di [Panduan Pengembang Amazon S3](#).

- Untuk detail API, lihat [ListBuckets](#) di Referensi AWS CLI Perintah.

## list-multipart-uploads

Contoh kode berikut menunjukkan cara menggunakan `list-multipart-uploads`.

### AWS CLI

Perintah berikut mencantumkan semua unggahan multipart aktif untuk bucket bernama: `my-bucket`

```
aws s3api list-multipart-uploads --bucket my-bucket
```

Output:

```
{  
  "Uploads": [  
    {  
      "Initiator": {  
        "DisplayName": "username",
```

```

        "ID": "arn:aws:iam::0123456789012:user/username"
    },
    "Initiated": "2015-06-02T18:01:30.000Z",
    "UploadId":
"dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZ1jF.Yxwh6XG7WfS2vC4to6HiV6Yj1x.cph0gtNBtJ8P3URC
    "StorageClass": "STANDARD",
    "Key": "multipart/01",
    "Owner": {
        "DisplayName": "aws-account-name",
        "ID":
"100719349fc3b6dcd7c820a124bf7aec408092c3d7b51b38494939801fc248b"
    }
    }
    ],
    "CommonPrefixes": []
}

```

Unggahan multipart yang sedang berlangsung menimbulkan biaya penyimpanan di Amazon S3. Selesaikan atau batalkan unggahan multibagian aktif untuk menghapus bagian-bagiannya dari akun Anda.

- Untuk detail API, lihat [ListMultipartUploads](#) di Referensi AWS CLI Perintah.

## list-object-versions

Contoh kode berikut menunjukkan cara menggunakan `list-object-versions`.

### AWS CLI

Perintah berikut mengambil informasi versi untuk objek dalam bucket bernama `my-bucket`:

```
aws s3api list-object-versions --bucket my-bucket --prefix index.html
```

Output:

```

{
  "DeleteMarkers": [
    {
      "Owner": {
        "DisplayName": "my-username",
        "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
      }
    },

```

```

    "IsLatest": true,
    "VersionId": "B2VsEK5saUNNHKc0AJj7hIE86RozToyq",
    "Key": "index.html",
    "LastModified": "2015-11-10T00:57:03.000Z"
  },
  {
    "Owner": {
      "DisplayName": "my-username",
      "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
    },
    "IsLatest": false,
    "VersionId": ".FLQEZscLIcfxSq.jsFJ.szUkmng2Yw6",
    "Key": "index.html",
    "LastModified": "2015-11-09T23:32:20.000Z"
  }
],
"Versions": [
  {
    "LastModified": "2015-11-10T00:20:11.000Z",
    "VersionId": "Rb_l2T8UHDkFEwCgJjhlGPOZC0qJ.vpD",
    "ETag": "\"0622528de826c0df5db1258a23b80be5\"",
    "StorageClass": "STANDARD",
    "Key": "index.html",
    "Owner": {
      "DisplayName": "my-username",
      "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
    },
    "IsLatest": false,
    "Size": 38
  },
  {
    "LastModified": "2015-11-09T23:26:41.000Z",
    "VersionId": "rasWWGpgk9E4s0LyTJgusGeRQKLVIAff",
    "ETag": "\"06225825b8028de826c0df5db1a23be5\"",
    "StorageClass": "STANDARD",
    "Key": "index.html",
    "Owner": {
      "DisplayName": "my-username",
      "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
    },
    "IsLatest": false,

```

```

        "Size": 38
      },
      {
        "LastModified": "2015-11-09T22:50:50.000Z",
        "VersionId": "null",
        "ETag": "\"d1f45267a863c8392e07d24dd592f1b9\"",
        "StorageClass": "STANDARD",
        "Key": "index.html",
        "Owner": {
          "DisplayName": "my-username",
          "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
        },
        "IsLatest": false,
        "Size": 533823
      }
    ]
  }
}

```

- Untuk detail API, lihat [ListObjectVersions](#) di Referensi AWS CLI Perintah.

## list-objects-v2

Contoh kode berikut menunjukkan cara menggunakan `list-objects-v2`.

### AWS CLI

Untuk mendapatkan daftar objek dalam ember

`list-objects-v2` Contoh berikut mencantumkan objek dalam bucket yang ditentukan.

```
aws s3api list-objects-v2 \
  --bucket my-bucket
```

Output:

```

{
  "Contents": [
    {
      "LastModified": "2019-11-05T23:11:50.000Z",
      "ETag": "\"621503c373607d548b37cff8778d992c\"",
      "StorageClass": "STANDARD",
      "Key": "doc1.rtf",

```

```

        "Size": 391
    },
    {
        "LastModified": "2019-11-05T23:11:50.000Z",
        "ETag": "\"a2cecc36ab7c7fe3a71a273b9d45b1b5\"",
        "StorageClass": "STANDARD",
        "Key": "doc2.rtf",
        "Size": 373
    },
    {
        "LastModified": "2019-11-05T23:11:50.000Z",
        "ETag": "\"08210852f65a2e9cb999972539a64d68\"",
        "StorageClass": "STANDARD",
        "Key": "doc3.rtf",
        "Size": 399
    },
    {
        "LastModified": "2019-11-05T23:11:50.000Z",
        "ETag": "\"d1852dd683f404306569471af106988e\"",
        "StorageClass": "STANDARD",
        "Key": "doc4.rtf",
        "Size": 6225
    }
]
}

```

- Untuk detail API, lihat [ListObjectsV2](#) di Referensi AWS CLI Perintah.

## list-objects

Contoh kode berikut menunjukkan cara menggunakan `list-objects`.

### AWS CLI

Contoh berikut menggunakan `list-objects` perintah untuk menampilkan nama-nama semua objek dalam bucket yang ditentukan:

```
aws s3api list-objects --bucket text-content --query 'Contents[].{Key: Key, Size: Size}'
```

Contoh menggunakan `--query` argumen untuk memfilter output `list-objects` turun ke nilai kunci dan ukuran untuk setiap objek

Untuk informasi selengkapnya tentang objek, lihat Bekerja dengan Objek Amazon S3 di Panduan Pengembang Amazon S3.

- Untuk detail API, lihat [ListObjects](#) di Referensi AWS CLI Perintah.

## list-parts

Contoh kode berikut menunjukkan cara menggunakan `list-parts`.

### AWS CLI

Perintah berikut mencantumkan semua bagian yang telah diunggah untuk unggahan multibagian dengan kunci `multipart/01` di bucket: `my-bucket`

```
aws s3api list-parts --bucket my-bucket --key 'multipart/01' --upload-id
dfRtDYU0WwCCcH43C3WfbkR0NycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3URCS
```

### Output:

```
{
  "Owner": {
    "DisplayName": "aws-account-name",
    "ID": "100719349fc3b6dcd7c820a124bf7aec408092c3d7b51b38494939801fc248b"
  },
  "Initiator": {
    "DisplayName": "username",
    "ID": "arn:aws:iam::0123456789012:user/username"
  },
  "Parts": [
    {
      "LastModified": "2015-06-02T18:07:35.000Z",
      "PartNumber": 1,
      "ETag": "\"e868e0f4719e394144ef36531ee6824c\"",
      "Size": 5242880
    },
    {
      "LastModified": "2015-06-02T18:07:42.000Z",
      "PartNumber": 2,
      "ETag": "\"6bb2b12753d66fe86da4998aa33fffb0\"",
      "Size": 5242880
    },
    {
      "LastModified": "2015-06-02T18:07:47.000Z",
```



```
        "PartNumber": 3,  
        "ETag": "\"d0a0112e841abec9c9ec83406f0159c8\"",  
        "Size": 5242880  
    }  
  ],  
  "StorageClass": "STANDARD"  
}
```

- Untuk detail API, lihat [ListParts](#) di Referensi AWS CLI Perintah.

## ls

Contoh kode berikut menunjukkan cara menggunakan `ls`.

### AWS CLI

Contoh 1: Daftar semua bucket milik pengguna

`ls` Perintah berikut mencantumkan semua bucket yang dimiliki oleh pengguna. Dalam contoh ini, pengguna memiliki ember `mybucket` dan `mybucket2`. Stempel waktu adalah tanggal ember dibuat, ditunjukkan di zona waktu mesin Anda. Tanggal ini dapat berubah saat membuat perubahan pada bucket Anda, seperti mengedit kebijakan bucket. Perhatikan jika `s3://` digunakan untuk argumen jalur `<S3Uri>`, itu akan mencantumkan semua bucket juga.

```
aws s3 ls
```

Output:

```
2013-07-11 17:08:50 mybucket  
2013-07-24 14:55:44 mybucket2
```

Contoh 2: Daftar semua awalan dan objek dalam ember

`ls` Perintah berikut mencantumkan objek dan awalan umum di bawah bucket dan awalan yang ditentukan. Dalam contoh ini, pengguna memiliki ember `mybucket` dengan objek `test.txt` dan `somePrefix/test.txt`. Itu `LastWriteTime` dan `Length` sewenang-wenang. Perhatikan bahwa karena `ls` perintah tidak memiliki interaksi dengan sistem file lokal, skema `s3://` URI tidak diperlukan untuk menyelesaikan ambiguitas dan dapat dihilangkan.

```
aws s3 ls s3://mybucket
```

**Output:**

```
2013-07-25 17:06:27      PRE somePrefix/  
                        88 test.txt
```

**Contoh 3: Daftar semua awalan dan objek dalam bucket dan awalan tertentu**

lsPerintah berikut mencantumkan objek dan awalan umum di bawah bucket dan awalan yang ditentukan. Namun, tidak ada objek atau awalan umum di bawah ember dan awalan yang ditentukan.

```
aws s3 ls s3://mybucket/noExistPrefix
```

**Output:**

```
None
```

**Contoh 4: Secara rekursif mencantumkan semua awalan dan objek dalam ember**

lsPerintah berikut akan secara rekursif mencantumkan objek dalam ember. Alih-alih ditampilkan PRE dirname/ dalam output, semua konten dalam ember akan terdaftar secara berurutan.

```
aws s3 ls s3://mybucket \  
--recursive
```

**Output:**

```
2013-09-02 21:37:53      10 a.txt  
2013-09-02 21:37:53  2863288 foo.zip  
2013-09-02 21:32:57      23 foo/bar/.baz/a  
2013-09-02 21:32:58      41 foo/bar/.baz/b  
2013-09-02 21:32:57     281 foo/bar/.baz/c  
2013-09-02 21:32:57      73 foo/bar/.baz/d  
2013-09-02 21:32:57     452 foo/bar/.baz/e  
2013-09-02 21:32:57     896 foo/bar/.baz/hooks/bar  
2013-09-02 21:32:57     189 foo/bar/.baz/hooks/foo  
2013-09-02 21:32:57     398 z.txt
```

**Contoh 5: Meringkas semua awalan dan objek dalam ember**

`ls` Perintah berikut menunjukkan perintah yang sama menggunakan opsi `--human-readable` dan `--summarize`. `--human-readable` menampilkan ukuran file dalam bytes/MiB/KiB/GiB/TiB/PiB/EiB. `--summarize` menampilkan jumlah total objek dan ukuran total di akhir daftar hasil:

```
aws s3 ls s3://mybucket \
  --recursive \
  --human-readable \
  --summarize
```

Output:

```
2013-09-02 21:37:53  10 Bytes a.txt
2013-09-02 21:37:53  2.9 MiB foo.zip
2013-09-02 21:32:57  23 Bytes foo/bar/.baz/a
2013-09-02 21:32:58  41 Bytes foo/bar/.baz/b
2013-09-02 21:32:57 281 Bytes foo/bar/.baz/c
2013-09-02 21:32:57  73 Bytes foo/bar/.baz/d
2013-09-02 21:32:57 452 Bytes foo/bar/.baz/e
2013-09-02 21:32:57 896 Bytes foo/bar/.baz/hooks/bar
2013-09-02 21:32:57 189 Bytes foo/bar/.baz/hooks/foo
2013-09-02 21:32:57 398 Bytes z.txt

Total Objects: 10
Total Size: 2.9 MiB
```

Contoh 6: Daftar dari titik akses S3

Berikut daftar `ls` perintah objek dari access point (myaccesspoint):

```
aws s3 ls s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
```

Output:

```
                PRE somePrefix/
2013-07-25 17:06:27      88 test.txt
```

- Untuk detail API, lihat [Ls](#) di Referensi AWS CLI Perintah.

**mb**

Contoh kode berikut menunjukkan cara menggunakan `mb`.

## AWS CLI

### Contoh 1: Buat ember

mbPerintah berikut membuat ember. Dalam contoh ini, pengguna membuat embermybucket. Bucket dibuat di wilayah yang ditentukan dalam file konfigurasi pengguna:

```
aws s3 mb s3://mybucket
```

Output:

```
make_bucket: s3://mybucket
```

### Contoh 2: Buat ember di wilayah yang ditentukan

mbPerintah berikut membuat bucket di wilayah yang ditentukan oleh `--region` parameter. Dalam contoh ini, pengguna membuat bucket mybucket di wilayah tersebutus-west-1:

```
aws s3 mb s3://mybucket \  
--region us-west-1
```

Output:

```
make_bucket: s3://mybucket
```

- Untuk detail API, lihat [Mb](#) di Referensi AWS CLI Perintah.

## mv

Contoh kode berikut menunjukkan cara menggunakanmv.

## AWS CLI

### Contoh 1: Pindahkan file lokal ke bucket yang ditentukan

mvPerintah berikut memindahkan satu file ke bucket dan kunci tertentu.

```
aws s3 mv test.txt s3://mybucket/test2.txt
```

Output:

```
move: test.txt to s3://mybucket/test2.txt
```

Contoh 2: Pindahkan objek ke bucket dan kunci yang ditentukan

mvPerintah berikut memindahkan objek s3 tunggal ke bucket dan kunci tertentu.

```
aws s3 mv s3://mybucket/test.txt s3://mybucket/test2.txt
```

Output:

```
move: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

Contoh 3: Pindahkan objek S3 ke direktori lokal

mvPerintah berikut memindahkan satu objek ke file tertentu secara lokal.

```
aws s3 mv s3://mybucket/test.txt test2.txt
```

Output:

```
move: s3://mybucket/test.txt to test2.txt
```

Contoh 4: Pindahkan objek dengan nama aslinya ke ember yang ditentukan

mvPerintah berikut memindahkan satu objek ke bucket tertentu sambil mempertahankan nama aslinya:

```
aws s3 mv s3://mybucket/test.txt s3://mybucket2/
```

Output:

```
move: s3://mybucket/test.txt to s3://mybucket2/test.txt
```

Contoh 5: Pindahkan semua objek dan awalan dalam ember ke direktori lokal

Ketika diteruskan dengan parameter `--recursive`, mv perintah berikut secara rekursif memindahkan semua objek di bawah awalan dan ember yang ditentukan ke direktori tertentu. Dalam contoh ini, ember mybucket memiliki objek test1.txt dan test2.txt.

```
aws s3 mv s3://mybucket . \
  --recursive
```

**Output:**

```
move: s3://mybucket/test1.txt to test1.txt
move: s3://mybucket/test2.txt to test2.txt
```

Contoh 6: Pindahkan semua objek dan awalan dalam ember ke direktori lokal, kecuali file ``.jpg``

Ketika diteruskan dengan parameter `--recursive`, `mv` perintah berikut secara rekursif memindahkan semua file di bawah direktori tertentu ke bucket dan awalan tertentu sementara mengecualikan beberapa file dengan menggunakan parameter. `--exclude` Dalam contoh ini, direktori `myDir` memiliki file `test1.txt` dan `test2.jpg`.

```
aws s3 mv myDir s3://mybucket/ \
  --recursive \
  --exclude "*.jpg"
```

**Output:**

```
move: myDir/test1.txt to s3://mybucket2/test1.txt
```

Contoh 7: Pindahkan semua objek dan awalan dalam ember ke direktori lokal, kecuali awalan yang ditentukan

Ketika diteruskan dengan parameter `--recursive`, `mv` perintah berikut secara rekursif memindahkan semua objek di bawah bucket tertentu ke bucket lain sambil mengecualikan beberapa objek dengan menggunakan parameter. `--exclude` Dalam contoh ini, ember `mybucket` memiliki objek `test1.txt` dan `another/test1.txt`.

```
aws s3 mv s3://mybucket/ s3://mybucket2/ \
  --recursive \
  --exclude "mybucket/another/*"
```

**Output:**

```
move: s3://mybucket/test1.txt to s3://mybucket2/test1.txt
```

### Contoh 8: Pindahkan objek ke bucket yang ditentukan dan atur ACL

`mv` Perintah berikut memindahkan satu objek ke bucket dan kunci tertentu saat mengatur ACL ke `public-read-write`.

```
aws s3 mv s3://mybucket/test.txt s3://mybucket/test2.txt \  
--acl public-read-write
```

Output:

```
move: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

### Contoh 9: Pindahkan file lokal ke bucket yang ditentukan dan berikan izin

`mv` Perintah berikut menggambarkan penggunaan `--grants` opsi untuk memberikan akses baca ke semua pengguna dan kontrol penuh ke pengguna tertentu yang diidentifikasi oleh alamat email mereka.

```
aws s3 mv file.txt s3://mybucket/ \  
--grants read=uri=http://acs.amazonaws.com/groups/global/AllUsers  
full=emailaddress=user@example.com
```

Output:

```
move: file.txt to s3://mybucket/file.txt
```

### Contoh 10: Pindahkan file ke titik akses S3

`mv` Perintah berikut memindahkan satu file bernama `mydoc.txt` ke titik akses bernama `myaccesspoint` pada kunci bernama `mykey`.

```
aws s3 mv mydoc.txt s3://arn:aws:s3:us-west-2:123456789012:accesspoint/  
myaccesspoint/mykey
```

Output:

```
move: mydoc.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/  
mykey
```

- Untuk detail API, lihat [Mv](#) di Referensi AWS CLI Perintah.

## presign

Contoh kode berikut menunjukkan cara menggunakan `presign`.

### AWS CLI

Contoh 1: Untuk membuat URL yang telah ditandatangani sebelumnya dengan masa pakai default satu jam yang menautkan ke objek dalam bucket S3

`presign` Perintah berikut menghasilkan URL yang telah ditandatangani sebelumnya untuk bucket dan kunci tertentu yang valid selama satu jam.

```
aws s3 presign s3://DOC-EXAMPLE-BUCKET/test2.txt
```

Output:

```
https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/key?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAEXAMPLE123456789%2F20210621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-Date=20210621T041609Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=EXAMPLE1234494d5fba3fed607f98018e1dfc62e2529ae96d844123456
```

Contoh 2: Untuk membuat URL yang telah ditandatangani sebelumnya dengan masa pakai khusus yang menautkan ke objek di bucket S3

`presign` Perintah berikut menghasilkan URL yang telah ditandatangani sebelumnya untuk bucket dan kunci tertentu yang berlaku selama satu minggu.

```
aws s3 presign s3://DOC-EXAMPLE-BUCKET/test2.txt \
  --expires-in 604800
```

Output:

```
https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/key?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAEXAMPLE123456789%2F20210621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-Date=20210621T041609Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=EXAMPLE1234494d5fba3fed607f98018e1dfc62e2529ae96d844123456
```



Untuk informasi selengkapnya, lihat [Berbagi Objek dengan Orang Lain](#) di panduan Panduan Pengembang S3.

- Untuk detail API, lihat [Presign](#) di Referensi AWS CLI Perintah.

## **put-bucket-accelerate-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-accelerate-configuration`.

### AWS CLI

Untuk mengatur konfigurasi percepatan ember

`put-bucket-accelerate-configuration` Contoh berikut memungkinkan konfigurasi percepatan untuk bucket yang ditentukan.

```
aws s3api put-bucket-accelerate-configuration \  
  --bucket my-bucket \  
  --accelerate-configuration Status=Enabled
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketAccelerateConfiguration](#) di Referensi AWS CLI Perintah.

## **put-bucket-acl**

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-acl`.

### AWS CLI

Contoh ini memberikan `full control` kepada dua AWS pengguna (`user1@example.com` dan `user2@example.com`) dan `read` izin untuk semua orang:

```
aws s3api put-bucket-acl --bucket MyBucket --grant-full-control  
  emailaddress=user1@example.com,emailaddress=user2@example.com --grant-read  
  uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Lihat <http://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketPUTacl.html> untuk detail tentang ACL kustom (perintah `s3api ACL`, seperti `put-bucket-acl`, menggunakan notasi argumen singkatan yang sama).

- Untuk detail API, lihat [PutBucketAcl](#) di Referensi AWS CLI Perintah.

## put-bucket-analytics-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-analytics-configuration`.

### AWS CLI

Untuk menetapkan konfigurasi analitik untuk bucket

`put-bucket-analytics-configuration` Contoh berikut mengonfigurasi analitik untuk bucket yang ditentukan.

```
aws s3api put-bucket-analytics-configuration \
  --bucket my-bucket --id 1 \
  --analytics-configuration '{"Id": "1","StorageClassAnalysis": {}}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketAnalyticsConfiguration](#) di Referensi AWS CLI Perintah.

## put-bucket-cors

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-cors`.

### AWS CLI

Contoh berikut memungkinkan PUT, POST, dan DELETE permintaan dari `www.example.com`, dan memungkinkan GET permintaan dari domain apa pun:

```
aws s3api put-bucket-cors --bucket MyBucket --cors-configuration file://cors.json

cors.json:
{
  "CORSRules": [
    {
      "AllowedOrigins": ["http://www.example.com"],
      "AllowedHeaders": ["*"],
      "AllowedMethods": ["PUT", "POST", "DELETE"],
      "MaxAgeSeconds": 3000,
      "ExposeHeaders": ["x-amz-server-side-encryption"]
    }
  ],
```

```

    {
      "AllowedOrigins": ["*"],
      "AllowedHeaders": ["Authorization"],
      "AllowedMethods": ["GET"],
      "MaxAgeSeconds": 3000
    }
  ]
}

```

- Untuk detail API, lihat [PutBucketCors](#) di Referensi AWS CLI Perintah.

## put-bucket-encryption

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-encryption`.

### AWS CLI

Untuk mengonfigurasi enkripsi sisi server untuk bucket

`put-bucket-encryption` Contoh berikut menetapkan enkripsi AES256 sebagai default untuk bucket yang ditentukan.

```

aws s3api put-bucket-encryption \
  --bucket my-bucket \
  --server-side-encryption-configuration '{"Rules":
  [{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm": "AES256"}}]}'

```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketEncryption](#) di Referensi AWS CLI Perintah.

## put-bucket-intelligent-tiering-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-intelligent-tiering-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi S3 Intelligent-Tiering pada bucket

`put-bucket-intelligent-tiering-configuration` Contoh berikut memperbarui konfigurasi S3 Intelligent-Tiering, bernama `ExampleConfig`, pada bucket. Konfigurasi akan

mengalihkan objek yang belum diakses di bawah gambar awalan ke Akses Arsip setelah 90 hari dan Akses Arsip Dalam setelah 180 hari.

```
aws s3api put-bucket-intelligent-tiering-configuration \  
  --bucket DOC-EXAMPLE-BUCKET \  
  --id "ExampleConfig" \  
  --intelligent-tiering-configuration file://intelligent-tiering-  
configuration.json
```

Isi dari `intelligent-tiering-configuration.json`:

```
{  
  "Id": "ExampleConfig",  
  "Status": "Enabled",  
  "Filter": {  
    "Prefix": "images"  
  },  
  "Tierings": [  
    {  
      "Days": 90,  
      "AccessTier": "ARCHIVE_ACCESS"  
    },  
    {  
      "Days": 180,  
      "AccessTier": "DEEP_ARCHIVE_ACCESS"  
    }  
  ]  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyetel Kepemilikan Objek pada bucket yang ada](#) di Panduan Pengguna Amazon S3.

- Untuk detail API, lihat [PutBucketIntelligentTieringConfiguration](#) di Referensi AWS CLI Perintah.

## **put-bucket-inventory-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-inventory-configuration`.

### AWS CLI

Contoh 1: Untuk mengatur konfigurasi inventaris untuk bucket

`put-bucket-inventory-configuration` Contoh berikut menetapkan laporan inventaris berformat ORC mingguan untuk bucket. `my-bucket`

```
aws s3api put-bucket-inventory-configuration \
  --bucket my-bucket \
  --id 1 \
  --inventory-configuration '{"Destination": { "S3BucketDestination":
{ "AccountId": "123456789012", "Bucket": "arn:aws:s3:::my-bucket", "Format":
"ORC" }}, "IsEnabled": true, "Id": "1", "IncludedObjectVersions": "Current",
"Schedule": { "Frequency": "Weekly" } }'
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk mengatur konfigurasi inventaris untuk bucket

`put-bucket-inventory-configuration` Contoh berikut menetapkan laporan inventaris berformat CSV harian untuk bucket. `my-bucket`

```
aws s3api put-bucket-inventory-configuration \
  --bucket my-bucket \
  --id 2 \
  --inventory-configuration '{"Destination": { "S3BucketDestination":
{ "AccountId": "123456789012", "Bucket": "arn:aws:s3:::my-bucket", "Format":
"CSV" }}, "IsEnabled": true, "Id": "2", "IncludedObjectVersions": "Current",
"Schedule": { "Frequency": "Daily" } }'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketInventoryConfiguration](#) di Referensi AWS CLI Perintah.

## **put-bucket-lifecycle-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-lifecycle-configuration`.

### AWS CLI

Perintah berikut menerapkan konfigurasi siklus hidup ke bucket bernama: `my-bucket`

```
aws s3api put-bucket-lifecycle-configuration --bucket my-bucket --lifecycle-
configuration file://lifecycle.json
```

File `lifecycle.json` adalah dokumen JSON di folder saat ini yang menentukan dua aturan:

```
{
  "Rules": [
    {
      "ID": "Move rotated logs to Glacier",
      "Prefix": "rotated/",
      "Status": "Enabled",
      "Transitions": [
        {
          "Date": "2015-11-10T00:00:00.000Z",
          "StorageClass": "GLACIER"
        }
      ]
    },
    {
      "Status": "Enabled",
      "Prefix": "",
      "NoncurrentVersionTransitions": [
        {
          "NoncurrentDays": 2,
          "StorageClass": "GLACIER"
        }
      ],
      "ID": "Move old versions to Glacier"
    }
  ]
}
```

Aturan pertama memindahkan file dengan awalan `rotated` ke Glacier pada tanggal yang ditentukan. Aturan kedua memindahkan versi objek lama ke Glacier ketika mereka tidak lagi terkini. Untuk informasi tentang format stempel waktu yang dapat diterima, lihat [Menentukan Nilai Parameter](#) dalam Panduan Pengguna CLI AWS .

- Untuk detail API, lihat [PutBucketLifecycleConfiguration](#) di Referensi AWS CLI Perintah.

## **put-bucket-lifecycle**

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-lifecycle`.

### AWS CLI

Perintah berikut menerapkan konfigurasi siklus hidup ke bucket: `my-bucket`

```
aws s3api put-bucket-lifecycle --bucket my-bucket --lifecycle-configuration file://lifecycle.json
```

File `lifecycle.json` adalah dokumen JSON di folder saat ini yang menentukan dua aturan:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (objects in logs/2015/)",
      "Prefix": "logs/2015/",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    },
    {
      "Expiration": {
        "Date": "2016-01-01T00:00:00.000Z"
      },
      "ID": "Delete 2014 logs in 2016.",
      "Prefix": "logs/2014/",
      "Status": "Enabled"
    }
  ]
}
```

Aturan pertama memindahkan file ke Amazon Glacier setelah enam puluh hari. Aturan kedua menghapus file dari Amazon S3 pada tanggal yang ditentukan. Untuk informasi tentang format stempel waktu yang dapat diterima, lihat Menentukan Nilai Parameter dalam Panduan Pengguna CLI AWS .

Setiap aturan dalam contoh di atas menentukan kebijakan (`Transition` atau `Expiration`) dan awalan file (nama folder) yang berlaku. Anda juga dapat membuat aturan yang berlaku untuk seluruh bucket dengan menentukan awalan kosong:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (all objects in bucket)",
      "Prefix": "",
```

```

    "Status": "Enabled",
    "Transition": {
      "Days": 60,
      "StorageClass": "GLACIER"
    }
  }
]
}

```

- Untuk detail API, lihat [PutBucketLifecycle](#) di Referensi AWS CLI Perintah.

## put-bucket-logging

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-logging`.

### AWS CLI

Contoh 1: Untuk mengatur pencatatan kebijakan bucket

`put-bucket-logging` Contoh berikut menetapkan kebijakan logging untuk MyBucket. Pertama, berikan izin utama layanan logging dalam kebijakan bucket Anda menggunakan `put-bucket-policy` perintah.

```

aws s3api put-bucket-policy \
  --bucket MyBucket \
  --policy file://policy.json

```

Isi dari `policy.json`:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "S3ServerAccessLogsPolicy",
      "Effect": "Allow",
      "Principal": {"Service": "logging.s3.amazonaws.com"},
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::MyBucket/Logs/*",
      "Condition": {
        "ArnLike": {"aws:SourceARN": "arn:aws:s3:::SOURCE-BUCKET-NAME"},
        "StringEquals": {"aws:SourceAccount": "SOURCE-AWS-ACCOUNT-ID"}
      }
    }
  ]
}

```



```
    }  
  ]  
}
```

Untuk menerapkan kebijakan pencatatan, gunakan `put-bucket-logging`.

```
aws s3api put-bucket-logging \  
  --bucket MyBucket \  
  --bucket-logging-status file://logging.json
```

Isi dari `logging.json`:

```
{  
  "LoggingEnabled": {  
    "TargetBucket": "MyBucket",  
    "TargetPrefix": "Logs/"  
  }  
}
```

`put-bucket-policy` Perintah diperlukan untuk memberikan `s3:PutObject` izin kepada prinsipal layanan logging.

Untuk informasi selengkapnya, lihat [Logging Akses Server Amazon S3](#) di Panduan Pengguna Amazon S3.

Contoh 2: Untuk menetapkan kebijakan bucket untuk akses log hanya ke satu pengguna

`put-bucket-logging` Contoh berikut menetapkan kebijakan logging untuk `MyBucket`. AWS Pengguna `bob@example.com` akan memiliki kontrol penuh atas file log, dan tidak ada orang lain yang memiliki akses. Pertama, berikan izin S3 dengan `put-bucket-acl`.

```
aws s3api put-bucket-acl \  
  --bucket MyBucket \  
  --grant-write URI=http://acs.amazonaws.com/groups/s3/LogDelivery \  
  --grant-read-acp URI=http://acs.amazonaws.com/groups/s3/LogDelivery
```

Kemudian terapkan kebijakan pencatatan menggunakan `put-bucket-logging`.

```
aws s3api put-bucket-logging \  
  --bucket MyBucket \  
  --bucket-logging-status file://logging.json
```

Isi dari `logging.json`:

```
{
  "LoggingEnabled": {
    "TargetBucket": "MyBucket",
    "TargetPrefix": "MyBucketLogs/",
    "TargetGrants": [
      {
        "Grantee": {
          "Type": "AmazonCustomerByEmail",
          "EmailAddress": "bob@example.com"
        },
        "Permission": "FULL_CONTROL"
      }
    ]
  }
}
```

`put-bucket-acl` perintah diperlukan untuk memberikan sistem pengiriman log S3 izin yang diperlukan (izin tulis dan baca-acl).

Untuk informasi selengkapnya, lihat [Logging Akses Server Amazon S3 di Panduan Pengembang Amazon S3](#).

- Untuk detail API, lihat [PutBucketLogging](#) di Referensi AWS CLI Perintah.

## **put-bucket-metrics-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-metrics-configuration`.

### AWS CLI

Untuk menyetel konfigurasi metrik untuk bucket

`put-bucket-metrics-configuration` Contoh berikut menetapkan konfigurasi metrik dengan ID 123 untuk bucket yang ditentukan.

```
aws s3api put-bucket-metrics-configuration \
  --bucket my-bucket \
  --id 123 \
  --metrics-configuration '{"Id": "123", "Filter": {"Prefix": "logs"}}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketMetricsConfiguration](#) di Referensi AWS CLI Perintah.

## put-bucket-notification-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-notification-configuration`.

### AWS CLI

Untuk mengaktifkan notifikasi yang ditentukan ke bucket

`put-bucket-notification-configuration` Contoh berikut menerapkan konfigurasi notifikasi ke bucket bernama `my-bucket`. File tersebut `notification.json` adalah dokumen JSON di folder saat ini yang menentukan topik SNS dan jenis acara untuk dipantau.

```
aws s3api put-bucket-notification-configuration \  
  --bucket my-bucket \  
  --notification-configuration file://notification.json
```

Isi dari `notification.json`:

```
{  
  "TopicConfigurations": [  
    {  
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic",  
      "Events": [  
        "s3:ObjectCreated:*"  
      ]  
    }  
  ]  
}
```

Topik SNS harus memiliki kebijakan IAM yang melekat padanya yang memungkinkan Amazon S3 untuk mempublikasikannya.

```
{  
  "Version": "2008-10-17",  
  "Id": "example-ID",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": "AmazonS3",  
      "Action": "sns:Publish",  
      "Resource": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic"  
    }  
  ]  
}
```

```
{
  "Sid": "example-statement-ID",
  "Effect": "Allow",
  "Principal": {
    "Service": "s3.amazonaws.com"
  },
  "Action": [
    "SNS:Publish"
  ],
  "Resource": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic",
  "Condition": {
    "ArnLike": {
      "aws:SourceArn": "arn:aws:s3:*:*:my-bucket"
    }
  }
}
```

- Untuk detail API, lihat [PutBucketNotificationConfiguration](#) di Referensi AWS CLI Perintah.

## put-bucket-notification

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-notification`.

### AWS CLI

Menerapkan konfigurasi notifikasi ke bucket bernama `my-bucket`:

```
aws s3api put-bucket-notification --bucket my-bucket --notification-configuration
file://notification.json
```

File `notification.json` ini adalah dokumen JSON di folder saat ini yang menentukan topik SNS dan jenis acara untuk dipantau:

```
{
  "TopicConfiguration": {
    "Event": "s3:ObjectCreated:*",
    "Topic": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic"
  }
}
```

Topik SNS harus memiliki kebijakan IAM yang melekat padanya yang memungkinkan Amazon S3 untuk mempublikasikannya:

```
{
  "Version": "2008-10-17",
  "Id": "example-ID",
  "Statement": [
    {
      "Sid": "example-statement-ID",
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": [
        "SNS:Publish"
      ],
      "Resource": "arn:aws:sns:us-west-2:123456789012:my-bucket",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:s3:*:*:my-bucket"
        }
      }
    }
  ]
}
```

- Untuk detail API, lihat [PutBucketNotification](#) di Referensi AWS CLI Perintah.

## put-bucket-ownership-controls

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-ownership-controls`.

### AWS CLI

Untuk memperbarui setelan kepemilikan bucket pada bucket

`put-bucket-ownership-controls` Contoh berikut memperbarui pengaturan kepemilikan bucket bucket.

```
aws s3api put-bucket-ownership-controls \
  --bucket DOC-EXAMPLE-BUCKET \
  --ownership-controls="Rules=[{ObjectOwnership=BucketOwnerEnforced}]"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyetel Kepemilikan Objek pada bucket yang ada](#) di Panduan Pengguna Amazon S3.

- Untuk detail API, lihat [PutBucketOwnershipControls](#) di Referensi AWS CLI Perintah.

## put-bucket-policy

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-policy`.

### AWS CLI

Contoh ini memungkinkan semua pengguna untuk mengambil objek apa pun MyBucket kecuali yang ada di MySecretFolder. Ini juga memberikan `put` dan `delete` izin kepada pengguna root AWS akun `1234-5678-9012`:

```
aws s3api put-bucket-policy --bucket MyBucket --policy file://policy.json
```

```
policy.json:
```

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::MyBucket/*"
    },
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::MyBucket/MySecretFolder/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:root"
      },
      "Action": [
        "s3:DeleteObject",
        "s3:PutObject"
      ]
    }
  ],
}
```

```

        "Resource": "arn:aws:s3:::MyBucket/*"
    }
]
}

```

- Untuk detail API, lihat [PutBucketPolicy](#) di Referensi AWS CLI Perintah.

## put-bucket-replication

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-replication`.

### AWS CLI

Untuk mengonfigurasi replikasi untuk bucket S3

`put-bucket-replication` Contoh berikut menerapkan konfigurasi replikasi ke bucket S3 yang ditentukan.

```

aws s3api put-bucket-replication \
  --bucket AWSDOC-EXAMPLE-BUCKET1 \
  --replication-configuration file://replication.json

```

Isi dari `replication.json`:

```

{
  "Role": "arn:aws:iam::123456789012:role/s3-replication-role",
  "Rules": [
    {
      "Status": "Enabled",
      "Priority": 1,
      "DeleteMarkerReplication": { "Status": "Disabled" },
      "Filter" : { "Prefix": ""},
      "Destination": {
        "Bucket": "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET2"
      }
    }
  ]
}

```

Bucket tujuan harus mengaktifkan versi. Peran yang ditentukan harus memiliki izin untuk menulis ke bucket tujuan dan memiliki hubungan kepercayaan yang memungkinkan Amazon S3 untuk mengambil peran tersebut.

## Contoh kebijakan izin peran:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetReplicationConfiguration",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET1"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObjectVersion",
        "s3:GetObjectVersionAcl",
        "s3:GetObjectVersionTagging"
      ],
      "Resource": [
        "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET1/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:ReplicateObject",
        "s3:ReplicateDelete",
        "s3:ReplicateTags"
      ],
      "Resource": "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET2/*"
    }
  ]
}
```

## Contoh kebijakan hubungan kepercayaan:

```
{
  "Version": "2012-10-17",
  "Statement": [
```



```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "s3.amazonaws.com"
  },
  "Action": "sts:AssumeRole"
}
]
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Ini adalah judul topik](#) di Panduan Pengguna Konsol Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [PutBucketReplication](#) di Referensi AWS CLI Perintah.

## put-bucket-request-payment

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-request-payment`.

### AWS CLI

Contoh 1: Untuk mengaktifkan konfigurasi `requester pays` untuk sebuah bucket

`put-bucket-request-payment` Contoh berikut memungkinkan `requester pays` untuk bucket yang ditentukan.

```
aws s3api put-bucket-request-payment \
  --bucket my-bucket \
  --request-payment-configuration '{"Payer":"Requester}"'
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menonaktifkan konfigurasi `requester pays` untuk bucket

`put-bucket-request-payment` Contoh berikut menonaktifkan `requester pays` untuk bucket yang ditentukan.

```
aws s3api put-bucket-request-payment \
  --bucket my-bucket \
```

```
--request-payment-configuration '{"Payer":"BucketOwner"}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutBucketRequestPayment](#) di Referensi AWS CLI Perintah.

## put-bucket-tagging

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-tagging`.

### AWS CLI

Perintah berikut menerapkan konfigurasi penandaan ke bucket bernama `my-bucket`:

```
aws s3api put-bucket-tagging --bucket my-bucket --tagging file://tagging.json
```

File `tagging.json` adalah dokumen JSON di folder saat ini yang menentukan tag:

```
{
  "TagSet": [
    {
      "Key": "organization",
      "Value": "marketing"
    }
  ]
}
```

Atau terapkan konfigurasi penandaan `my-bucket` langsung dari baris perintah:

```
aws s3api put-bucket-tagging --bucket my-bucket --tagging
'TagSet=[{Key=organization,Value=marketing}]'
```

- Untuk detail API, lihat [PutBucketTagging](#) di Referensi AWS CLI Perintah.

## put-bucket-versioning

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-versioning`.

### AWS CLI

Perintah berikut memungkinkan pembuatan versi pada bucket bernama: `my-bucket`

```
aws s3api put-bucket-versioning --bucket my-bucket --versioning-configuration
Status=Enabled
```

Perintah berikut memungkinkan pembuatan versi, dan menggunakan kode mfa

```
aws s3api put-bucket-versioning --bucket my-bucket --versioning-configuration
Status=Enabled --mfa "SERIAL 123456"
```

- Untuk detail API, lihat [PutBucketVersioning](#) di Referensi AWS CLI Perintah.

## put-bucket-website

Contoh kode berikut menunjukkan cara menggunakan `put-bucket-website`.

### AWS CLI

Menerapkan konfigurasi situs web statis ke bucket bernama `my-bucket`:

```
aws s3api put-bucket-website --bucket my-bucket --website-configuration file://
website.json
```

File tersebut `website.json` adalah dokumen JSON di folder saat ini yang menentukan halaman indeks dan kesalahan untuk situs web:

```
{
  "IndexDocument": {
    "Suffix": "index.html"
  },
  "ErrorDocument": {
    "Key": "error.html"
  }
}
```

- Untuk detail API, lihat [PutBucketWebsite](#) di Referensi AWS CLI Perintah.

## put-object-acl

Contoh kode berikut menunjukkan cara menggunakan `put-object-acl`.

## AWS CLI

Perintah berikut memberikan `full control` kepada dua AWS pengguna (`user1@example.com` dan `user2@example.com`) dan `read` izin untuk semua orang:

```
aws s3api put-object-acl --bucket MyBucket --key file.txt --grant-full-control
emailaddress=user1@example.com,emailaddress=user2@example.com --grant-read
uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Lihat <http://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketPUTacl.html> untuk detail tentang ACL kustom (perintah `s3api ACL`, seperti `put-object-acl`, menggunakan notasi argumen singkatan yang sama).

- Untuk detail API, lihat [PutObjectAcl](#) di Referensi AWS CLI Perintah.

## `put-object-legal-hold`

Contoh kode berikut menunjukkan cara menggunakan `put-object-legal-hold`.

### AWS CLI

Untuk menerapkan Penahanan Hukum pada suatu objek

`put-object-legal-hold` Contoh berikut menetapkan Penahanan Hukum pada objek `doc1.rtf`.

```
aws s3api put-object-legal-hold \
  --bucket my-bucket-with-object-lock \
  --key doc1.rtf \
  --legal-hold Status=ON
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutObjectLegalHold](#) di Referensi AWS CLI Perintah.

## `put-object-lock-configuration`

Contoh kode berikut menunjukkan cara menggunakan `put-object-lock-configuration`.

### AWS CLI

Untuk mengatur konfigurasi kunci objek pada ember

`put-object-lock-configuration` Contoh berikut menetapkan kunci objek 50 hari pada bucket yang ditentukan.

```
aws s3api put-object-lock-configuration \  
  --bucket my-bucket-with-object-lock \  
  --object-lock-configuration '{ "ObjectLockEnabled": "Enabled", "Rule":  
{ "DefaultRetention": { "Mode": "COMPLIANCE", "Days": 50 } } }'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutObjectLockConfiguration](#) di Referensi AWS CLI Perintah.

## **put-object-retention**

Contoh kode berikut menunjukkan cara menggunakan `put-object-retention`.

### AWS CLI

Untuk mengatur konfigurasi retensi objek untuk objek

`put-object-retention` Contoh berikut menetapkan konfigurasi retensi objek untuk objek yang ditentukan hingga 2025-01-01.

```
aws s3api put-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf \  
  --retention '{ "Mode": "GOVERNANCE", "RetainUntilDate": "2025-01-01T00:00:00" }'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutObjectRetention](#) di Referensi AWS CLI Perintah.

## **put-object-tagging**

Contoh kode berikut menunjukkan cara menggunakan `put-object-tagging`.

### AWS CLI

Untuk mengatur tag pada objek

`put-object-tagging` Contoh berikut menetapkan tag dengan kunci `designation` dan nilai `confidential` pada objek tertentu.

```
aws s3api put-object-tagging \  
  --bucket my-bucket \  
  --key doc1.rtf \  
  --tagging '{"TagSet": [{ "Key": "designation", "Value": "confidential" }]}'
```

Perintah ini tidak menghasilkan output.

`put-object-tagging` Contoh berikut menetapkan beberapa tag set pada objek tertentu.

```
aws s3api put-object-tagging \  
  --bucket my-bucket-example \  
  --key doc3.rtf \  
  --tagging '{"TagSet": [{ "Key": "designation", "Value": "confidential" },  
  { "Key": "department", "Value": "finance" }, { "Key": "team", "Value":  
  "payroll" } ]}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutObjectTagging](#) di Referensi AWS CLI Perintah.

## put-object

Contoh kode berikut menunjukkan cara menggunakan `put-object`.

### AWS CLI

Contoh berikut menggunakan `put-object` perintah untuk mengunggah objek ke Amazon S3:

```
aws s3api put-object --bucket text-content --key dir-1/my_images.tar.bz2 --body  
my_images.tar.bz2
```

Contoh berikut menunjukkan unggahan file video (File video ditentukan menggunakan sintaks sistem file Windows. ):

```
aws s3api put-object --bucket text-content --key dir-1/big-video-file.mp4 --body e:  
\media\videos\f-sharp-3-data-services.mp4
```

Untuk informasi selengkapnya tentang mengunggah objek, lihat [Mengunggah Objek](#) di Panduan Pengembang Amazon S3.

- Untuk detail API, lihat [PutObject](#) di Referensi AWS CLI Perintah.

## put-public-access-block

Contoh kode berikut menunjukkan cara menggunakan `put-public-access-block`.

### AWS CLI

Untuk menyetel konfigurasi blokir akses publik untuk bucket

`put-public-access-block` Contoh berikut menetapkan konfigurasi akses publik blok restriktif untuk bucket yang ditentukan.

```
aws s3api put-public-access-block \  
  --bucket my-bucket \  
  --public-access-block-configuration  
  "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutPublicAccessBlock](#) di Referensi AWS CLI Perintah.

## rb

Contoh kode berikut menunjukkan cara menggunakan `rb`.

### AWS CLI

Contoh 1: Hapus ember

`rb` Perintah berikut menghapus ember. Dalam contoh ini, bucket pengguna adalah `mybucket`. Perhatikan bahwa ember harus kosong untuk menghapus:

```
aws s3 rb s3://mybucket
```

Output:

```
remove_bucket: mybucket
```

Contoh 2: Paksa menghapus ember

`rb` Perintah berikut menggunakan `--force` parameter untuk pertama-tama menghapus semua objek di ember dan kemudian menghapus ember itu sendiri. Dalam contoh ini, bucket pengguna adalah `mybucket` dan objek di dalamnya `mybucket` adalah `test1.txt` dan `test2.txt`:

```
aws s3 rb s3://mybucket \  
--force
```

### Output:

```
delete: s3://mybucket/test1.txt  
delete: s3://mybucket/test2.txt  
remove_bucket: mybucket
```

- Untuk detail API, lihat [Rb](#) di Referensi AWS CLI Perintah.

## restore-object

Contoh kode berikut menunjukkan cara menggunakan `restore-object`.

### AWS CLI

Untuk membuat permintaan pemulihan untuk objek

`restore-object` Contoh berikut mengembalikan objek Amazon S3 Glacier yang ditentukan untuk bucket selama 10 hari. `my-glacier-bucket`

```
aws s3api restore-object \  
--bucket my-glacier-bucket \  
--key doc1.rtf \  
--restore-request Days=10
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RestoreObject](#) di Referensi AWS CLI Perintah.

## rm

Contoh kode berikut menunjukkan cara menggunakan `rm`.

### AWS CLI

Contoh 1: Hapus objek S3

`rm` Perintah berikut menghapus objek s3 tunggal:



```
aws s3 rm s3://mybucket/test2.txt
```

Output:

```
delete: s3://mybucket/test2.txt
```

Contoh 2: Hapus semua konten dalam ember

`rm`Perintah berikut secara rekursif menghapus semua objek di bawah bucket dan awalan tertentu ketika diteruskan dengan parameter. `--recursive` Dalam contoh ini, ember `mybucket` berisi objek `test1.txt` dan `test2.txt`:

```
aws s3 rm s3://mybucket \  
--recursive
```

Output:

```
delete: s3://mybucket/test1.txt  
delete: s3://mybucket/test2.txt
```

Contoh 3: Hapus semua konten dalam ember, kecuali file ``.jpg``

`rm`Perintah berikut secara rekursif menghapus semua objek di bawah bucket dan awalan tertentu ketika diteruskan dengan parameter `--recursive` sementara mengecualikan beberapa objek dengan menggunakan parameter. `--exclude` Dalam contoh ini, ember `mybucket` memiliki objek `test1.txt` dan `test2.jpg`:

```
aws s3 rm s3://mybucket/ \  
--recursive \  
--exclude "*.jpg"
```

Output:

```
delete: s3://mybucket/test1.txt
```

Contoh 4: Hapus semua konten dalam ember, kecuali objek di bawah awalan yang ditentukan

`rm`Perintah berikut secara rekursif menghapus semua objek di bawah bucket dan awalan tertentu ketika diteruskan dengan parameter `--recursive` sementara mengecualikan semua objek di

bawah awalan tertentu dengan menggunakan parameter. `--exclude` Dalam contoh ini, ember `mybucket` memiliki objek `test1.txt` dan `another/test.txt`:

```
aws s3 rm s3://mybucket/ \  
  --recursive \  
  --exclude "another/*"
```

Output:

```
delete: s3://mybucket/test1.txt
```

Contoh 5: Hapus objek dari titik akses S3

`rm` Perintah berikut menghapus satu objek (`mykey`) dari titik akses (`myaccesspoint`). :: `rm` Perintah berikut menghapus satu objek (`mykey`) dari titik akses (`myaccesspoint`).

```
aws s3 rm s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
```

Output:

```
delete: s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
```

- Untuk detail API, lihat [Rm](#) di Referensi AWS CLI Perintah.

## **select-object-content**

Contoh kode berikut menunjukkan cara menggunakan `select-object-content`.

### AWS CLI

Untuk memfilter konten objek Amazon S3 berdasarkan pernyataan SQL

`select-object-content` Contoh berikut menyaring objek `my-data-file.csv` dengan pernyataan SQL tertentu dan mengirimkan output ke file.

```
aws s3api select-object-content \  
  --bucket my-bucket \  
  --key my-data-file.csv \  
  --expression "select * from s3object limit 100" \  
  --expression-type 'SQL' \  
  --input-serialization '{"CSV": {}, "CompressionType": "NONE"}' \  
  --output-file my-output-file.csv
```

```
--output-serialization '{"CSV": {}}' "output.csv"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SelectObjectContent](#) di Referensi AWS CLI Perintah.

## sync

Contoh kode berikut menunjukkan cara menggunakan `sync`.

### AWS CLI

Contoh 1: Sinkronkan semua objek lokal ke bucket yang ditentukan

`sync` Perintah berikut menyinkronkan objek dari direktori lokal ke awalan dan bucket yang ditentukan dengan mengunggah file lokal ke S3. File lokal akan memerlukan pengunggahan jika ukuran file lokal berbeda dari ukuran objek S3, waktu modifikasi terakhir dari file lokal lebih baru dari waktu modifikasi terakhir dari objek S3, atau file lokal tidak ada di bawah ember dan awalan yang ditentukan. Dalam contoh ini, pengguna menyinkronkan bucket `mybucket` ke direktori lokal saat ini. Direktori lokal saat ini berisi file `test.txt` dan `filetest2.txt`. Ember tidak `mybucket` berisi benda.

```
aws s3 sync . s3://mybucket
```

Output:

```
upload: test.txt to s3://mybucket/test.txt
upload: test2.txt to s3://mybucket/test2.txt
```

Contoh 2: Sinkronkan semua objek S3 dari bucket S3 yang ditentukan ke bucket lain

`sync` Perintah berikut menyinkronkan objek di bawah awalan dan bucket yang ditentukan ke objek di bawah awalan dan bucket lain yang ditentukan dengan menyalin objek S3. Objek S3 akan memerlukan penyalinan jika ukuran dua objek S3 berbeda, waktu modifikasi terakhir dari sumber lebih baru dari waktu modifikasi terakhir dari tujuan, atau objek S3 tidak ada di bawah bucket dan tujuan awalan yang ditentukan.

Dalam contoh ini, pengguna menyinkronkan bucket `mybucket` ke `bucketmybucket2`. Ember `mybucket` berisi benda-benda `test.txt` dan `test2.txt`. Ember tidak `mybucket2` berisi benda:

```
aws s3 sync s3://mybucket s3://mybucket2
```

### Output:

```
copy: s3://mybucket/test.txt to s3://mybucket2/test.txt
copy: s3://mybucket/test2.txt to s3://mybucket2/test2.txt
```

### Contoh 3: Sinkronkan semua objek S3 dari bucket S3 yang ditentukan ke direktori lokal

`sync` Perintah berikut menyinkronkan file dari bucket S3 yang ditentukan ke direktori lokal dengan mengunduh objek S3. Objek S3 akan memerlukan pengunduhan jika ukuran objek S3 berbeda dari ukuran file lokal, waktu modifikasi terakhir dari objek S3 lebih baru dari waktu modifikasi terakhir dari file lokal, atau objek S3 tidak ada di direktori lokal. Perhatikan bahwa ketika objek diunduh dari S3, waktu modifikasi terakhir dari file lokal diubah ke waktu modifikasi terakhir dari objek S3. Dalam contoh ini, pengguna menyinkronkan bucket `mybucket` ke direktori lokal saat ini. Ember `mybucket` berisi benda-benda `test.txt` dan `test2.txt`. Direktori lokal saat ini tidak memiliki file:

```
aws s3 sync s3://mybucket .
```

### Output:

```
download: s3://mybucket/test.txt to test.txt
download: s3://mybucket/test2.txt to test2.txt
```

### Contoh 4: Sinkronkan semua objek lokal ke bucket yang ditentukan dan hapus semua file yang tidak cocok

`sync` Perintah berikut menyinkronkan objek di bawah awalan dan bucket yang ditentukan ke file di direktori lokal dengan mengunggah file lokal ke S3. Karena `--delete` parameternya, file apa pun yang ada di bawah awalan dan bucket yang ditentukan tetapi tidak ada di direktori lokal akan dihapus. Dalam contoh ini, pengguna menyinkronkan bucket `mybucket` ke direktori lokal saat ini. Direktori lokal saat ini berisi file `test.txt` dan `filetest2.txt`. Ember `mybucket` berisi objek `test3.txt`:

```
aws s3 sync . s3://mybucket \
  --delete
```

**Output:**

```
upload: test.txt to s3://mybucket/test.txt
upload: test2.txt to s3://mybucket/test2.txt
delete: s3://mybucket/test3.txt
```

Contoh 5: Sinkronkan semua objek lokal ke bucket yang ditentukan kecuali file ``.jpg``

`sync` Perintah berikut menyinkronkan objek di bawah awalan dan bucket yang ditentukan ke file di direktori lokal dengan mengunggah file lokal ke S3. Karena `--exclude` parameternya, semua file yang cocok dengan pola yang ada di S3 dan secara lokal akan dikecualikan dari sinkronisasi. Dalam contoh ini, pengguna menyinkronkan bucket `mybucket` ke direktori lokal saat ini. Direktori lokal saat ini berisi file `test.jpg` dan `filetest2.txt`. Ember `mybucket` berisi objek dengan `test.jpg` ukuran yang berbeda dari lokal `test.jpg`:

```
aws s3 sync . s3://mybucket \
  --exclude "*.jpg"
```

**Output:**

```
upload: test2.txt to s3://mybucket/test2.txt
```

Contoh 6: Sinkronkan semua objek lokal ke bucket yang ditentukan kecuali file ``.jpg``

`sync` Perintah berikut menyinkronkan file di bawah direktori lokal ke objek di bawah awalan dan ember tertentu dengan mengunduh objek S3. Contoh ini menggunakan flag `--exclude` parameter untuk mengecualikan direktori tertentu dan awalan S3 dari perintah. `sync` Dalam contoh ini, pengguna menyinkronkan direktori lokal saat ini ke bucket `mybucket`. Direktori lokal saat ini berisi file `test.txt` dan file `another/test2.txt`. Ember `mybucket` berisi benda-benda `another/test5.txt` dan `test1.txt`:

```
aws s3 sync s3://mybucket/ . \
  --exclude "*another/*"
```

**Output:**

```
download: s3://mybucket/test1.txt to test1.txt
```

Contoh 7: Sinkronkan semua objek antar ember di berbagai wilayah

`sync` Perintah berikut menyinkronkan file antara dua bucket di wilayah yang berbeda:

```
aws s3 sync s3://my-us-west-2-bucket s3://my-us-east-1-bucket \  
  --source-region us-west-2 \  
  --region us-east-1
```

Output:

```
download: s3://my-us-west-2-bucket/test1.txt to s3://my-us-east-1-bucket/test1.txt
```

Contoh 8: Sinkronkan ke titik akses S3

`sync` Perintah berikut menyinkronkan direktori saat ini ke titik akses (`myaccesspoint`):

```
aws s3 sync . s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
```

Output:

```
upload: test.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/  
myaccesspoint/test.txt  
upload: test2.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/  
myaccesspoint/test2.txt
```

- Untuk detail API, lihat [Sinkronisasi](#) dalam Referensi AWS CLI Perintah.

## upload-part-copy

Contoh kode berikut menunjukkan cara menggunakan `upload-part-copy`.

### AWS CLI

Untuk meng-upload bagian dari objek dengan menyalin data dari objek yang ada sebagai sumber data

`upload-part-copy` Contoh berikut mengunggah bagian dengan menyalin data dari objek yang ada sebagai sumber data.

```
aws s3api upload-part-copy \  
  --bucket my-bucket \  
  --source-object test.txt
```

```
--key "Map_Data_June.mp4" \
--copy-source "my-bucket/copy_of_Map_Data_June.mp4" \
--part-number 1 \
--upload-id
"bq0tdE1CDpWQYRPLHuNG50xAT6pA5D.m_RiBy0gg0H6b13pVRY7QjvL1f75iFdJqp_2wztk5hvpUM2SesXgrzbehG5"
```

Output:

```
{
  "CopyPartResult": {
    "LastModified": "2019-12-13T23:16:03.000Z",
    "ETag": "\"711470fc377698c393d94aed6305e245\""
  }
}
```

- Untuk detail API, lihat [UploadPartCopy](#) di Referensi AWS CLI Perintah.

## upload-part

Contoh kode berikut menunjukkan cara menggunakan `upload-part`.

### AWS CLI

Perintah berikut mengunggah bagian pertama dalam unggahan multibagian yang dimulai dengan perintah: `create-multipart-upload`

```
aws s3api upload-part --bucket my-bucket --key 'multipart/01' --part-number 1 --body
part01 --upload-id
"dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZ1jF.Yxwh6XG7WfS2vC4to6HiV6Yj1x.cph0gtNBtJ8P3URC"
```

`body` opsi ini mengambil nama atau jalur file lokal untuk diunggah (jangan gunakan awalan `file://`). Ukuran bagian minimum adalah 5 MB. Upload ID dikembalikan oleh `create-multipart-upload` dan juga dapat diambil dengan `list-multipart-uploads`. Bucket dan kunci ditentukan saat Anda membuat unggahan multipart.

Output:

```
{
  "ETag": "\"e868e0f4719e394144ef36531ee6824c\""
}
```

Simpan nilai ETag dari setiap bagian untuk nanti. Mereka diminta untuk menyelesaikan unggahan multipart.

- Untuk detail API, lihat [UploadPart](#) di Referensi AWS CLI Perintah.

## website

Contoh kode berikut menunjukkan cara menggunakan `website`.

### AWS CLI

Konfigurasi bucket S3 sebagai situs web statis

Perintah berikut mengonfigurasi bucket bernama `my-bucket` sebagai situs web statis. Opsi dokumen indeks menentukan file di mana pengunjung `my-bucket` akan diarahkan ketika mereka menavigasi ke URL situs web. Dalam hal ini, ember berada di wilayah `us-west-2`, sehingga situs akan muncul di `http://my-bucket.s3-website-us-west-2.amazonaws.com`

Semua file dalam ember yang muncul di situs statis harus dikonfigurasi untuk memungkinkan pengunjung membukanya. Izin file dikonfigurasi secara terpisah dari konfigurasi situs web bucket.

```
aws s3 website s3://my-bucket/ \  
  --index-document index.html \  
  --error-document error.html
```

Untuk informasi tentang hosting situs web statis di Amazon S3, lihat [Hosting Situs Web Statis di Panduan](#) Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [Situs Web](#) di Referensi AWS CLI Perintah.

## Contoh Kontrol Amazon S3 menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum AWS Command Line Interface dengan menggunakan Kontrol Amazon S3.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.



Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **create-access-point**

Contoh kode berikut menunjukkan cara menggunakan `create-access-point`.

## AWS CLI

Untuk membuat titik akses

`create-access-point` Contoh berikut membuat titik akses yang diberi nama `finance-ap` untuk bucket `business-records` di akun `123456789012`. Sebelum menjalankan contoh ini, ganti nama titik akses, nama bucket, dan nomor akun dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control create-access-point \  
  --account-id 123456789012 \  
  --bucket business-records \  
  --name finance-ap
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Titik Akses](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [CreateAccessPoint](#) di Referensi AWS CLI Perintah.

### **create-job**

Contoh kode berikut menunjukkan cara menggunakan `create-job`.

## AWS CLI

Untuk membuat pekerjaan operasi batch Amazon S3

`create-job` Contoh berikut membuat pekerjaan operasi batch Amazon S3 untuk menandai objek sebagai `confidential` in the bucket `employee-records`

```
aws s3control create-job \
  --account-id 123456789012 \
  --operation '{"S3PutObjectTagging": { "TagSet": [{"Key":"confidential",
"Value":"true"}] }}' \
  --report '{"Bucket":"arn:aws:s3:::employee-records-logs","Prefix":"batch-op-
create-job",
"Format":"Report_CSV_20180820","Enabled":true,"ReportScope":"AllTasks"}' \
  --manifest '{"Spec":{"Format":"S3BatchOperations_CSV_20180820","Fields":
["Bucket","Key"]},"Location":{"ObjectArn":"arn:aws:s3:::employee-records-logs/inv-
report/7a6a9be4-072c-407e-85a2-
ec3e982f773e.csv"},"ETag":"69f52a4e9f797e987155d9c8f5880897"}'} \
  --priority 42 \
  --role-arn arn:aws:iam::123456789012:role/S3BatchJobRole
```

Output:

```
{
  "JobId": "93735294-df46-44d5-8638-6356f335324e"
}
```

- Untuk detail API, lihat [CreateJob](#) di Referensi AWS CLI Perintah.

## delete-access-point-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-access-point-policy`.

### AWS CLI

Untuk menghapus kebijakan titik akses

`delete-access-point-policy` Contoh berikut menghapus kebijakan titik akses dari titik akses yang disebutkan `finance-ap` di akun 123456789012. Sebelum menjalankan contoh ini, ganti nama titik akses dan nomor akun dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control delete-access-point-policy \
  --account-id 123456789012 \
  --name finance-ap
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Akses Data dengan Titik Akses Amazon S3](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [DeleteAccessPointPolicy](#) di Referensi AWS CLI Perintah.

## **delete-access-point**

Contoh kode berikut menunjukkan cara menggunakan `delete-access-point`.

### AWS CLI

Untuk menghapus titik akses

`delete-access-point` Contoh berikut menghapus titik akses bernama `finance-ap` dalam akun `123456789012`. Sebelum menjalankan contoh ini, ganti nama titik akses dan nomor akun dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control delete-access-point \  
  --account-id 123456789012 \  
  --name finance-ap
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Akses Data dengan Titik Akses Amazon S3](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [DeleteAccessPoint](#) di Referensi AWS CLI Perintah.

## **delete-public-access-block**

Contoh kode berikut menunjukkan cara menggunakan `delete-public-access-block`.

### AWS CLI

Untuk menghapus blokir pengaturan akses publik untuk akun

`delete-public-access-block` Contoh berikut menghapus blokir pengaturan akses publik untuk akun yang ditentukan.

```
aws s3control delete-public-access-block \  
  --account-id 123456789012 \  
  --bucket-name my-bucket \  
  --public-access-block-id my-public-access-block-id
```

```
--account-id 123456789012
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePublicAccessBlock](#) di Referensi AWS CLI Perintah.

## describe-job

Contoh kode berikut menunjukkan cara menggunakan `describe-job`.

### AWS CLI

Untuk menggambarkan pekerjaan operasi batch Amazon S3

Berikut ini `describe-job` menyediakan parameter konfigurasi dan status untuk pekerjaan operasi batch yang ditentukan.

```
aws s3control describe-job \  
  --account-id 123456789012 \  
  --job-id 93735294-df46-44d5-8638-6356f335324e
```

Output:

```
{  
  "Job": {  
    "TerminationDate": "2019-10-03T21:49:53.944Z",  
    "JobId": "93735294-df46-44d5-8638-6356f335324e",  
    "FailureReasons": [],  
    "Manifest": {  
      "Spec": {  
        "Fields": [  
          "Bucket",  
          "Key"  
        ],  
        "Format": "S3BatchOperations_CSV_20180820"  
      },  
      "Location": {  
        "ETag": "69f52a4e9f797e987155d9c8f5880897",  
        "ObjectArn": "arn:aws:s3:::employee-records-logs/inv-report/7a6a9be4-072c-407e-85a2-ec3e982f773e.csv"  
      }  
    },  
  },  
}
```

```

"Operation": {
  "S3PutObjectTagging": {
    "TagSet": [
      {
        "Value": "true",
        "Key": "confidential"
      }
    ]
  }
},
"RoleArn": "arn:aws:iam::123456789012:role/S3BatchJobRole",
"ProgressSummary": {
  "TotalNumberOfTasks": 8,
  "NumberOfTasksFailed": 0,
  "NumberOfTasksSucceeded": 8
},
"Priority": 42,
"Report": {
  "ReportScope": "AllTasks",
  "Format": "Report_CSV_20180820",
  "Enabled": true,
  "Prefix": "batch-op-create-job",
  "Bucket": "arn:aws:s3:::employee-records-logs"
},
"JobArn": "arn:aws:s3:us-west-2:123456789012:job/93735294-
df46-44d5-8638-6356f335324e",
"CreationTime": "2019-10-03T21:48:48.048Z",
"Status": "Complete"
}
}

```

- Untuk detail API, lihat [DescribeJob](#) di Referensi AWS CLI Perintah.

## get-access-point-policy-status

Contoh kode berikut menunjukkan cara menggunakan `get-access-point-policy-status`.

### AWS CLI

Untuk mengambil status kebijakan jalur akses

`get-access-point-policy-status` Contoh berikut mengambil status kebijakan titik akses untuk titik akses yang disebutkan `finance-ap` di akun `123456789012`. Status kebijakan

titik akses menunjukkan apakah kebijakan titik akses mengizinkan akses publik. Sebelum menjalankan contoh ini, ganti nama titik akses dan nomor akun dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control get-access-point-policy-status \  
  --account-id 123456789012 \  
  --name finance-ap
```

Output:

```
{  
  "PolicyStatus": {  
    "IsPublic": false  
  }  
}
```

Untuk informasi selengkapnya tentang kapan kebijakan jalur akses dianggap publik, lihat [Arti "Publik"](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [GetAccessPointPolicyStatus](#) di Referensi AWS CLI Perintah.

## get-access-point-policy

Contoh kode berikut menunjukkan cara menggunakan `get-access-point-policy`.

AWS CLI

Untuk mengambil kebijakan titik akses

`get-access-point-policy` Contoh berikut mengambil kebijakan titik akses dari titik akses yang disebutkan `finance-ap` di akun `123456789012`. Sebelum menjalankan contoh ini, ganti nama titik akses dan nomor akun dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control get-access-point-policy \  
  --account-id 123456789012 \  
  --name finance-ap
```

Output:

```
{
```

```
"Policy": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": [\n    {\n      \"Effect\": \"Allow\",\n      \"Principal\": {\n        \"AWS\": \"arn:aws:iam::123456789012:role/Admin\"\n      },\n      \"Action\": \"s3:GetObject\",\n      \"Resource\": \"arn:aws:s3:us-west-2:123456789012:accesspoint/finance-ap/object/records/*\"\n    }\n  ]\n}"
```

Untuk informasi selengkapnya, lihat [Mengelola Akses Data dengan Titik Akses Amazon S3](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [GetAccessPointPolicy](#) di Referensi AWS CLI Perintah.

## get-access-point

Contoh kode berikut menunjukkan cara menggunakan `get-access-point`.

### AWS CLI

Untuk mengambil detail konfigurasi titik akses

`get-access-point` Contoh berikut mengambil rincian konfigurasi untuk titik akses yang disebutkan `finance-ap` dalam akun `123456789012`. Sebelum menjalankan contoh ini, ganti nama titik akses dan nomor akun dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control get-access-point \
  --account-id 123456789012 \
  --name finance-ap
```

Output:

```
{
  "Name": "finance-ap",
  "Bucket": "business-records",
  "NetworkOrigin": "Internet",
  "PublicAccessBlockConfiguration": {
    "BlockPublicAcls": false,
    "IgnorePublicAcls": false,
    "BlockPublicPolicy": false,
    "RestrictPublicBuckets": false
  },
  "CreationDate": "2020-01-01T00:00:00Z"
}
```

Untuk informasi selengkapnya, lihat [Mengelola Akses Data dengan Titik Akses Amazon S3](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [GetAccessPoint](#) di Referensi AWS CLI Perintah.

## get-multi-region-access-point-routes

Contoh kode berikut menunjukkan cara menggunakan `get-multi-region-access-point-routes`.

### AWS CLI

Untuk menanyakan konfigurasi rute Titik Akses Multi-Wilayah saat ini

`get-multi-region-access-point-routes` Contoh berikut mengembalikan konfigurasi routing saat ini untuk Titik Akses Multi-Region yang ditentukan.

```
aws s3control get-multi-region-access-point-routes \
  --region Region \
  --account-id 111122223333 \
  --mrap MultiRegionAccessPoint_ARN
```

### Output:

```
{
  "Mrap": "arn:aws:s3:::111122223333:accesspoint/0000000000000000.mrap",
  "Routes": [
    {
      "Bucket": "DOC-EXAMPLE-BUCKET-1",
      "Region": "ap-southeast-2",
      "TrafficDialPercentage": 100
    },
    {
      "Bucket": "DOC-EXAMPLE-BUCKET-2",
      "Region": "us-west-1",
      "TrafficDialPercentage": 0
    }
  ]
}
```

- Untuk detail API, lihat [GetMultiRegionAccessPointRoutes](#) di Referensi AWS CLI Perintah.



## get-public-access-block

Contoh kode berikut menunjukkan cara menggunakan `get-public-access-block`.

### AWS CLI

Untuk mencantumkan setelan akses blokir publik untuk akun

`get-public-access-block` Contoh berikut menampilkan pengaturan blok akses publik untuk akun yang ditentukan.

```
aws s3control get-public-access-block \  
  --account-id 123456789012
```

Output:

```
{  
  "PublicAccessBlockConfiguration": {  
    "BlockPublicPolicy": true,  
    "RestrictPublicBuckets": true,  
    "IgnorePublicAcls": true,  
    "BlockPublicAcls": true  
  }  
}
```

- Untuk detail API, lihat [GetPublicAccessBlock](#) di Referensi AWS CLI Perintah.

## list-access-points

Contoh kode berikut menunjukkan cara menggunakan `list-access-points`.

### AWS CLI

Contoh 1: Untuk mengambil daftar semua titik akses untuk akun

`list-access-points` Contoh berikut menampilkan daftar semua titik akses yang dilampirkan ke bucket yang dimiliki oleh akun 123456789012.

```
aws s3control list-access-points \  
  --account-id 123456789012
```

**Output:**

```
{
  "AccessPointList": [
    {
      "Name": "finance-ap",
      "NetworkOrigin": "Internet",
      "Bucket": "business-records"
    },
    {
      "Name": "managers-ap",
      "NetworkOrigin": "Internet",
      "Bucket": "business-records"
    },
    {
      "Name": "private-network-ap",
      "NetworkOrigin": "VPC",
      "VpcConfiguration": {
        "VpcId": "1a2b3c"
      },
      "Bucket": "business-records"
    },
    {
      "Name": "customer-ap",
      "NetworkOrigin": "Internet",
      "Bucket": "external-docs"
    },
    {
      "Name": "public-ap",
      "NetworkOrigin": "Internet",
      "Bucket": "external-docs"
    }
  ]
}
```

Contoh 2: Untuk mengambil daftar semua titik akses untuk ember

`list-access-points` Contoh berikut mengambil daftar semua titik akses yang dilampirkan ke bucket yang `external-docs` dimiliki oleh akun 123456789012.

```
aws s3control list-access-points \
  --account-id 123456789012 \
```

```
--bucket external-docs
```

Output:

```
{
  "AccessPointList": [
    {
      "Name": "customer-ap",
      "NetworkOrigin": "Internet",
      "Bucket": "external-docs"
    },
    {
      "Name": "public-ap",
      "NetworkOrigin": "Internet",
      "Bucket": "external-docs"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Akses Data dengan Titik Akses Amazon S3](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [ListAccessPoints](#) di Referensi AWS CLI Perintah.

## list-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-jobs`.

### AWS CLI

Untuk membuat daftar akun, pekerjaan operasi batch Amazon S3

`list-jobs` Contoh berikut mencantumkan semua pekerjaan operasi batch terbaru untuk akun yang ditentukan.

```
aws s3control list-jobs \
  --account-id 123456789012
```

Output:

```
{
  "Jobs": [
```

```

    {
      "Operation": "S3PutObjectTagging",
      "ProgressSummary": {
        "NumberOfTasksFailed": 0,
        "NumberOfTasksSucceeded": 8,
        "TotalNumberOfTasks": 8
      },
      "CreationTime": "2019-10-03T21:48:48.048Z",
      "Status": "Complete",
      "JobId": "93735294-df46-44d5-8638-6356f335324e",
      "Priority": 42
    },
    {
      "Operation": "S3PutObjectTagging",
      "ProgressSummary": {
        "NumberOfTasksFailed": 0,
        "NumberOfTasksSucceeded": 0,
        "TotalNumberOfTasks": 0
      },
      "CreationTime": "2019-10-03T21:46:07.084Z",
      "Status": "Failed",
      "JobId": "3f3c7619-02d3-4779-97f6-1d98dd313108",
      "Priority": 42
    },
  ],
}

```

- Untuk detail API, lihat [ListJobs](#) di Referensi AWS CLI Perintah.

## put-access-point-policy

Contoh kode berikut menunjukkan cara menggunakan `put-access-point-policy`.

### AWS CLI

Untuk menetapkan kebijakan titik akses

`put-access-point-policy` Contoh berikut menempatkan kebijakan titik akses yang ditentukan untuk titik akses `finance-ap` di akun `123456789012`. Jika titik akses `finance-ap` sudah memiliki kebijakan, perintah ini menggantikan kebijakan yang ada dengan yang ditentukan dalam perintah ini. Sebelum menjalankan contoh ini, ganti nomor akun, nama titik akses, dan pernyataan kebijakan dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control put-access-point-policy \  
  --account-id 123456789012 \  
  --name finance-ap \  
  --policy file://ap-policy.json
```

Isi dari ap-policy.json:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "arn:aws:iam::123456789012:user/Alice"  
      },  
      "Action": "s3:GetObject",  
      "Resource": "arn:aws:s3:us-west-2:123456789012:accesspoint/finance-ap/  
object/Alice/*"  
    }  
  ]  
}
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Akses Data dengan Titik Akses Amazon S3](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

- Untuk detail API, lihat [PutAccessPointPolicy](#) di Referensi AWS CLI Perintah.

## put-public-access-block

Contoh kode berikut menunjukkan cara menggunakan `put-public-access-block`.

### AWS CLI

Mengedit blokir setelan akses publik untuk akun

`put-public-access-block` Contoh berikut mengaktifkan semua pengaturan blokir akses publik `true` untuk akun yang ditentukan.

```
aws s3control put-public-access-block \  
  --account-id 123456789012 \  
  --
```

```
--public-access-block-configuration '{"BlockPublicAcls": true,
"IgnorePublicAcls": true, "BlockPublicPolicy": true, "RestrictPublicBuckets":
true}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutPublicAccessBlock](#) di Referensi AWS CLI Perintah.

## submit-multi-region-access-point-routes

Contoh kode berikut menunjukkan cara menggunakan `submit-multi-region-access-point-routes`.

### AWS CLI

Untuk memperbarui konfigurasi perutean Titik Akses Multi-Wilayah

`submit-multi-region-access-point-routes` Contoh berikut memperbarui status perutean `DOC-EXAMPLE-BUCKET-1` dan `DOC-EXAMPLE-BUCKET-2` di `ap-southeast-2` Wilayah untuk Titik Akses Multi-Wilayah Anda.

```
aws s3control submit-multi-region-access-point-routes \
  --region ap-southeast-2 \
  --account-id 111122223333 \
  --mrap MultiRegionAccessPoint_ARN \
  --route-updates Bucket=DOC-EXAMPLE-BUCKET-1,TrafficDialPercentage=100
  Bucket=DOC-EXAMPLE-BUCKET-2,TrafficDialPercentage=0
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SubmitMultiRegionAccessPointRoutes](#) di Referensi AWS CLI Perintah.

## update-job-priority

Contoh kode berikut menunjukkan cara menggunakan `update-job-priority`.

### AWS CLI

Untuk memperbarui prioritas pekerjaan pekerjaan operasi batch Amazon S3

`update-job-priority` Contoh berikut memperbarui pekerjaan yang ditentukan ke prioritas baru.

```
aws s3control update-job-priority \  
  --account-id 123456789012 \  
  --job-id 8d9a18fe-c303-4d39-8ccc-860d372da386 \  
  --priority 52
```

Output:

```
{  
  "JobId": "8d9a18fe-c303-4d39-8ccc-860d372da386",  
  "Priority": 52  
}
```

- Untuk detail API, lihat [UpdateJobPriority](#) di Referensi AWS CLI Perintah.

## update-job-status

Contoh kode berikut menunjukkan cara menggunakan `update-job-status`.

### AWS CLI

Untuk memperbarui status pekerjaan operasi batch Amazon S3

`update-job-status` Contoh berikut membatalkan pekerjaan yang ditentukan yang menunggu persetujuan.

```
aws s3control update-job-status \  
  --account-id 123456789012 \  
  --job-id 8d9a18fe-c303-4d39-8ccc-860d372da386 \  
  --requested-job-status Cancelled
```

Output:

```
{  
  "Status": "Cancelled",  
  "JobId": "8d9a18fe-c303-4d39-8ccc-860d372da386"  
}
```

`update-job-status` Contoh berikut mengkonfirmasi dan menjalankan yang ditentukan yang menunggu persetujuan.

```
aws s3control update-job-status \  
  --account-id 123456789012 \  
  --job-id 5782949f-3301-4fb3-be34-8d5bab54dbca \  
  --requested-job-status Ready
```

Output::

```
{  
  "Status": "Ready",  
  "JobId": "5782949f-3301-4fb3-be34-8d5bab54dbca"  
}
```

update-job-status Contoh berikut membatalkan pekerjaan tertentu yang sedang berjalan.

```
aws s3control update-job-status \  
  --account-id 123456789012 \  
  --job-id 5782949f-3301-4fb3-be34-8d5bab54dbca \  
  --requested-job-status Cancelled
```

Output::

```
{  
  "Status": "Cancelling",  
  "JobId": "5782949f-3301-4fb3-be34-8d5bab54dbca"  
}
```

- Untuk detail API, lihat [UpdateJobStatus](#) di Referensi AWS CLI Perintah.

## Contoh S3 Glacier menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan gletser AWS Command Line Interface with S3.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.



## Topik

- [Tindakan](#)

## Tindakan

### **abort-multipart-upload**

Contoh kode berikut menunjukkan cara menggunakan `abort-multipart-upload`.

#### AWS CLI

Perintah berikut menghapus unggahan multipart yang sedang berlangsung ke vault bernama: `my-vault`

```
aws glacier abort-multipart-upload --account-id - --vault-name my-vault
--upload-id 19gaRezEXAMPLES6Ry5YYdqthH0C_kGRCT03L9yetr220UmPtBYKk-
0ssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ
```

Perintah ini tidak menghasilkan output apa pun. Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan. ID unggahan dikembalikan oleh `aws glacier initiate-multipart-upload` perintah dan juga dapat diperoleh dengan menggunakan `aws glacier list-multipart-uploads`.

Untuk informasi selengkapnya tentang unggahan multipart ke Amazon Glacier menggunakan CLI, lihat [Menggunakan Amazon Glacier AWS](#) di Panduan Pengguna CLI.AWS

- Untuk detail API, lihat [AbortMultipartUpload](#) di Referensi AWS CLI Perintah.

### **abort-vault-lock**

Contoh kode berikut menunjukkan cara menggunakan `abort-vault-lock`.

#### AWS CLI

Untuk membatalkan proses penguncian vault yang sedang berlangsung

`abort-vault-lock` Contoh berikut menghapus kebijakan kunci vault dari vault yang ditentukan dan mengatur ulang status kunci kunci vault menjadi tidak terkunci.

```
aws glacier abort-vault-lock \
```

```
--account-id - \  
--vault-name MyVaultName
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Batalkan Kunci Vault \(HAPUS kebijakan kunci\)](#) di Panduan Pengembang Amazon Glacier API.

- Untuk detail API, lihat [AbortVaultLock](#) di Referensi AWS CLI Perintah.

## add-tags-to-vault

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-vault`.

### AWS CLI

Perintah berikut menambahkan dua tag ke vault bernama `my-vault`:

```
aws glacier add-tags-to-vault --account-id - --vault-name my-vault --tags  
id=1234,date=july2015
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [AddTagsToVault](#) di Referensi AWS CLI Perintah.

## complete-multipart-upload

Contoh kode berikut menunjukkan cara menggunakan `complete-multipart-upload`.

### AWS CLI

Perintah berikut menyelesaikan unggahan multipart untuk arsip 3 MiB:

```
aws glacier complete-multipart-upload --archive-size 3145728 --checksum  
9628195fcdcbbe76cdde456d4646fa7de5f219fb39823836d81f0cc0e18aa67  
--upload-id 19gaRezEXAMPLES6Ry5YYdqthH0C_kGRCT03L9yetr220UmPtBYKk-  
0ssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ --account-id - --vault-name my-vault
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

ID unggahan dikembalikan oleh `aws glacier initiate-multipart-upload` perintah dan juga dapat diperoleh dengan menggunakan `aws glacier list-multipart-uploads`. Parameter checksum mengambil hash pohon SHA-256 dari arsip dalam heksadesimal.

Untuk informasi selengkapnya tentang unggahan multipart ke Amazon Glacier menggunakan AWS CLI, termasuk petunjuk cara menghitung hash pohon, lihat [Menggunakan Amazon Glacier di Panduan Pengguna CLI.AWS](#)

- Untuk detail API, lihat [CompleteMultipartUpload](#) di Referensi AWS CLI Perintah.

## **complete-vault-lock**

Contoh kode berikut menunjukkan cara menggunakan `complete-vault-lock`.

### AWS CLI

Untuk menyelesaikan proses penguncian vault yang sedang berlangsung

`complete-vault-lock` Contoh berikut melengkapi progres penguncian yang sedang berlangsung untuk vault yang ditentukan dan menyetel status kunci vault ke. Locked Anda mendapatkan nilai untuk `lock-id` parameter saat Anda menjalankan `initiate-lock-process`.

```
aws glacier complete-vault-lock \  
  --account-id - \  
  --vault-name MyVaultName \  
  --lock-id 9QZgEXAMPLEPhvL6xEXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Complete Vault Lock \(POST LockId\)](#) di Panduan Pengembang Amazon Glacier API.

- Untuk detail API, lihat [CompleteVaultLock](#) di Referensi AWS CLI Perintah.

## **create-vault**

Contoh kode berikut menunjukkan cara menggunakan `create-vault`.

### AWS CLI

Perintah berikut membuat vault baru bernama `my-vault`:

```
aws glacier create-vault --vault-name my-vault --account-id -
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [CreateVault](#) di Referensi AWS CLI Perintah.

## delete-archive

Contoh kode berikut menunjukkan cara menggunakan `delete-archive`.

### AWS CLI

Untuk menghapus arsip dari vault

`delete-archive` Contoh berikut menghapus arsip yang ditentukan dari `example_vault`.

```
aws glacier delete-archive \  
  --account-id 111122223333 \  
  --vault-name example_vault \  
  --archive-id Sc0u9ZP8yaWkmh-XG1IvAVprtLhaLCGnNwN15I5x9HqPIkX5mjc0DrId3Ln-  
  Gi_k2HzmlIDZUz117KSdVMdMXLuFwi9PJUitxW073edQ43eT1MwKH0pd9zVSAuV_XXZBVhKhyGhJ7w
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteArchive](#) di Referensi AWS CLI Perintah.

## delete-vault-access-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-vault-access-policy`.

### AWS CLI

Untuk menghapus kebijakan akses vault

`delete-vault-access-policy` Contoh berikut menghapus kebijakan akses untuk vault yang ditentukan.

```
aws glacier delete-vault-access-policy \  
  --account-id 111122223333 \  
  --vault-name example_vault
```

```
--vault-name example_vault
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteVaultAccessPolicy](#) di Referensi AWS CLI Perintah.

## delete-vault-notifications

Contoh kode berikut menunjukkan cara menggunakan `delete-vault-notifications`.

### AWS CLI

Untuk menghapus notifikasi SNS untuk vault

`delete-vault-notifications` Contoh berikut menghapus notifikasi yang dikirim oleh Amazon Simple Notification Service (Amazon SNS) untuk vault yang ditentukan.

```
aws glacier delete-vault-notifications \  
  --account-id 111122223333 \  
  --vault-name example_vault
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteVaultNotifications](#) di Referensi AWS CLI Perintah.

## delete-vault

Contoh kode berikut menunjukkan cara menggunakan `delete-vault`.

### AWS CLI

Perintah berikut menghapus vault bernama: `my-vault`

```
aws glacier delete-vault --vault-name my-vault --account-id -
```

Perintah ini tidak menghasilkan output apa pun. Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [DeleteVault](#) di Referensi AWS CLI Perintah.

## describe-job

Contoh kode berikut menunjukkan cara menggunakan `describe-job`.

### AWS CLI

Perintah berikut mengambil informasi tentang pekerjaan pengambilan inventaris pada vault bernama: `my-vault`

```
aws glacier describe-job --account-id - --vault-name my-  
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-  
R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW
```

Output:

```
{  
  "InventoryRetrievalParameters": {  
    "Format": "JSON"  
  },  
  "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",  
  "Completed": false,  
  "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-  
R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW",  
  "Action": "InventoryRetrieval",  
  "CreationDate": "2015-07-17T20:23:41.616Z",  
  "StatusCode": "InProgress"  
}
```

ID pekerjaan dapat ditemukan di output `aws glacier initiate-job` dan `aws glacier list-jobs`. Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [DescribeJob](#) di Referensi AWS CLI Perintah.

## describe-vault

Contoh kode berikut menunjukkan cara menggunakan `describe-vault`.

### AWS CLI

Perintah berikut mengambil data tentang vault bernama: `my-vault`

```
aws glacier describe-vault --vault-name my-vault --account-id -
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [DescribeVault](#) di Referensi AWS CLI Perintah.

## get-data-retrieval-policy

Contoh kode berikut menunjukkan cara menggunakan `get-data-retrieval-policy`.

### AWS CLI

Perintah berikut mendapatkan kebijakan pengambilan data untuk akun yang sedang digunakan:

```
aws glacier get-data-retrieval-policy --account-id -
```

Output:

```
{
  "Policy": {
    "Rules": [
      {
        "BytesPerHour": 10737418240,
        "Strategy": "BytesPerHour"
      }
    ]
  }
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [GetDataRetrievalPolicy](#) di Referensi AWS CLI Perintah.

## get-job-output

Contoh kode berikut menunjukkan cara menggunakan `get-job-output`.

## AWS CLI

Perintah berikut menyimpan output dari pekerjaan inventaris vault ke file di direktori saat ini bernama `output.json`:

```
aws glacier get-job-output --account-id - --vault-name my-  
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-  
R047Yc6FxsdBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW output.json
```

`job-id` Tersedia dalam output dari `aws glacier list-jobs`. Perhatikan bahwa nama file output adalah argumen posisi yang tidak diawali dengan nama opsi. Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Output:

```
{  
  "status": 200,  
  "acceptRanges": "bytes",  
  "contentType": "application/json"  
}
```

`output.json`:

```
{"VaultARN":"arn:aws:glacier:us-west-2:0123456789012:vaults/  
my-vault", "InventoryDate":"2015-04-07T00:26:18Z", "ArchiveList":  
[{"ArchiveId":"kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIEWQX-  
ybtRDvc2VkJPSDtfKmQrj0IRQLSGsNuDp-  
AJVlu2ccmDSyDUMzWkwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw", "ArchiveDescription":"multipart  
upload  
test", "CreationDate":"2015-04-06T22:24:34Z", "Size":3145728, "SHA256TreeHash":"9628195fcdbcb
```

- Untuk detail API, lihat [GetJobOutput](#) di Referensi AWS CLI Perintah.

## get-vault-access-policy

Contoh kode berikut menunjukkan cara menggunakan `get-vault-access-policy`.

## AWS CLI

Untuk mengambil kebijakan akses vault



`get-vault-access-policy` Contoh berikut mengambil kebijakan akses untuk vault yang ditentukan.

```
aws glacier get-vault-access-policy \
  --account-id 111122223333 \
  --vault-name example_vault
```

Output:

```
{
  "policy": {
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam:444455556666:root\"},\"Action\":\"glacier:ListJobs\",\"Resource\":\"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"},{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam:444455556666:root\"},\"Action\":\"glacier:UploadArchive\",\"Resource\":\"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}]}"
  }
}
```

- Untuk detail API, lihat [GetVaultAccessPolicy](#) di Referensi AWS CLI Perintah.

## get-vault-lock

Contoh kode berikut menunjukkan cara menggunakan `get-vault-lock`.

### AWS CLI

Untuk mendapatkan detail kunci brankas

`get-vault-lock` Contoh berikut mengambil detail tentang kunci untuk brankas yang ditentukan.

```
aws glacier get-vault-lock \
  --account-id - \
  --vault-name MyVaultName
```

Output:

```
{
  "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":\"Define-vault-lock\",\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"arn:aws:iam:999999999999:root\"},\"Action\":\"glacier>DeleteArchive\",\"Resource\":\"arn:aws:glacier:us-
```

```
west-2:9999999999:vaults/MyVaultName\", \"Condition\": {\"NumericLessThanEquals\":
{ \"glacier:ArchiveAgeinDays\": \"365\" } } }\",
  \"State\": \"Locked\",
  \"CreationDate\": \"2019-07-29T22:25:28.640Z\"
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan Kunci Vault \(kebijakan kunci GET\)](#) di Panduan Pengembang Amazon Glacier API.

- Untuk detail API, lihat [GetVaultLock](#) di Referensi AWS CLI Perintah.

## get-vault-notifications

Contoh kode berikut menunjukkan cara menggunakan `get-vault-notifications`.

### AWS CLI

Perintah berikut mendapatkan deskripsi konfigurasi notifikasi untuk vault bernama `my-vault`:

```
aws glacier get-vault-notifications --account-id - --vault-name my-vault
```

### Output:

```
{
  "vaultNotificationConfig": {
    "Events": [
      "InventoryRetrievalCompleted",
      "ArchiveRetrievalCompleted"
    ],
    "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault"
  }
}
```

Jika tidak ada pemberitahuan yang dikonfigurasi untuk brankas, kesalahan akan dikembalikan. Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [GetVaultNotifications](#) di Referensi AWS CLI Perintah.

## initiate-job

Contoh kode berikut menunjukkan cara menggunakan `initiate-job`.

## AWS CLI

Perintah berikut memulai pekerjaan untuk mendapatkan inventaris `my-vault` vault:

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters
'{"Type": "inventory-retrieval"}'
```

Output:

```
{
  "location": "/0123456789012/vaults/my-vault/jobs/
zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3R1oGduS7Eg-
R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW",
  "jobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3R1oGduS7Eg-
R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW"
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Perintah berikut memulai pekerjaan untuk mengambil arsip dari vault: `my-vault`

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters
file://job-archive-retrieval.json
```

`job-archive-retrieval.json` adalah file JSON di folder lokal yang menentukan jenis pekerjaan, ID arsip, dan beberapa parameter opsional:

```
{
  "Type": "archive-retrieval",
  "ArchiveId": "kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkJPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDumZwKwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",
  "Description": "Retrieve archive on 2015-07-17",
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-topic"
}
```

ID arsip tersedia dalam output `aws glacier upload-archive` dan `aws glacier get-job-output`.

Output:

```
{
  "location": "/011685312445/vaults/mwunderl/jobs/17IL5-
EkXyEY9Ws95fClzIbk205uLYaFdAY0i-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJ059-8N9aHWav",
  "jobId": "17IL5-EkXy205uLYaFdAY0iEY9Ws95fClzIbk-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJ059-8N9aHWav"
}
```

Lihat Memulai Pekerjaan di Referensi Amazon Glacier API untuk detail tentang format parameter pekerjaan.

- Untuk detail API, lihat [InitiateJob](#) di Referensi AWS CLI Perintah.

## **initiate-multipart-upload**

Contoh kode berikut menunjukkan cara menggunakan `initiate-multipart-upload`.

### AWS CLI

Perintah berikut memulai unggahan multipart ke vault bernama `my-vault` dengan ukuran bagian 1 MiB (1024 x 1024 byte) per file:

```
aws glacier initiate-multipart-upload --account-id - --part-size 1048576 --vault-
name my-vault --archive-description "multipart upload test"
```

Parameter deskripsi arsip adalah opsional. Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Perintah ini mengeluarkan ID unggahan saat berhasil. Gunakan ID unggahan saat mengunggah setiap bagian arsip Anda. `aws glacier upload-multipart-part` Untuk informasi selengkapnya tentang unggahan multipart ke Amazon Glacier menggunakan CLI, lihat [Menggunakan Amazon Glacier AWS di Panduan Pengguna CLI.AWS](#)

- Untuk detail API, lihat [InitiateMultipartUpload](#) di Referensi AWS CLI Perintah.

## **initiate-vault-lock**

Contoh kode berikut menunjukkan cara menggunakan `initiate-vault-lock`.

## AWS CLI

Untuk memulai proses penguncian vault

`initiate-vault-lock` Contoh berikut menginstal kebijakan kunci vault pada vault yang ditentukan dan menyetel status kunci vault ke `InProgress`. Anda harus menyelesaikan proses dengan menelepon `complete-vault-lock` dalam waktu 24 jam untuk mengatur status kunci brankas. `Locked`

```
aws glacier initiate-vault-lock \
  --account-id - \
  --vault-name MyVaultName \
  --policy file://vault_lock_policy.json
```

Isi dari `vault_lock_policy.json`:

```
{"Policy": "{\"Version\":\"2012-10-17\",\"Statement\": [{\"Sid\":\"Define-vault-lock\",\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"arn:aws:iam::999999999999:root\"},\"Action\":\"glacier:DeleteArchive\",\"Resource\":\"arn:aws:glacier:us-west-2:999999999999:vaults/examplevault\",\"Condition\":{\"NumericLessThanEquals\":{\"glacier:ArchiveAgeinDays\":\"365\"}}}]\"}
```

Outputnya adalah ID kunci vault yang dapat Anda gunakan untuk menyelesaikan proses penguncian vault.

```
{
  "lockId": "9QZgEXAMPLEPhvL6xEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Memulai Kunci Vault \(kebijakan penguncian POST\)](#) di Panduan Pengembang Amazon Glacier API.

- Untuk detail API, lihat [InitiateVaultLock](#) di Referensi AWS CLI Perintah.

## list-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-jobs`.

## AWS CLI

Perintah berikut mencantumkan pekerjaan yang sedang berlangsung dan baru saja diselesaikan untuk vault bernama: `my-vault`

```
aws glacier list-jobs --account-id - --vault-name my-vault
```

Output:

```
{
  "JobList": [
    {
      "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
      "RetrievalByteRange": "0-3145727",
      "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
      "Completed": false,
      "SHA256TreeHash":
"9628195fcdcbbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
      "JobId": "l7IL5-EkXyEY9Ws95fClzIbk205uLYaFdAY0i-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJ059-8N9aHWav",
      "ArchiveId": "kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIEWQX-
ybtRDvc2VkpSDtFkmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",
      "JobDescription": "Retrieve archive on 2015-07-17",
      "ArchiveSizeInBytes": 3145728,
      "Action": "ArchiveRetrieval",
      "ArchiveSHA256TreeHash":
"9628195fcdcbbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
      "CreationDate": "2015-07-17T21:16:13.840Z",
      "StatusCode": "InProgress"
    },
    {
      "InventoryRetrievalParameters": {
        "Format": "JSON"
      },
      "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
      "Completed": false,
      "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
R047Yc6FxsdBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61ds04CnMW",
      "Action": "InventoryRetrieval",
      "CreationDate": "2015-07-17T20:23:41.616Z",
      "StatusCode": ""InProgress""
    }
  ]
}
```

```
]
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [ListJobs](#) di Referensi AWS CLI Perintah.

## **list-multipart-uploads**

Contoh kode berikut menunjukkan cara menggunakan `list-multipart-uploads`.

### AWS CLI

Perintah berikut menunjukkan semua unggahan multipart yang sedang berlangsung untuk vault bernama: `my-vault`

```
aws glacier list-multipart-uploads --account-id - --vault-name my-vault
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Untuk informasi selengkapnya tentang unggahan multipart ke Amazon Glacier menggunakan CLI, lihat [Menggunakan Amazon Glacier AWS](#) di Panduan Pengguna CLI.AWS

- Untuk detail API, lihat [ListMultipartUploads](#) di Referensi AWS CLI Perintah.

## **list-parts**

Contoh kode berikut menunjukkan cara menggunakan `list-parts`.

### AWS CLI

Perintah berikut mencantumkan bagian yang diunggah untuk unggahan multibagian ke vault bernama: `my-vault`

```
aws glacier list-parts --account-id - --vault-name my-vault --upload-id "SYZi7qnL-
YGqGwAm8Kn3BLP2E1NCvnB-5961R09CSaPmPwkYGH0qeN_nX3-Vhnd2yF0KfB5FkmbnBU9Gubbdrcs8ut-D"
```

Output:

```
{
  "MultipartUploadId": "SYZi7qnL-
YGqGwAm8Kn3BLP2E1NCvnB-5961R09CSaPmPwkYGH0qeN_nX3-Vhnd2yF0KfB5FkmbnBU9GubbdıCs8ut-
D",
  "Parts": [
    {
      "RangeInBytes": "0-1048575",
      "SHA256TreeHash":
"e1f2a7cd6e047350f69b9f8cfa60fa606fe2f02802097a9a026360a7edc1f553"
    },
    {
      "RangeInBytes": "1048576-2097151",
      "SHA256TreeHash":
"43cf3061fb95796aed99a11a6aa3cd8f839eed15e655ab0a597126210636aee6"
    }
  ],
  "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
  "CreationDate": "2015-07-18T00:05:23.830Z",
  "PartSizeInBytes": 1048576
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Untuk informasi selengkapnya tentang unggahan multipart ke Amazon Glacier menggunakan CLI, lihat Menggunakan Amazon Glacier AWS di Panduan Pengguna CLI.AWS

- Untuk detail API, lihat [ListParts](#) di Referensi AWS CLI Perintah.

## list-provisioned-capacity

Contoh kode berikut menunjukkan cara menggunakan `list-provisioned-capacity`.

### AWS CLI

Untuk mengambil unit kapasitas yang disediakan

`list-provisioned-capacity` Contoh berikut mengambil rincian untuk setiap unit kapasitas yang disediakan untuk akun yang ditentukan.

```
aws glacier list-provisioned-capacity \
  --account-id 111122223333
```



**Output:**

```
{
  "ProvisionedCapacityList": [
    {
      "CapacityId": "HpASAvfRFiVDb0jMfEIcr8K",
      "ExpirationDate": "2020-03-18T19:59:24.000Z",
      "StartDate": "2020-02-18T19:59:24.912Z"
    }
  ]
}
```

- Untuk detail API, lihat [ListProvisionedCapacity](#) di Referensi AWS CLI Perintah.

**list-tags-for-vault**

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-vault`.

**AWS CLI**

Perintah berikut mencantumkan tag yang diterapkan ke vault bernama `my-vault`:

```
aws glacier list-tags-for-vault --account-id - --vault-name my-vault
```

**Output:**

```
{
  "Tags": {
    "date": "july2015",
    "id": "1234"
  }
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [ListTagsForVault](#) di Referensi AWS CLI Perintah.

**list-vaults**

Contoh kode berikut menunjukkan cara menggunakan `list-vaults`.

## AWS CLI

Perintah berikut mencantumkan vault di akun dan wilayah default:

```
aws glacier list-vaults --account-id -
```

Output:

```
{
  "VaultList": [
    {
      "SizeInBytes": 3178496,
      "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
      "LastInventoryDate": "2015-04-07T00:26:19.028Z",
      "VaultName": "my-vault",
      "NumberOfArchives": 1,
      "CreationDate": "2015-04-06T21:23:45.708Z"
    }
  ]
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [ListVaults](#) di Referensi AWS CLI Perintah.

## purchase-provisioned-capacity

Contoh kode berikut menunjukkan cara menggunakan `purchase-provisioned-capacity`.

### AWS CLI

Untuk membeli unit kapasitas yang disediakan

`purchase-provisioned-capacity` Contoh berikut membeli unit kapasitas yang disediakan.

```
aws glacier purchase-provisioned-capacity \
  --account-id 111122223333
```

Output:

```
{
```

```
"capacityId": "HpASAUvfRFiVDb0jMfEicr8K"
}
```

- Untuk detail API, lihat [PurchaseProvisionedCapacity](#) di Referensi AWS CLI Perintah.

## **remove-tags-from-vault**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-vault`.

### AWS CLI

Perintah berikut menghapus tag dengan kunci `date` dari vault bernama `my-vault`:

```
aws glacier remove-tags-from-vault --account-id - --vault-name my-vault --tag-keys
date
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [RemoveTagsFromVault](#) di Referensi AWS CLI Perintah.

## **set-data-retrieval-policy**

Contoh kode berikut menunjukkan cara menggunakan `set-data-retrieval-policy`.

### AWS CLI

Perintah berikut mengonfigurasi kebijakan pengambilan data untuk akun yang sedang digunakan:

```
aws glacier set-data-retrieval-policy --account-id - --policy file://data-retrieval-
policy.json
```

`data-retrieval-policy.json` adalah file JSON di folder saat ini yang menentukan kebijakan pengambilan data:

```
{
  "Rules": [
    {
      "Strategy": "BytesPerHour",
      "BytesPerHour": 10737418240
    }
  ]
}
```

```

    }
  ]
}

```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Perintah berikut menetapkan kebijakan pengambilan data untuk FreeTier menggunakan JSON inline:

```
aws glacier set-data-retrieval-policy --account-id - --policy '{"Rules": [{"Strategy": "FreeTier"}]}'
```

Lihat Menyetel Kebijakan Pengambilan Data di Referensi API Amazon Glacier untuk detail tentang format kebijakan.

- Untuk detail API, lihat [SetDataRetrievalPolicy](#) di Referensi AWS CLI Perintah.

## set-vault-access-policy

Contoh kode berikut menunjukkan cara menggunakan `set-vault-access-policy`.

### AWS CLI

Untuk menyetel kebijakan akses vault

`set-vault-access-policy` Contoh berikut melampirkan kebijakan izin ke vault yang ditentukan.

```
aws glacier set-vault-access-policy \
  --account-id 111122223333 \
  --vault-name example_vault
  --policy '{"Policy": "{\"Version\": \"2012-10-17\", \"Statement\": [{"Effect\": \"Allow\", \"Principal\": {\"AWS\": \"arn:aws:iam:444455556666:root\"}, \"Action\": \"glacier:ListJobs\", \"Resource\": \"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}, {\"Effect\": \"Allow\", \"Principal\": {\"AWS\": \"arn:aws:iam:444455556666:root\"}, \"Action\": \"glacier:UploadArchive\", \"Resource\": \"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}]}\"}'
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetVaultAccessPolicy](#) di Referensi AWS CLI Perintah.

## set-vault-notifications

Contoh kode berikut menunjukkan cara menggunakan `set-vault-notifications`.

### AWS CLI

Perintah berikut mengonfigurasi notifikasi SNS untuk vault bernama: `my-vault`

```
aws glacier set-vault-notifications --account-id - --vault-name my-vault --vault-notification-config file://notificationconfig.json
```

`notificationconfig.json` adalah file JSON di folder saat ini yang menentukan topik SNS dan peristiwa yang akan dipublikasikan:

```
{
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
  "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

- Untuk detail API, lihat [SetVaultNotifications](#) di Referensi AWS CLI Perintah.

## upload-archive

Contoh kode berikut menunjukkan cara menggunakan `upload-archive`.

### AWS CLI

Perintah berikut mengunggah arsip di folder saat ini bernama `archive.zip` ke vault bernama: `my-vault`

```
aws glacier upload-archive --account-id - --vault-name my-vault --body archive.zip
```

Output:

```
{
```

```

    "archiveId": "kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIEWQX-
ybtRDvc2VkpSDtfKmQrj0IRQLSGsNuDp-
AJV1u2ccmDSyDumZwKwbwpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",
    "checksum": "969fb39823836d81f0cc028195fcdbcbbe76cdde932d4646fa7de5f21e18aa67",
    "location": "/0123456789012/vaults/my-vault/archives/
kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIEWQX-ybtRDvc2VkpSDtfKmQrj0IRQLSGsNuDp-
AJV1u2ccmDSyDumZwKwbwpAdGATGDiB3hH00bjbGehXTcApVud_wyDw"
}

```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Untuk mengambil arsip yang diunggah, mulailah pekerjaan pengambilan dengan perintah `aws glacier initiate-job`.

- Untuk detail API, lihat [Upload Archived](#) di Referensi AWS CLI Perintah.

## upload-multipart-part

Contoh kode berikut menunjukkan cara menggunakan `upload-multipart-part`.

### AWS CLI

Perintah berikut mengunggah bagian 1 MiB (1024 x 1024 byte) pertama dari arsip:

```

aws glacier upload-multipart-part --body part1 --range 'bytes
0-1048575/*' --account-id - --vault-name my-vault --upload-
id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
0ssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ

```

Amazon Glacier memerlukan argumen ID akun saat melakukan operasi, tetapi Anda dapat menggunakan tanda hubung untuk menentukan akun yang sedang digunakan.

Parameter `body` mengambil jalur ke file bagian pada sistem file lokal. Parameter `range` mengambil rentang konten HTTP yang menunjukkan byte yang ditempati bagian dalam arsip lengkap. ID unggahan dikembalikan oleh `aws glacier initiate-multipart-upload` perintah dan juga dapat diperoleh dengan menggunakan `aws glacier list-multipart-uploads`.

Untuk informasi selengkapnya tentang unggahan multipart ke Amazon Glacier menggunakan CLI, lihat [Menggunakan Amazon Glacier AWS](#) di Panduan Pengguna CLI.AWS

- Untuk detail API, lihat [UploadMultipartPart](#) di Referensi AWS CLI Perintah.

## Secrets Manager contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Secrets Manager AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **batch-get-secret-value**

Contoh kode berikut menunjukkan cara menggunakan `batch-get-secret-value`.

AWS CLI

Contoh 1: Untuk mengambil nilai rahasia untuk sekelompok rahasia yang terdaftar berdasarkan nama

`batch-get-secret-value` Contoh berikut mendapatkan rahasia nilai rahasia untuk tiga rahasia.

```
aws secretsmanager batch-get-secret-value \  
  --secret-id-list MySecret1 MySecret2 MySecret3
```

Output:

```
{  
  "SecretValues": [  
    {  
      "SecretId": "MySecret1",  
      "SecretValue": "MySecret1Value"  
    },  
    {  
      "SecretId": "MySecret2",  
      "SecretValue": "MySecret2Value"  
    },  
    {  
      "SecretId": "MySecret3",  
      "SecretValue": "MySecret3Value"  
    }  
  ]  
}
```

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-
a1b2c3",
  "Name": "MySecret1",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
  "SecretString": "{\"username\":\"diego_ramirez\",\"password\":\"EXAMPLE-
PASSWORD\",\"engine\":\"mysql\",\"host\":\"secretsmanagertutorial.cluster.us-
west-2.rds.amazonaws.com\",\"port\":\"3306\",\"dbClusterIdentifier\":
\"secretsmanagertutorial\"}",
  "VersionStages": [
    "AWSCURRENT"
  ],
  "CreateDate": "1523477145.729"
},
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-
a1b2c3",
  "Name": "MySecret2",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
  "SecretString": "{\"username\":\"akua_mansa\",\"password\":\"EXAMPLE-
PASSWORD\"}",
  "VersionStages": [
    "AWSCURRENT"
  ],
  "CreateDate": "1673477781.275"
},
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
  "Name": "MySecret3",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEccccc",
  "SecretString": "{\"username\":\"jie_liu\",\"password\":\"EXAMPLE-
PASSWORD\"}",
  "VersionStages": [
    "AWSCURRENT"
  ],
  "CreateDate": "1373477721.124"
}
],
"Errors": []
}
```



Untuk informasi selengkapnya, lihat [Mengambil sekelompok rahasia dalam satu kelompok](#) di Panduan Pengguna AWS Secrets Manager.

Contoh 2: Untuk mengambil nilai rahasia untuk sekelompok rahasia yang dipilih oleh filter

`batch-get-secret-value` Contoh berikut mendapatkan rahasia nilai rahasia di akun Anda yang memiliki `MySecret` nama. Pemfilteran berdasarkan nama peka huruf besar/kecil.

```
aws secretsmanager batch-get-secret-value \
  --filters Key="name",Values="MySecret"
```

Output:

```
{
  "SecretValues": [
    {
      "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-
a1b2c3",
      "Name": "MySecret1",
      "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
      "SecretString": "{\"username\":\"diego_ramirez\",\"password\":\"EXAMPLE-
PASSWORD\",\"engine\":\"mysql\",\"host\":\"secretsmanagertutorial.cluster.us-
west-2.rds.amazonaws.com\",\"port\":3306,\"dbClusterIdentifier\":
\"secretsmanagertutorial\"}",
      "VersionStages": [
        "AWSCURRENT"
      ],
      "CreateDate": "1523477145.729"
    },
    {
      "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-
a1b2c3",
      "Name": "MySecret2",
      "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
      "SecretString": "{\"username\":\"akua_mansa\",\"password\":\"EXAMPLE-
PASSWORD\"",
      "VersionStages": [
        "AWSCURRENT"
      ],
      "CreateDate": "1673477781.275"
    },
    {
```

```

    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
    "Name": "MySecret3",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEccccc",
    "SecretString": "{\"username\":\"jie_liu\",\"password\":\"EXAMPLE-
PASSWORD\""}",
    "VersionStages": [
        "AWSCURRENT"
    ],
    "CreateDate": "1373477721.124"
  }
],
"Errors": []
}

```

Untuk informasi selengkapnya, lihat [Mengambil sekelompok rahasia dalam satu kelompok](#) di Panduan Pengguna AWS Secrets Manager.

- Untuk detail API, lihat [BatchGetSecretValue](#) di Referensi AWS CLI Perintah.

## cancel-rotate-secret

Contoh kode berikut menunjukkan cara menggunakan `cancel-rotate-secret`.

### AWS CLI

Untuk mematikan rotasi otomatis untuk rahasia

`cancel-rotate-secret` Contoh berikut mematikan rotasi otomatis untuk sebuah rahasia. Untuk melanjutkan rotasi, hubungi `rotate-secret`.

```
aws secretsmanager cancel-rotate-secret \
  --secret-id MyTestSecret
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret"
}
```

Untuk informasi selengkapnya, lihat [Memutar rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [CancelRotateSecret](#) di Referensi AWS CLI Perintah.

## create-secret

Contoh kode berikut menunjukkan cara menggunakan `create-secret`.

### AWS CLI

Contoh 1: Untuk membuat rahasia

`create-secret` Contoh berikut menciptakan rahasia dengan dua pasangan kunci-nilai.

```
aws secretsmanager create-secret \  
  --name MyTestSecret \  
  --description "My test secret created with the CLI." \  
  --secret-string "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}"
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",  
  "Name": "MyTestSecret",  
  "VersionId": "EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Membuat rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk membuat rahasia dari kredensial dalam file JSON

`create-secret` Contoh berikut membuat rahasia dari kredensial dalam file. Untuk informasi selengkapnya, lihat [Memuat parameter AWS CLI dari file](#) di Panduan Pengguna AWS CLI.

```
aws secretsmanager create-secret \  
  --name MyTestSecret \  
  --secret-string file://mycreds.json
```

Isi dari `mycreds.json`:

```
{  
  "engine": "mysql",
```

```
"username": "saanvis",
"password": "EXAMPLE-PASSWORD",
"host": "my-database-endpoint.us-west-2.rds.amazonaws.com",
"dbname": "myDatabase",
"port": "3306"
}
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Membuat rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [CreateSecret](#) di Referensi AWS CLI Perintah.

## delete-resource-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-policy`.

### AWS CLI

Untuk menghapus kebijakan berbasis sumber daya yang dilampirkan pada rahasia

`delete-resource-policy` Contoh berikut menghapus kebijakan berbasis sumber daya yang dilampirkan pada rahasia.

```
aws secretsmanager delete-resource-policy \
  --secret-id MyTestSecret
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret"
}
```

Untuk informasi selengkapnya, lihat [Otentikasi dan kontrol akses](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [DeleteResourcePolicy](#) di Referensi AWS CLI Perintah.

## delete-secret

Contoh kode berikut menunjukkan cara menggunakan `delete-secret`.

### AWS CLI

Contoh 1: Untuk menghapus rahasia

`delete-secret` Contoh berikut menghapus rahasia. Anda dapat memulihkan rahasia dengan `restore-secret` sampai tanggal dan waktu di bidang `DeletionDate` respons. Untuk menghapus rahasia yang direplikasi ke wilayah lain, pertama-tama hapus replika dengan `remove-regions-from-replication`, lalu panggil `delete-secret`

```
aws secretsmanager delete-secret \  
  --secret-id MyTestSecret \  
  --recovery-window-in-days 7
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret",  
  "DeletionDate": 1524085349.095  
}
```

Untuk informasi selengkapnya, lihat [Menghapus rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk menghapus rahasia segera

`delete-secret` Contoh berikut menghapus rahasia segera tanpa jendela pemulihan. Anda tidak dapat memulihkan rahasia ini.

```
aws secretsmanager delete-secret \  
  --secret-id MyTestSecret \  
  --force-delete-without-recovery
```

**Output:**

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret",
  "DeletionDate": 1508750180.309
}
```

Untuk informasi selengkapnya, lihat [Menghapus rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [DeleteSecret](#) di Referensi AWS CLI Perintah.

**describe-secret**

Contoh kode berikut menunjukkan cara menggunakan `describe-secret`.

**AWS CLI**

Untuk mengambil detail rahasia

`describe-secret` Contoh berikut menunjukkan rincian rahasia.

```
aws secretsmanager describe-secret \
  --secret-id MyTestSecret
```

**Output:**

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
Ca8JGt",
  "Name": "MyTestSecret",
  "Description": "My test secret",
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/EXAMPLE1-90ab-cdef-fedc-
ba987EXAMPLE",
  "RotationEnabled": true,
  "RotationLambdaARN": "arn:aws:lambda:us-
west-2:123456789012:function:MyTestRotationLambda",
  "RotationRules": {
    "AutomaticallyAfterDays": 2,
    "Duration": "2h",
    "ScheduleExpression": "cron(0 16 1,15 * ? *)"
  },
}
```

```
"LastRotatedDate": 1525747253.72,
"LastChangedDate": 1523477145.729,
"LastAccessedDate": 1524572133.25,
"Tags": [
  {
    "Key": "SecondTag",
    "Value": "AnotherValue"
  },
  {
    "Key": "FirstTag",
    "Value": "SomeValue"
  }
],
"VersionIdsToStages": {
  "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
    "AWSPREVIOUS"
  ],
  "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222": [
    "AWSCURRENT"
  ],
  "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333": [
    "AWSPENDING"
  ]
},
"CreateDate": 1521534252.66,
"PrimaryRegion": "us-west-2",
"ReplicationStatus": [
  {
    "Region": "eu-west-3",
    "KmsKeyId": "alias/aws/secretsmanager",
    "Status": "InSync",
    "StatusMessage": "Replication succeeded"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [DescribeSecret](#) di Referensi AWS CLI Perintah.

## get-random-password

Contoh kode berikut menunjukkan cara menggunakan `get-random-password`.

## AWS CLI

Untuk menghasilkan kata sandi acak

`get-random-password` Contoh berikut menghasilkan kata sandi acak sepanjang 20 karakter yang mencakup setidaknya satu huruf besar, huruf kecil, angka, dan tanda baca.

```
aws secretsmanager get-random-password \  
  --require-each-included-type \  
  --password-length 20
```

Output:

```
{  
  "RandomPassword": "EXAMPLE-PASSWORD"  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan mengelola rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [GetRandomPassword](#) di Referensi AWS CLI Perintah.

## `get-resource-policy`

Contoh kode berikut menunjukkan cara menggunakan `get-resource-policy`.

## AWS CLI

Untuk mengambil kebijakan berbasis sumber daya yang dilampirkan pada rahasia

`get-resource-policy` Contoh berikut mengambil kebijakan berbasis sumber daya yang melekat pada rahasia.

```
aws secretsmanager get-resource-policy \  
  --secret-id MyTestSecret
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",
```



```

    "Name": "MyTestSecret",
    "ResourcePolicy": "{\n\"Version\": \"2012-10-17\", \n\"Statement\": [{\n\"Effect\":
\n\"Allow\", \n
\n\"Principal\": {\n\"AWS\": \"arn:aws:iam::123456789012:root\"}, \n\"Action\":
\n\"secretsmanager:GetSecretValue\", \n\"Resource\": \"*\"}]\n}"
}

```

Untuk informasi selengkapnya, lihat [Otentikasi dan kontrol akses](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [GetResourcePolicy](#) di Referensi AWS CLI Perintah.

## get-secret-value

Contoh kode berikut menunjukkan cara menggunakan `get-secret-value`.

### AWS CLI

Contoh 1: Untuk mengambil nilai rahasia terenkripsi dari sebuah rahasia

`get-secret-value` Contoh berikut mendapatkan nilai rahasia saat ini.

```

aws secretsmanager get-secret-value \
  --secret-id MyTestSecret

```

Output:

```

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "SecretString": "{\n\"user\": \"diegor\", \n\"password\": \"EXAMPLE-PASSWORD\"}",
  "VersionStages": [
    "AWSCURRENT"
  ],
  "CreateDate": 1523477145.713
}

```

Untuk informasi selengkapnya, lihat [Mengambil rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk mengambil nilai rahasia sebelumnya

`get-secret-value` Contoh berikut mendapatkan nilai rahasia sebelumnya. :

```
aws secretsmanager get-secret-value \  
  --secret-id MyTestSecret \  
  --version-stage AWSPREVIOUS
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret",  
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "SecretString": "{\"user\":\"diegor\",\"password\":\"PREVIOUS-EXAMPLE-PASSWORD  
\"}",  
  "VersionStages": [  
    "AWSPREVIOUS"  
  ],  
  "CreateDate": 1523477145.713  
}
```

Untuk informasi selengkapnya, lihat [Mengambil rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [GetSecretValue](#) di Referensi AWS CLI Perintah.

## **list-secret-version-ids**

Contoh kode berikut menunjukkan cara menggunakan `list-secret-version-ids`.

AWS CLI

Untuk daftar semua versi rahasia yang terkait dengan rahasia

`list-secret-version-ids` Contoh berikut mendapat daftar semua versi rahasia.

```
aws secretsmanager list-secret-version-ids \  
  --secret-id MyTestSecret
```

Output:

```
{
```

```

"Versions": [
  {
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "VersionStages": [
      "AWSPREVIOUS"
    ],
    "LastAccessedDate": 1523477145.713,
    "CreateDate": 1523477145.713
  },
  {
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "VersionStages": [
      "AWSCURRENT"
    ],
    "LastAccessedDate": 1523477145.713,
    "CreateDate": 1523486221.391
  },
  {
    "CreateDate": 1.51197446236E9,
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333;"
  }
],
"ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
"Name": "MyTestSecret"
}

```

Untuk informasi selengkapnya, lihat [Versi](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [ListSecretVersionIds](#) di Referensi AWS CLI Perintah.

## list-secrets

Contoh kode berikut menunjukkan cara menggunakan `list-secrets`.

### AWS CLI

Contoh 1: Untuk membuat daftar rahasia di akun Anda

`list-secrets` Contoh berikut mendapatkan daftar rahasia di akun Anda.

```
aws secretsmanager list-secrets
```

**Output:**

```
{
  "SecretList": [
    {
      "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-a1b2c3",
      "Name": "MyTestSecret",
      "LastChangedDate": 1523477145.729,
      "SecretVersionsToStages": {
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
          "AWSCURRENT"
        ]
      }
    },
    {
      "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:AnotherSecret-d4e5f6",
      "Name": "AnotherSecret",
      "LastChangedDate": 1523482025.685,
      "SecretVersionsToStages": {
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222": [
          "AWSCURRENT"
        ]
      }
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menemukan rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk memfilter daftar rahasia di akun Anda

`list-secrets` Contoh berikut mendapatkan daftar rahasia di akun Anda yang ada Test di nama. Pemfilteran berdasarkan nama peka huruf besar/kecil.

```
aws secretsmanager list-secrets \
  --filter Key="name",Values="Test"
```

**Output:**

```
{
```

```

"SecretList": [
  {
    "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-a1b2c3",
    "Name": "MyTestSecret",
    "LastChangedDate": 1523477145.729,
    "SecretVersionsToStages": {
      "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
        "AWSCURRENT"
      ]
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [Menemukan rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 3: Untuk membuat daftar rahasia di akun Anda yang dikelola oleh layanan lain

`list-secrets` Contoh berikut mengembalikan rahasia di akun Anda yang dikelola oleh Amazon RDS.

```

aws secretsmanager list-secrets \
  --filter Key="owning-service",Values="rds"

```

Output:

```

{
  "SecretList": [
    {
      "Name": "rds!cluster-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Tags": [
        {
          "Value": "arn:aws:rds:us-
west-2:123456789012:cluster:database-1",
          "Key": "aws:rds:primaryDBClusterArn"
        },
        {
          "Value": "rds",
          "Key": "aws:secretsmanager:owningService"
        }
      ]
    }
  ],
}

```

```

    "RotationRules": {
      "AutomaticallyAfterDays": 1
    },
    "LastChangedDate": 1673477781.275,
    "LastRotatedDate": 1673477781.26,
    "SecretVersionsToStages": {
      "a1b2c3d4-5678-90ab-cdef-EXAMPLEeaaaa": [
        "AWSPREVIOUS"
      ],
      "a1b2c3d4-5678-90ab-cdef-EXAMPLEebbbbb": [
        "AWSCURRENT",
        "AWSPENDING"
      ]
    },
    "OwningService": "rds",
    "RotationEnabled": true,
    "CreatedDate": 1673467300.7,
    "LastAccessedDate": 1673395200.0,
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:rds!
cluster-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111-a1b2c3",
    "Description": "Secret associated with primary RDS DB cluster:
arn:aws:rds:us-west-2:123456789012:cluster:database-1"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Rahasia yang dikelola oleh layanan lain](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [ListSecrets](#) di Referensi AWS CLI Perintah.

## put-resource-policy

Contoh kode berikut menunjukkan cara menggunakan `put-resource-policy`.

### AWS CLI

Untuk menambahkan kebijakan berbasis sumber daya ke rahasia

`put-resource-policy` Contoh berikut menambahkan kebijakan izin ke rahasia, memeriksa terlebih dahulu bahwa kebijakan tersebut tidak menyediakan akses luas ke rahasia tersebut. Kebijakan dibaca dari file. Untuk informasi selengkapnya, lihat [Memuat parameter AWS CLI dari file](#) di Panduan Pengguna AWS CLI.

```
aws secretsmanager put-resource-policy \  
  --secret-id MyTestSecret \  
  --resource-policy file://mypolicy.json \  
  --block-public-policy
```

Isi dari `mypolicy.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "arn:aws:iam::123456789012:role/MyRole"  
      },  
      "Action": "secretsmanager:GetSecretValue",  
      "Resource": "*"\  
    }  
  ]  
}
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Untuk informasi selengkapnya, lihat [Melampirkan kebijakan izin ke rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [PutResourcePolicy](#) di Referensi AWS CLI Perintah.

## put-secret-value

Contoh kode berikut menunjukkan cara menggunakan `put-secret-value`.

### AWS CLI

Contoh 1: Untuk menyimpan nilai rahasia baru dalam rahasia

`put-secret-value` Contoh berikut membuat versi baru dari rahasia dengan dua pasangan kunci-nilai.

```
aws secretsmanager put-secret-value \  
  --secret-id MyTestSecret \  
  --secret-string "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}"
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-  
west-2:123456789012:secret:MyTestSecret-1a2b3c",  
  "Name": "MyTestSecret",  
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "VersionStages": [  
    "AWSCURRENT"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk menyimpan nilai rahasia baru dari kredensial dalam file JSON

`put-secret-value` Contoh berikut membuat versi baru dari rahasia dari kredensial dalam file. Untuk informasi selengkapnya, lihat [Memuat parameter AWS CLI dari file](#) di Panduan Pengguna AWS CLI.

```
aws secretsmanager put-secret-value \  
  --secret-id MyTestSecret \  
  --secret-string file://mycreds.json
```

Isi dari `mycreds.json`:

```
{  
  "engine": "mysql",  
  "username": "saanvis",  
  "password": "EXAMPLE-PASSWORD",  
  "host": "my-database-endpoint.us-west-2.rds.amazonaws.com",  
  "dbname": "myDatabase",
```



```
"port": "3306"
}
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "VersionStages": [
    "AWSCURRENT"
  ]
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [PutSecretValue](#) di Referensi AWS CLI Perintah.

## **remove-regions-from-replication**

Contoh kode berikut menunjukkan cara menggunakan `remove-regions-from-replication`.

### AWS CLI

Untuk menghapus rahasia replika

`remove-regions-from-replication` Contoh berikut menghapus rahasia replika di `eu-west-3`. Untuk menghapus rahasia utama yang direplikasi ke wilayah lain, pertama-tama hapus replika dan kemudian panggil `delete-secret`

```
aws secretsmanager remove-regions-from-replication \
  --secret-id MyTestSecret \
  --remove-replica-regions eu-west-3
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
```

```
"ReplicationStatus": []
}
```

Untuk informasi selengkapnya, lihat [Menghapus rahasia replika](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [RemoveRegionsFromReplication](#) di Referensi AWS CLI Perintah.

## replicate-secret-to-regions

Contoh kode berikut menunjukkan cara menggunakan `replicate-secret-to-regions`.

### AWS CLI

Untuk mereplikasi rahasia ke wilayah lain

`replicate-secret-to-regions` Contoh berikut mereplikasi rahasia eu-west-3. Replika dienkripsi dengan kunci yang dikelola. AWS `aws/secretsmanager`

```
aws secretsmanager replicate-secret-to-regions \
  --secret-id MyTestSecret \
  --add-replica-regions Region=eu-west-3
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
  "ReplicationStatus": [
    {
      "Region": "eu-west-3",
      "KmsKeyId": "alias/aws/secretsmanager",
      "Status": "InProgress"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mereplikasi rahasia ke Wilayah lain](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [ReplicateSecretToRegions](#) di Referensi AWS CLI Perintah.

## restore-secret

Contoh kode berikut menunjukkan cara menggunakan `restore-secret`.

### AWS CLI

Untuk mengembalikan rahasia yang sebelumnya dihapus

`restore-secret` Contoh berikut mengembalikan rahasia yang sebelumnya dijadwalkan untuk dihapus.

```
aws secretsmanager restore-secret \  
  --secret-id MyTestSecret
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Untuk informasi selengkapnya, lihat [Menghapus rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [RestoreSecret](#) di Referensi AWS CLI Perintah.

## rotate-secret

Contoh kode berikut menunjukkan cara menggunakan `rotate-secret`.

### AWS CLI

Contoh 1: Untuk mengkonfigurasi dan memulai rotasi otomatis untuk rahasia

`rotate-secret` Contoh berikut mengkonfigurasi dan memulai rotasi otomatis untuk rahasia. Secrets Manager memutar rahasia sekali segera, dan kemudian setiap delapan jam dalam jendela dua jam. Output menunjukkan `VersionId` versi rahasia baru yang dibuat oleh rotasi.

```
aws secretsmanager rotate-secret \  
  --secret-id MyTestDatabaseSecret \  
  --rotation-lambda-arn arn:aws:lambda:us-  
west-2:1234566789012:function:SecretsManagerTestRotationLambda \  
  --rotation-period 8h
```

```
--rotation-rules "{\"ScheduleExpression\": \"cron(0 8/8 * * ? *)\", \"Duration\": \"2h\"}"
```

Output:

```
{
  "ARN": "aws:arn:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Memutar rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk mengkonfigurasi dan memulai rotasi otomatis pada interval rotasi

`rotate-secret` Contoh berikut mengkonfigurasi dan memulai rotasi otomatis untuk rahasia. Secrets Manager memutar rahasia sekali segera, dan kemudian setiap 10 hari. Output menunjukkan `VersionId` versi rahasia baru yang dibuat oleh rotasi.

```
aws secretsmanager rotate-secret \
  --secret-id MyTestDatabaseSecret \
  --rotation-lambda-arn arn:aws:lambda:us-west-2:1234566789012:function:SecretsManagerTestRotationLambda \
  --rotation-rules "{\"ScheduleExpression\": \"rate(10 days)\"}"
```

Output:

```
{
  "ARN": "aws:arn:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Memutar rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 3: Untuk segera memutar rahasia

`rotate-secret` Contoh berikut memulai rotasi langsung. Output menunjukkan `VersionId` versi rahasia baru yang dibuat oleh rotasi. Rahasiannya harus sudah memiliki rotasi yang dikonfigurasi.

```
aws secretsmanager rotate-secret \  
  --secret-id MyTestDatabaseSecret
```

Output:

```
{  
  "ARN": "aws:arn:secretsmanager:us-  
west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",  
  "Name": "MyTestDatabaseSecret",  
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Untuk informasi selengkapnya, lihat [Memutar rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [RotateSecret](#) di Referensi AWS CLI Perintah.

## stop-replication-to-replica

Contoh kode berikut menunjukkan cara menggunakan `stop-replication-to-replica`.

### AWS CLI

Untuk mempromosikan rahasia replika ke primer

`stop-replication-to-replica` Contoh berikut menghapus link antara rahasia replika ke primer. Rahasia replika dipromosikan menjadi rahasia utama di wilayah replika. Anda harus menelepon `stop-replication-to-replica` dari dalam wilayah replika.

```
aws secretsmanager stop-replication-to-replica \  
  --secret-id MyTestSecret
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3"  
}
```

Untuk informasi selengkapnya, lihat [Mempromosikan rahasia replika](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [StopReplicationToReplica](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Contoh 1: Untuk menambahkan tag ke rahasia

Contoh berikut menunjukkan cara melampirkan tag dengan sintaks singkat.

```
aws secretsmanager tag-resource \  
  --secret-id MyTestSecret \  
  --tags Key=FirstTag,Value=FirstValue
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai rahasia Anda](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk menambahkan beberapa tag ke rahasia

`tag-resource` Contoh berikut melampirkan dua tag kunci-nilai ke rahasia.

```
aws secretsmanager tag-resource \  
  --secret-id MyTestSecret \  
  --tags '[{"Key": "FirstTag", "Value": "FirstValue"}, {"Key": "SecondTag",  
  "Value": "SecondValue"}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari rahasia

`untag-resource` Contoh berikut menghapus dua tag dari rahasia. Untuk setiap tag, kunci dan nilai dihapus.

```
aws secretsmanager untag-resource \  
  --secret-id MyTestSecret \  
  --tag-keys '[ "FirstTag", "SecondTag"]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-secret-version-stage

Contoh kode berikut menunjukkan cara menggunakan `update-secret-version-stage`.

### AWS CLI

Contoh 1: Untuk mengembalikan rahasia ke versi sebelumnya

`update-secret-version-stage` Contoh berikut memindahkan label pementasan AWS CURRENT ke versi rahasia sebelumnya, yang mengembalikan rahasia ke versi sebelumnya. Untuk menemukan ID untuk versi sebelumnya, gunakan `list-secret-version-ids`. Untuk contoh ini, versi dengan label AWS CURRENT adalah `A1b2c3d4-5678-90ab-cdef-example11111` dan versi dengan label SEBELUMNYA adalah `A1b2c3d4-5678-90ab-cdef-example22222`. AWS Dalam contoh ini, Anda memindahkan label AWS CURRENT dari versi 11111 ke 22222. Karena label AWS CURRENT dihapus dari versi, `update-secret-version-stage` secara otomatis memindahkan label AWS PREVIOUS ke versi tersebut (11111). Efeknya adalah versi AWS CURRENT dan AWS PREVIOUS ditukar.

```
aws secretsmanager update-secret-version-stage \  
  --secret-id MyTestSecret \  
  --version-stage AWSCURRENT \  
  --move-to-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 \  
  --remove-from-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Untuk informasi selengkapnya, lihat [Versi](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk menambahkan label pementasan yang dilampirkan ke versi rahasia

`update-secret-version-stage` Contoh berikut menambahkan label pementasan ke versi rahasia. Anda dapat meninjau hasilnya dengan menjalankan `list-secret-version-ids` dan melihat bidang `VersionStages` respons untuk versi yang terpengaruh.

```
aws secretsmanager update-secret-version-stage \  
  --secret-id MyTestSecret \  
  --version-stage STAGINGLABEL1 \  
  --move-to-version-id EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Untuk informasi selengkapnya, lihat [Versi](#) di Panduan Pengguna Secrets Manager.

Contoh 3: Untuk menghapus label pementasan yang dilampirkan ke versi rahasia

`update-secret-version-stage` Contoh berikut menghapus label pementasan yang dilampirkan ke versi rahasia. Anda dapat meninjau hasilnya dengan menjalankan `list-secret-version-ids` dan melihat bidang `VersionStages` respons untuk versi yang terpengaruh.

```
aws secretsmanager update-secret-version-stage \  
  --secret-id MyTestSecret \  
  --version-stage STAGINGLABEL1 \  
  --remove-from-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```



Untuk informasi selengkapnya, lihat [Versi](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [UpdateSecretVersionStaged](#) di Referensi AWS CLI Perintah.

## update-secret

Contoh kode berikut menunjukkan cara menggunakan `update-secret`.

### AWS CLI

Contoh 1: Untuk memperbarui deskripsi rahasia

`update-secret` Contoh berikut memperbarui deskripsi rahasia.

```
aws secretsmanager update-secret \  
  --secret-id MyTestSecret \  
  --description "This is a new description for the secret."
```

Output:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi rahasia](#) di Panduan Pengguna Secrets Manager.

Contoh 2: Untuk memperbarui kunci enkripsi yang terkait dengan rahasia

`update-secret` Contoh berikut memperbarui kunci KMS yang digunakan untuk mengenkripsi nilai rahasia. Kunci KMS harus berada di wilayah yang sama dengan rahasia.

```
aws secretsmanager update-secret \  
  --secret-id MyTestSecret \  
  --kms-key-id arn:aws:kms:us-west-2:123456789012:key/EXAMPLE1-90ab-cdef-fedc-  
ba987EXAMPLE
```

Output:

```
{
```

```
"ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret"
}
```

Untuk informasi selengkapnya, lihat [Memodifikasi rahasia](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [UpdateSecret](#) di Referensi AWS CLI Perintah.

## **validate-resource-policy**

Contoh kode berikut menunjukkan cara menggunakan `validate-resource-policy`.

### AWS CLI

Untuk memvalidasi kebijakan sumber daya

`validate-resource-policy` Contoh berikut memeriksa bahwa kebijakan sumber daya tidak memberikan akses luas ke rahasia. Kebijakan dibaca dari file pada disk. Untuk informasi selengkapnya, lihat [Memuat parameter AWS CLI dari file](#) di Panduan Pengguna AWS CLI.

```
aws secretsmanager validate-resource-policy \
  --resource-policy file://mypolicy.json
```

Isi dari `mypolicy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/MyRole"
      },
      "Action": "secretsmanager:GetSecretValue",
      "Resource": "*"
    }
  ]
}
```

Output:

```
{
  "PolicyValidationPassed": true,
  "ValidationErrors": []
}
```

Untuk informasi selengkapnya, lihat [Referensi izin untuk Secrets Manager](#) di Panduan Pengguna Secrets Manager.

- Untuk detail API, lihat [ValidateResourcePolicy](#) di Referensi AWS CLI Perintah.

## Contoh Security Hub menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Security Hub.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **accept-administrator-invitation**

Contoh kode berikut menunjukkan cara menggunakan `accept-administrator-invitation`.

AWS CLI

Untuk menerima undangan dari akun administrator

`accept-administrator-invitation` Contoh berikut menerima undangan yang ditentukan dari akun administrator yang ditentukan.

```
aws securityhub accept-invitation \  
  --administrator-id 123456789012 \  
  --invitation-id 7ab938c5d52d7904ad09f9e7c20cc4eb
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [AcceptAdministratorInvitation](#) di Referensi AWS CLI Perintah.

## **accept-invitation**

Contoh kode berikut menunjukkan cara menggunakan `accept-invitation`.

### AWS CLI

Untuk menerima undangan dari akun administrator

`accept-invitation` Contoh berikut menerima undangan yang ditentukan dari akun administrator yang ditentukan.

```
aws securityhub accept-invitation \  
  --master-id 123456789012 \  
  --invitation-id 7ab938c5d52d7904ad09f9e7c20cc4eb
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [AcceptInvitation](#) di Referensi AWS CLI Perintah.

## **batch-delete-automation-rules**

Contoh kode berikut menunjukkan cara menggunakan `batch-delete-automation-rules`.

### AWS CLI

Untuk menghapus aturan otomatisasi

`batch-delete-automation-rules` Contoh berikut menghapus aturan otomatisasi yang ditentukan. Anda dapat menghapus satu atau beberapa aturan dengan satu perintah. Hanya akun administrator Security Hub yang dapat menjalankan perintah ini.

```
aws securityhub batch-delete-automation-rules \  
  --automation-rules-arns '["arn:aws:securityhub:us-  
east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"]'
```

Output:

```
{  
  "ProcessedAutomationRules": [  
    "arn:aws:securityhub:us-east-1:123456789012:automation-rule/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
  ],  
  "UnprocessedAutomationRules": []  
}
```

Untuk informasi selengkapnya, lihat [Menghapus aturan otomatisasi](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchDeleteAutomationRules](#) di Referensi AWS CLI Perintah.

## **batch-disable-standards**

Contoh kode berikut menunjukkan cara menggunakan `batch-disable-standards`.

### AWS CLI

Untuk menonaktifkan standar

`batch-disable-standards` Contoh berikut menonaktifkan standar yang terkait dengan ARN berlangganan yang ditentukan.

```
aws securityhub batch-disable-standards \  
  --standards-subscription-arns "arn:aws:securityhub:us-  
west-1:123456789012:subscription/pci-dss/v/3.2.1"
```

Output:

```
{  
  "StandardsSubscriptions": [  

```

```

    {
      "StandardsArn": "arn:aws:securityhub:eu-central-1::standards/pci-dss/
v/3.2.1",
      "StandardsInput": { },
      "StandardsStatus": "DELETING",
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Menonaktifkan atau mengaktifkan standar keamanan](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchDisableStandards](#) di Referensi AWS CLI Perintah.

## batch-enable-standards

Contoh kode berikut menunjukkan cara menggunakan `batch-enable-standards`.

### AWS CLI

Untuk mengaktifkan standar

`batch-enable-standards` Contoh berikut memungkinkan standar PCI DSS untuk akun yang meminta.

```

aws securityhub batch-enable-standards \
  --standards-subscription-requests '{"StandardsArn":"arn:aws:securityhub:us-
west-1::standards/pci-dss/v/3.2.1"}'

```

Output:

```

{
  "StandardsSubscriptions": [
    {
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/
v/3.2.1",
      "StandardsInput": { },
      "StandardsStatus": "PENDING",
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
    }
  ]
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Menonaktifkan atau mengaktifkan standar keamanan](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchEnableStandards](#) di Referensi AWS CLI Perintah.

## batch-get-automation-rules

Contoh kode berikut menunjukkan cara menggunakan `batch-get-automation-rules`.

### AWS CLI

Untuk mendapatkan detail untuk aturan otomatisasi

`batch-get-automation-rules` Contoh berikut mendapatkan rincian untuk aturan otomatisasi yang ditentukan. Anda bisa mendapatkan detail untuk satu atau lebih aturan otomatisasi dengan satu perintah.

```
aws securityhub batch-get-automation-rules \
  --automation-rules-arns '["arn:aws:securityhub:us-
east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"]'
```

### Output:

```
{
  "Rules": [
    {
      "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "RuleStatus": "ENABLED",
      "RuleOrder": 1,
      "RuleName": "Suppress informational findings",
      "Description": "Suppress GuardDuty findings with Informational
severity",
      "IsTerminal": false,
      "Criteria": {
        "ProductName": [
          {
            "Value": "GuardDuty",
            "Comparison": "EQUALS"
          }
        ]
      }
    }
  ]
}
```

```
    ],
    "SeverityLabel": [
      {
        "Value": "INFORMATIONAL",
        "Comparison": "EQUALS"
      }
    ],
    "WorkflowStatus": [
      {
        "Value": "NEW",
        "Comparison": "EQUALS"
      }
    ],
    "RecordState": [
      {
        "Value": "ACTIVE",
        "Comparison": "EQUALS"
      }
    ]
  },
  "Actions": [
    {
      "Type": "FINDING_FIELDS_UPDATE",
      "FindingFieldsUpdate": {
        "Note": {
          "Text": "Automatically suppress GuardDuty findings with
Informational severity",
          "UpdatedBy": "sechub-automation"
        },
        "Workflow": {
          "Status": "SUPPRESSED"
        }
      }
    }
  ],
  "CreatedAt": "2023-05-31T17:56:14.837000+00:00",
  "UpdatedAt": "2023-05-31T17:59:38.466000+00:00",
  "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
}
],
"UnprocessedAutomationRules": []
}
```



Untuk informasi selengkapnya, lihat [Melihat aturan otomatisasi](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchGetAutomationRules](#) di Referensi AWS CLI Perintah.

## batch-get-configuration-policy-associations

Contoh kode berikut menunjukkan cara menggunakan `batch-get-configuration-policy-associations`.

### AWS CLI

Untuk mendapatkan detail asosiasi konfigurasi untuk sejumlah target

`batch-get-configuration-policy-associations` Contoh berikut mengambil rincian asosiasi untuk target yang ditentukan. Anda dapat memberikan ID akun, ID unit organisasi, atau ID root untuk target.

```
aws securityhub batch-get-configuration-policy-associations \  
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

### Output:

```
{  
  "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
  "TargetId": "ou-6hi7-8j91kl2m",  
  "TargetType": "ORGANIZATIONAL_UNIT",  
  "AssociationType": "APPLIED",  
  "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",  
  "AssociationStatus": "SUCCESS",  
  "AssociationStatusMessage": "Association applied successfully on this target."  
}
```

Untuk informasi selengkapnya, lihat [Melihat kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchGetConfigurationPolicyAssociations](#) di Referensi AWS CLI Perintah.

## batch-get-security-controls

Contoh kode berikut menunjukkan cara menggunakan `batch-get-security-controls`.

## AWS CLI

Untuk mendapatkan detail kontrol keamanan

`batch-get-security-controls` Contoh berikut mendapatkan rincian untuk kontrol keamanan ACM.1 dan IAM.1 di akun saat ini dan Wilayah. AWS AWS

```
aws securityhub batch-get-security-controls \
  --security-control-ids ['ACM.1', 'IAM.1']
```

Output:

```
{
  "SecurityControls": [
    {
      "SecurityControlId": "ACM.1",
      "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/ACM.1",
      "Title": "Imported and ACM-issued certificates should be renewed after a
specified time period",
      "Description": "This control checks whether an AWS Certificate Manager
(ACM) certificate is renewed within the specified time period. It checks both
imported certificates and certificates provided by ACM. The control fails if the
certificate isn't renewed within the specified time period. Unless you provide a
custom parameter value for the renewal period, Security Hub uses a default value of
30 days.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
ACM.1/remediation",
      "SeverityRating": "MEDIUM",
      "SecurityControlStatus": "ENABLED"
      "UpdateStatus": "READY",
      "Parameters": {
        "daysToExpiration": {
          "ValueType": CUSTOM,
          "Value": {
            "Integer": 15
          }
        }
      },
      "LastUpdateReason": "Updated control parameter"
    },
    {
      "SecurityControlId": "IAM.1",
```

```

        "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/IAM.1",
        "Title": "IAM policies should not allow full \"*\
privileges",
        "Description": "This AWS control checks whether the default version of
AWS Identity and Access Management (IAM) policies (also known as customer managed
policies) do not have administrator access with a statement that has \"Effect\
\": \"Allow\" with \"Action\": \"*\
\" over \"Resource\": \"*\
\". It only checks for
the Customer Managed Policies that you created, but not inline and AWS Managed
Policies.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
IAM.1/remediation",
        "SeverityRating": "HIGH",
        "SecurityControlStatus": "ENABLED"
        "UpdateStatus": "READY",
        "Parameters": {}
    }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat detail untuk kontrol](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchGetSecurityControls](#) di Referensi AWS CLI Perintah.

## batch-get-standards-control-associations

Contoh kode berikut menunjukkan cara menggunakan `batch-get-standards-control-associations`.

### AWS CLI

Untuk mendapatkan status pemberdayaan kontrol

`batch-get-standards-control-associations` Contoh berikut mengidentifikasi apakah kontrol yang ditentukan diaktifkan dalam standar yang ditentukan.

```

aws securityhub batch-get-standards-control-associations \
  --standards-control-association-ids '[{"SecurityControlId":
  "Config.1", "StandardsArn": "arn:aws:securityhub:us-east-1:123456789012:ruleset/cis-
aws-foundations-benchmark/v/1.2.0"}, {"SecurityControlId": "IAM.6", "StandardsArn":

```

```
"arn:aws:securityhub:us-east-1:123456789012:standards/aws-foundational-security-best-practices/v/1.0.0"]}]'
```

## Output:

```
{
  "StandardsControlAssociationDetails": [
    {
      "StandardsArn": "arn:aws:securityhub::ruleset/cis-aws-foundations-benchmark/v/1.2.0",
      "SecurityControlId": "Config.1",
      "SecurityControlArn": "arn:aws:securityhub:us-east-1:068873283051:security-control/Config.1",
      "AssociationStatus": "ENABLED",
      "RelatedRequirements": [
        "CIS AWS Foundations 2.5"
      ],
      "UpdatedAt": "2022-10-27T16:07:12.960000+00:00",
      "StandardsControlTitle": "Ensure AWS Config is enabled",
      "StandardsControlDescription": "AWS Config is a web service that performs configuration management of supported AWS resources within your account and delivers log files to you. The recorded information includes the configuration item (AWS resource), relationships between configuration items (AWS resources), and any configuration changes between resources. It is recommended to enable AWS Config in all regions.",
      "StandardsControlArns": [
        "arn:aws:securityhub:us-east-1:068873283051:control/cis-aws-foundations-benchmark/v/1.2.0/2.5"
      ]
    },
    {
      "StandardsArn": "arn:aws:securityhub:us-east-1::standards/aws-foundational-security-best-practices/v/1.0.0",
      "SecurityControlId": "IAM.6",
      "SecurityControlArn": "arn:aws:securityhub:us-east-1:068873283051:security-control/IAM.6",
      "AssociationStatus": "DISABLED",
      "RelatedRequirements": [],
      "UpdatedAt": "2022-11-22T21:30:35.080000+00:00",
      "UpdatedReason": "test",
      "StandardsControlTitle": "Hardware MFA should be enabled for the root user",

```

```

    "StandardsControlDescription": "This AWS control checks whether your AWS
account is enabled to use a hardware multi-factor authentication (MFA) device to
sign in with root user credentials.",
    "StandardsControlArns": [
        "arn:aws:securityhub:us-east-1:068873283051:control/aws-
foundational-security-best-practices/v/1.0.0/IAM.6"
    ]
}
]
}

```

Untuk informasi selengkapnya, lihat [Mengaktifkan dan menonaktifkan kontrol dalam standar tertentu di Panduan Pengguna AWS Security Hub](#).

- Untuk detail API, lihat [BatchGetStandardsControlAssociations](#) di Referensi AWS CLI Perintah.

## batch-import-findings

Contoh kode berikut menunjukkan cara menggunakan `batch-import-findings`.

### AWS CLI

Untuk memperbarui temuan

`batch-import-findings` Contoh berikut memperbarui temuan.

```

aws securityhub batch-import-findings \
  --findings '
  [{
    "AwsAccountId": "123456789012",
    "CreatedAt": "2020-05-27T17:05:54.832Z",
    "Description": "Vulnerability in a CloudTrail trail",
    "FindingProviderFields": {
      "Severity": {
        "Label": "LOW",
        "Original": "10"
      },
      "Types": [
        "Software and Configuration Checks/Vulnerabilities/CVE"
      ]
    },
    "GeneratorId": "TestGeneratorId",
    "Id": "Id1",
  }
]
'

```

```

    "ProductArn": "arn:aws:securityhub:us-
west-1:123456789012:product/123456789012/default",
    "Resources": [
      {
        "Id": "arn:aws:cloudtrail:us-west-1:123456789012:trail/
TrailName",
        "Partition": "aws",
        "Region": "us-west-1",
        "Type": "AwsCloudTrailTrail"
      }
    ],
    "SchemaVersion": "2018-10-08",
    "Title": "CloudTrail trail vulnerability",
    "UpdatedAt": "2020-06-02T16:05:54.832Z"
  ]]'

```

### Output:

```

{
  "FailedCount": 0,
  "SuccessCount": 1,
  "FailedFindings": []
}

```

Untuk informasi selengkapnya, lihat [Menggunakan BatchImportFindings untuk membuat dan memperbarui temuan](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchImportFindings](#) di Referensi AWS CLI Perintah.

## batch-update-automation-rules

Contoh kode berikut menunjukkan cara menggunakan `batch-update-automation-rules`.

### AWS CLI

Untuk memperbarui aturan otomatisasi

`batch-update-automation-rules` Contoh berikut memperbarui aturan otomatisasi yang ditentukan. Anda dapat memperbarui satu atau beberapa aturan dengan satu perintah. Hanya akun administrator Security Hub yang dapat menjalankan perintah ini.

```
aws securityhub batch-update-automation-rules \
```

```

--update-automation-rules-request-items '[ \
  { \
    "Actions": [{ \
      "Type": "FINDING_FIELDS_UPDATE", \
      "FindingFieldsUpdate": { \
        "Note": { \
          "Text": "Known issue that is a risk", \
          "UpdatedBy": "sechub-automation" \
        }, \
        "Workflow": { \
          "Status": "NEW" \
        } \
      } \
    }], \
    "Criteria": { \
      "SeverityLabel": [{ \
        "Value": "LOW", \
        "Comparison": "EQUALS" \
      }] \
    }, \
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111", \
    "RuleOrder": 1, \
    "RuleStatus": "DISABLED" \
  } \
]'

```

### Output:

```

{
  "ProcessedAutomationRules": [
    "arn:aws:securityhub:us-east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  ],
  "UnprocessedAutomationRules": []
}

```

Untuk informasi selengkapnya, lihat [Mengedit aturan otomatisasi](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchUpdateAutomationRules](#) di Referensi AWS CLI Perintah.

## batch-update-findings

Contoh kode berikut menunjukkan cara menggunakan `batch-update-findings`.

### AWS CLI

Contoh 1: Untuk memperbarui temuan

`batch-update-findings` Contoh berikut memperbarui dua temuan untuk menambahkan catatan, mengubah label keparahan, dan menyelesaikannya.

```
aws securityhub batch-update-findings \
  --finding-identifiers '[{"Id": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111", "ProductArn": "arn:aws:securityhub:us-
west-1::product/aws/securityhub"}, {"Id": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222", "ProductArn": "arn:aws:securityhub:us-
west-1::product/aws/securityhub"}]' \
  --note '{"Text": "Known issue that is not a risk.", "UpdatedBy": "user1"}' \
  --severity '{"Label": "LOW"}' \
  --workflow '{"Status": "RESOLVED"}'
```

Output:

```
{
  "ProcessedFindings": [
    {
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
    },
    {
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
    }
  ],
  "UnprocessedFindings": []
}
```

Untuk informasi selengkapnya, lihat [Menggunakan BatchUpdateFindings untuk memperbarui temuan](#) di Panduan Pengguna AWS Security Hub.



## Contoh 2: Untuk memperbarui temuan menggunakan sintaks singkatan

batch-update-findings Contoh berikut memperbarui dua temuan untuk menambahkan catatan, mengubah label keparahan, dan menyelesaikannya menggunakan sintaks singkatan.

```
aws securityhub batch-update-findings \  
  --finding-identifiers Id="arn:aws:securityhub:us-  
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",ProductArn="arn:aws:securityhub:us-  
west-1::product/aws/securityhub" Id="arn:aws:securityhub:us-  
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",ProductArn="arn:aws:securityhub:us-  
west-1::product/aws/securityhub" \  
  --note Text="Known issue that is not a risk.",UpdatedBy="user1" \  
  --severity Label="LOW" \  
  --workflow Status="RESOLVED"
```

### Output:

```
{  
  "ProcessedFindings": [  
    {  
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/  
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"  
    },  
    {  
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/  
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"  
    }  
  ],  
  "UnprocessedFindings": []  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan BatchUpdateFindings untuk memperbarui temuan](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [BatchUpdateFindings](#) di Referensi AWS CLI Perintah.

## batch-update-standards-control-associations

Contoh kode berikut menunjukkan cara menggunakan `batch-update-standards-control-associations`.

### AWS CLI

Untuk memperbarui status pemberdayaan kontrol dalam standar yang diaktifkan

`batch-update-standards-control-associations` Contoh berikut menonaktifkan CloudTrail .1 dalam standar yang ditentukan.

```
aws securityhub batch-update-standards-control-associations \  
  --standards-control-association-updates '[{"SecurityControlId": "CloudTrail.1",  
  "StandardsArn": "arn:aws:securityhub::ruleset/cis-aws-foundations-benchmark/  
v/1.2.0", "AssociationStatus": "DISABLED", "UpdatedReason": "Not applicable  
to environment"}, {"SecurityControlId": "CloudTrail.1", "StandardsArn":  
"arn:aws:securityhub::standards/cis-aws-foundations-benchmark/v/1.4.0",  
"AssociationStatus": "DISABLED", "UpdatedReason": "Not applicable to  
environment"}]'
```

Perintah ini tidak menghasilkan output saat berhasil.

Untuk informasi selengkapnya, lihat [Mengaktifkan dan menonaktifkan kontrol dalam standar tertentu](#) serta [Mengaktifkan dan menonaktifkan kontrol di semua standar dalam Panduan Pengguna Security Hub](#).AWS

- Untuk detail API, lihat [BatchUpdateStandardsControlAssociations](#) di Referensi AWS CLI Perintah.

## create-action-target

Contoh kode berikut menunjukkan cara menggunakan `create-action-target`.

### AWS CLI

Untuk membuat tindakan kustom

`create-action-target` Contoh berikut membuat tindakan kustom. Ini memberikan nama, deskripsi, dan pengidentifikasi untuk tindakan tersebut.

```
aws securityhub create-action-target \  

```

```
--name "Send to remediation" \  
--description "Action to send the finding for remediation tracking" \  
--id "Remediation"
```

Output:

```
{  
  "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
Remediation"  
}
```

Untuk informasi selengkapnya, lihat [Membuat tindakan kustom dan mengaitkannya dengan aturan CloudWatch Peristiwa](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [CreateActionTarget](#) di Referensi AWS CLI Perintah.

## create-automation-rule

Contoh kode berikut menunjukkan cara menggunakan `create-automation-rule`.

### AWS CLI

Untuk membuat aturan otomatisasi

`create-automation-rule` Contoh berikut membuat aturan otomatisasi di AWS akun saat ini dan AWS Wilayah. Security Hub memfilter temuan Anda berdasarkan kriteria yang ditentukan dan menerapkan tindakan pada temuan yang cocok. Hanya akun administrator Security Hub yang dapat menjalankan perintah ini.

```
aws securityhub create-automation-rule \  
  --actions '[{ \  
    "Type": "FINDING_FIELDS_UPDATE", \  
    "FindingFieldsUpdate": { \  
      "Severity": { \  
        "Label": "HIGH" \  
      }, \  
      "Note": { \  
        "Text": "Known issue that is a risk. Updated by automation rules", \  
        "UpdatedBy": "sechub-automation" \  
      } \  
    } \  
  ]' \  
  --name "Send to remediation" \  
  --description "Action to send the finding for remediation tracking" \  
  --id "Remediation"
```

```

--criteria '{ \
  "SeverityLabel": [{ \
    "Value": "INFORMATIONAL", \
    "Comparison": "EQUALS" \
  }] \
}' \
--description "A sample rule" \
--no-is-terminal \
--rule-name "sample rule" \
--rule-order 1 \
--rule-status "ENABLED"

```

### Output:

```

{
  "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}

```

Untuk informasi selengkapnya, lihat [Membuat aturan otomatisasi](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [CreateAutomationRule](#) di Referensi AWS CLI Perintah.

## create-configuration-policy

Contoh kode berikut menunjukkan cara menggunakan `create-configuration-policy`.

### AWS CLI

Untuk membuat kebijakan konfigurasi

`create-configuration-policy` Contoh berikut membuat kebijakan konfigurasi dengan pengaturan yang ditentukan.

```

aws securityhub create-configuration-policy \
  --name "SampleConfigurationPolicy" \
  --description "SampleDescription" \
  --configuration-policy '{"SecurityHub": {"ServiceEnabled":
true, "EnabledStandardIdentifiers": ["arn:aws:securityhub:eu-
central-1::standards/aws-foundational-security-best-practices/
v/1.0.0","arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/

```

```
v/1.2.0"], "SecurityControlsConfiguration": {"DisabledSecurityControlIdentifiers":
["CloudTrail.2"], "SecurityControlCustomParameters": [{"SecurityControlId":
"ACM.1", "Parameters": {"daysToExpiration": {"ValueType": "CUSTOM", "Value":
{"Integer": 15}}}]}}]} \
--tags '{"Environment": "Prod"}'
```

## Output:

```
{
  "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Name": "SampleConfigurationPolicy",
  "Description": "SampleDescription",
  "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
  "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
  "ConfigurationPolicy": {
    "SecurityHub": {
      "ServiceEnabled": true,
      "EnabledStandardIdentifiers": [
        "arn:aws:securityhub:eu-central-1::standards/aws-foundational-
security-best-practices/v/1.0.0",
        "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"
      ],
      "SecurityControlsConfiguration": {
        "DisabledSecurityControlIdentifiers": [
          "CloudTrail.2"
        ],
        "SecurityControlCustomParameters": [
          {
            "SecurityControlId": "ACM.1",
            "Parameters": {
              "daysToExpiration": {
                "ValueType": "CUSTOM",
                "Value": {
                  "Integer": 15
                }
              }
            }
          }
        ]
      }
    }
  }
}
```

```
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Membuat dan mengaitkan kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [CreateConfigurationPolicy](#) di Referensi AWS CLI Perintah.

## create-finding-aggregator

Contoh kode berikut menunjukkan cara menggunakan `create-finding-aggregator`.

### AWS CLI

Untuk mengaktifkan menemukan agregasi

`create-finding-aggregator` Contoh berikut mengkonfigurasi agregasi pencarian. Dijalankan dari US East (Virginia), yang menunjuk US East (Virginia) sebagai Wilayah agregasi. Ini menunjukkan untuk hanya menautkan Wilayah tertentu, dan untuk tidak secara otomatis menautkan Wilayah baru. Ini memilih AS Barat (California N.) dan AS Barat (Oregon) sebagai Wilayah terkait.

```
aws securityhub create-finding-aggregator \  
  --region us-east-1 \  
  --region-linking-mode SPECIFIED_REGIONS \  
  --regions us-west-1,us-west-2
```

Output:

```
{  
  "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-  
aggregator/123e4567-e89b-12d3-a456-426652340000",  
  "FindingAggregationRegion": "us-east-1",  
  "RegionLinkingMode": "SPECIFIED_REGIONS",  
  "Regions": "us-west-1,us-west-2"  
}
```

Untuk informasi selengkapnya, lihat [Mengaktifkan agregasi pencarian](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [CreateFindingAggregator](#) di Referensi AWS CLI Perintah.

## create-insight

Contoh kode berikut menunjukkan cara menggunakan `create-insight`.

### AWS CLI

Untuk membuat wawasan khusus

`create-insight` Contoh berikut menciptakan wawasan khusus bernama Temuan peran kritis yang mengembalikan temuan penting yang terkait dengan AWS peran.

```
aws securityhub create-insight \  
  --filters '{"ResourceType": [{ "Comparison": "EQUALS", "Value": "AwsIamRole"}],  
  "SeverityLabel": [{"Comparison": "EQUALS", "Value": "CRITICAL"}]}' \  
  --group-by-attribute "ResourceId" \  
  --name "Critical role findings"
```

Output:

```
{  
  "InsightArn": "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/  
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola wawasan khusus](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [CreateInsight](#) di Referensi AWS CLI Perintah.

## create-members

Contoh kode berikut menunjukkan cara menggunakan `create-members`.

### AWS CLI

Untuk menambahkan akun sebagai akun anggota

`create-members` Contoh berikut menambahkan dua akun sebagai akun anggota ke akun administrator yang meminta.

```
aws securityhub create-members \  

```

```
--account-details '[{"AccountId": "123456789111"}, {"AccountId": "123456789222"}]'
```

Output:

```
{
  "UnprocessedAccounts": []
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [CreateMembers](#) di Referensi AWS CLI Perintah.

## decline-invitations

Contoh kode berikut menunjukkan cara menggunakan `decline-invitations`.

### AWS CLI

Untuk menolak undangan menjadi akun anggota

`decline-invitations` Contoh berikut menolak undangan untuk menjadi akun anggota dari akun administrator yang ditentukan. Akun anggota adalah akun yang meminta.

```
aws securityhub decline-invitations \
  --account-ids "123456789012"
```

Output:

```
{
  "UnprocessedAccounts": []
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeclineInvitations](#) di Referensi AWS CLI Perintah.

## delete-action-target

Contoh kode berikut menunjukkan cara menggunakan `delete-action-target`.



## AWS CLI

Untuk menghapus tindakan kustom

`delete-action-target` Contoh berikut menghapus tindakan kustom yang diidentifikasi oleh ARN yang ditentukan.

```
aws securityhub delete-action-target \  
  --action-target-arn "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
Remediation"
```

Output:

```
{  
  "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
Remediation"  
}
```

Untuk informasi selengkapnya, lihat [Membuat tindakan kustom dan mengaitkannya dengan aturan CloudWatch Peristiwa](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeleteActionTarget](#) di Referensi AWS CLI Perintah.

## **delete-configuration-policy**

Contoh kode berikut menunjukkan cara menggunakan `delete-configuration-policy`.

## AWS CLI

Untuk menghapus kebijakan konfigurasi

`delete-configuration-policy` Contoh berikut menghapus kebijakan konfigurasi yang ditentukan.

```
aws securityhub delete-configuration-policy \  
  --identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus dan melepaskan kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeleteConfigurationPolicy](#) di Referensi AWS CLI Perintah.

## delete-finding-aggregator

Contoh kode berikut menunjukkan cara menggunakan `delete-finding-aggregator`.

### AWS CLI

Untuk berhenti menemukan agregasi

`delete-finding-aggregator` Contoh berikut berhenti menemukan agregasi. Dijalankan dari US East (Virginia), yang merupakan Wilayah agregasi.

```
aws securityhub delete-finding-aggregator \
  --region us-east-1 \
  --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-
  aggregator/123e4567-e89b-12d3-a456-426652340000
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Berhenti menemukan agregasi](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeleteFindingAggregator](#) di Referensi AWS CLI Perintah.

## delete-insight

Contoh kode berikut menunjukkan cara menggunakan `delete-insight`.

### AWS CLI

Untuk menghapus wawasan kustom

`delete-insight` Contoh berikut menghapus wawasan kustom dengan ARN yang ditentukan.

```
aws securityhub delete-insight \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
```

```
"InsightArn": "arn:aws:securityhub:eu-central-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE111111"}
}
```

Untuk informasi selengkapnya, lihat [Mengelola wawasan khusus](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeleteInsight](#) di Referensi AWS CLI Perintah.

## delete-invitations

Contoh kode berikut menunjukkan cara menggunakan `delete-invitations`.

### AWS CLI

Untuk menghapus undangan untuk menjadi akun anggota

`delete-invitations` Contoh berikut menghapus undangan untuk menjadi akun anggota untuk akun administrator yang ditentukan. Akun anggota adalah akun yang meminta.

```
aws securityhub delete-invitations \
  --account-ids "123456789012"
```

Output:

```
{
  "UnprocessedAccounts": []
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeleteInvitations](#) di Referensi AWS CLI Perintah.

## delete-members

Contoh kode berikut menunjukkan cara menggunakan `delete-members`.

### AWS CLI

Untuk menghapus akun anggota

`delete-members` Contoh berikut menghapus akun anggota yang ditentukan dari akun administrator yang meminta.

```
aws securityhub delete-members \  
  --account-ids "123456789111" "123456789222"
```

Output:

```
{  
  "UnprocessedAccounts": []  
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DeleteMembers](#) di Referensi AWS CLI Perintah.

## **describe-action-targets**

Contoh kode berikut menunjukkan cara menggunakan `describe-action-targets`.

AWS CLI

Untuk mengambil detail tentang tindakan kustom

`describe-action-targets` Contoh berikut mengambil informasi tentang tindakan kustom yang diidentifikasi oleh ARN yang ditentukan.

```
aws securityhub describe-action-targets \  
  --action-target-arns "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
Remediation"
```

Output:

```
{  
  "ActionTargets": [  
    {  
      "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/  
custom/Remediation",  
      "Description": "Action to send the finding for remediation tracking",
```

```
        "Name": "Send to remediation"
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Membuat tindakan kustom dan mengaitkannya dengan aturan CloudWatch Peristiwa](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DescribeActionTargets](#) di Referensi AWS CLI Perintah.

## describe-hub

Contoh kode berikut menunjukkan cara menggunakan `describe-hub`.

### AWS CLI

Untuk mendapatkan informasi tentang sumber daya hub

`describe-hub` Contoh berikut mengembalikan tanggal berlangganan untuk sumber daya hub yang ditentukan. Sumber daya hub diidentifikasi oleh ARN-nya.

```
aws securityhub describe-hub \
  --hub-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default"
```

Output:

```
{
  "HubArn": "arn:aws:securityhub:us-west-1:123456789012:hub/default",
  "SubscribedAt": "2019-11-19T23:15:10.046Z"
}
```

Untuk informasi selengkapnya, lihat [AWS:SecurityHub:::Hub](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [DescribeHub](#) di Referensi AWS CLI Perintah.

## describe-organization-configuration

Contoh kode berikut menunjukkan cara menggunakan `describe-organization-configuration`.

## AWS CLI

Untuk melihat cara Security Hub dikonfigurasi untuk organisasi

`describe-organization-configuration` Contoh berikut menampilkan informasi tentang cara organisasi dikonfigurasi di Security Hub. Dalam contoh ini, organisasi menggunakan konfigurasi pusat. Hanya akun administrator Security Hub yang dapat menjalankan perintah ini.

```
aws securityhub describe-organization-configuration
```

Output:

```
{
  "AutoEnable": false,
  "MemberAccountLimitReached": false,
  "AutoEnableStandards": "NONE",
  "OrganizationConfiguration": {
    "ConfigurationType": "LOCAL",
    "Status": "ENABLED",
    "StatusMessage": "Central configuration has been enabled successfully"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun dengan AWS Organizations](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DescribeOrganizationConfiguration](#) di Referensi AWS CLI Perintah.

## **describe-products**

Contoh kode berikut menunjukkan cara menggunakan `describe-products`.

## AWS CLI

Untuk mengembalikan informasi tentang integrasi produk yang tersedia

`describe-products` Contoh berikut mengembalikan integrasi produk yang tersedia satu per satu.

```
aws securityhub describe-products \
```

```
--max-results 1
```

## Output:

```
{
  "NextToken": "U2FsdGVkX18vvP10qb7RDrWRWVFBJI46M0IAb+nZmRJM15NoRi2gm13sdQEn30/
pq/78dGs+bKpgA+7HMPH00qX33/zoRI+uIG/F9yLNhc0r0WzFUdy36JcXLQji3Rpnn/
cD1SVkGA98qI3zPOSDg==",
  "Products": [
    {
      "ProductArn": "arn:aws:securityhub:us-west-1:123456789333:product/
crowdstrike/crowdstrike-falcon",
      "ProductName": "CrowdStrike Falcon",
      "CompanyName": "CrowdStrike",
      "Description": "CrowdStrike Falcon's single lightweight sensor unifies
next-gen antivirus, endpoint detection and response, and 24/7 managed hunting, via
the cloud.",
      "Categories": [
        "Endpoint Detection and Response (EDR)",
        "AV Scanning and Sandboxing",
        "Threat Intelligence Feeds and Reports",
        "Endpoint Forensics",
        "Network Forensics"
      ],
      "IntegrationTypes": [
        "SEND_FINDINGS_TO_SECURITY_HUB"
      ],
      "MarketplaceUrl": "https://aws.amazon.com/marketplace/seller-profile?
id=a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ActivationUrl": "https://falcon.crowdstrike.com/support/documentation",
      "ProductSubscriptionResourcePolicy": "{\"Version\":
\\\"2012-10-17\\\",\\\"Statement\\\":[[{\\\"Effect\\\":\\\"Allow\\\",\\\"Principal\\\":{\\\"AWS\\\":
\\\"123456789333\\\"},\\\"Action\\\":[\\\"securityhub:BatchImportFindings\\\"],\\\"Resource\\\":
\\\"arn:aws:securityhub:us-west-1:123456789012:product-subscription/crowdstrike/
crowdstrike-falcon\\\",\\\"Condition\\\":{\\\"StringEquals\\\":{\\\"securityhub:TargetAccount
\\\":\\\"123456789012\\\"}}},{\\\"Effect\\\":\\\"Allow\\\",\\\"Principal\\\":{\\\"AWS\\\":
\\\"123456789012\\\"},\\\"Action\\\":[\\\"securityhub:BatchImportFindings\\\"],\\\"Resource
\\\":\\\"arn:aws:securityhub:us-west-1:123456789333:product/crowdstrike/crowdstrike-
falcon\\\",\\\"Condition\\\":{\\\"StringEquals\\\":{\\\"securityhub:TargetAccount\\\":
\\\"123456789012\\\"}}]}]}\"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola integrasi produk](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DescribeProducts](#) di Referensi AWS CLI Perintah.

## describe-standards-controls

Contoh kode berikut menunjukkan cara menggunakan `describe-standards-controls`.

### AWS CLI

Untuk meminta daftar kontrol dalam standar yang diaktifkan

`describe-standards-controls` Contoh berikut meminta daftar kontrol dalam langganan akun pemohon ke standar PCI DSS. Permintaan mengembalikan dua kontrol pada satu waktu.

```
aws securityhub describe-standards-controls \
  --standards-subscription-arn "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1" \
  --max-results 2
```

### Output:

```
{
  "Controls": [
    {
      "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.AutoScaling.1",
      "ControlStatus": "ENABLED",
      "ControlStatusUpdatedAt": "2020-05-15T18:49:04.473000+00:00",
      "ControlId": "PCI.AutoScaling.1",
      "Title": "Auto scaling groups associated with a load balancer should use
health checks",
      "Description": "This AWS control checks whether your Auto Scaling groups
that are associated with a load balancer are using Elastic Load Balancing health
checks.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
PCI.AutoScaling.1/remediation",
      "SeverityRating": "LOW",
      "RelatedRequirements": [
        "PCI DSS 2.2"
      ]
    }
  ]
}
```



```

    },
    {
      "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.CW.1",
      "ControlStatus": "ENABLED",
      "ControlStatusUpdatedAt": "2020-05-15T18:49:04.498000+00:00",
      "ControlId": "PCI.CW.1",
      "Title": "A log metric filter and alarm should exist for usage of the
\"root\" user",
      "Description": "This control checks for the CloudWatch metric
filters using the following pattern { $.userIdentity.type = \"Root\" &&
$.userIdentity.invokedBy NOT EXISTS && $.eventType != \"AwsServiceEvent\" }
It checks that the log group name is configured for use with active multi-
region CloudTrail, that there is at least one Event Selector for a Trail with
IncludeManagementEvents set to true and ReadWriteType set to All, and that there is
at least one active subscriber to an SNS topic associated with the alarm.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
PCI.CW.1/remediation",
      "SeverityRating": "MEDIUM",
      "RelatedRequirements": [
        "PCI DSS 7.2.1"
      ]
    }
  ],
  "NextToken": "U2FsdGvkX1+eNkPoZHV111ip5HUYQPWSWZGmftcmJiHL8JoKEsCDuaKayiPDyLK
+LiTkShveo0dvfxXck0BaGhohIXhsIedN+LSjQV/
17kfCfJcq4PziNC1N9xe9aq2pjlLVZnznTfSImrodT5bRNHe4fELCQq/z+5ka
+5Lzmc11axcwTd5lKgQyQqmUVoeriHZhyIiBgWKf7oNYdBVG80EortVWvSkoUTt
+B2ThcnC7143kI0UNx1kZ6sc64AsW"
}

```

Untuk informasi selengkapnya, lihat [Melihat detail untuk kontrol](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DescribeStandardsControls](#) di Referensi AWS CLI Perintah.

## describe-standards

Contoh kode berikut menunjukkan cara menggunakan `describe-standards`.

### AWS CLI

Untuk mengembalikan daftar standar yang tersedia

`describe-standards` Contoh berikut mengembalikan daftar standar yang tersedia.

```
aws securityhub describe-standards
```

Output:

```
{
  "Standards": [
    {
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/aws-
foundational-security-best-practices/v/1.0.0",
      "Name": "AWS Foundational Security Best Practices v1.0.0",
      "Description": "The AWS Foundational Security Best Practices standard
is a set of automated security checks that detect when AWS accounts and deployed
resources do not align to security best practices. The standard is defined by AWS
security experts. This curated set of controls helps improve your security posture
in AWS, and cover AWS's most popular and foundational services.",
      "EnabledByDefault": true
    },
    {
      "StandardsArn": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",
      "Name": "CIS AWS Foundations Benchmark v1.2.0",
      "Description": "The Center for Internet Security (CIS) AWS Foundations
Benchmark v1.2.0 is a set of security configuration best practices for AWS. This
Security Hub standard automatically checks for your compliance readiness against a
subset of CIS requirements.",
      "EnabledByDefault": true
    },
    {
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/
v/3.2.1",
      "Name": "PCI DSS v3.2.1",
      "Description": "The Payment Card Industry Data Security Standard (PCI
DSS) v3.2.1 is an information security standard for entities that store, process,
and/or transmit cardholder data. This Security Hub standard automatically checks
for your compliance readiness against a subset of PCI DSS requirements.",
      "EnabledByDefault": false
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Standar keamanan di AWS Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DescribeStandards](#) di Referensi AWS CLI Perintah.

## **disable-import-findings-for-product**

Contoh kode berikut menunjukkan cara menggunakan `disable-import-findings-for-product`.

### AWS CLI

Untuk berhenti menerima temuan dari integrasi produk

`disable-import-findings-for-product` Contoh berikut menonaktifkan aliran temuan untuk langganan yang ditentukan untuk integrasi produk.

```
aws securityhub disable-import-findings-for-product \  
  --product-subscription-arn "arn:aws:securityhub:us-west-1:123456789012:product-  
subscription/crowdstrike/crowdstrike-falcon"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola integrasi produk](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DisableImportFindingsForProduct](#) di Referensi AWS CLI Perintah.

## **disable-organization-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `disable-organization-admin-account`.

### AWS CLI

Untuk menghapus akun administrator Security Hub

`disable-organization-admin-account` Contoh berikut mencabut penetapan akun yang ditentukan sebagai akun administrator Security Hub untuk Organizations AWS .

```
aws securityhub disable-organization-admin-account \  
  --admin-account-id 777788889999
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menetapkan akun administrator Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DisableOrganizationAdminAccount](#) di Referensi AWS CLI Perintah.

## **disable-security-hub**

Contoh kode berikut menunjukkan cara menggunakan `disable-security-hub`.

### AWS CLI

Untuk menonaktifkan AWS Security Hub

`disable-security-hub` Contoh berikut menonaktifkan AWS Security Hub untuk akun yang meminta.

```
aws securityhub disable-security-hub
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan AWS Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DisableSecurityHub](#) di Referensi AWS CLI Perintah.

## **disassociate-from-administrator-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-from-administrator-account`.

### AWS CLI

Untuk memisahkan diri dari akun administrator

`disassociate-from-administrator-account` Contoh berikut memisahkan akun yang meminta dari akun administrator saat ini.

```
aws securityhub disassociate-from-administrator-account
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DisassociateFromAdministratorAccount](#) di Referensi AWS CLI Perintah.

## **disassociate-from-master-account**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-from-master-account`.

### AWS CLI

Untuk memisahkan diri dari akun administrator

`disassociate-from-master-account` Contoh berikut memisahkan akun yang meminta dari akun administrator saat ini.

```
aws securityhub disassociate-from-master-account
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DisassociateFromMasterAccount](#) di Referensi AWS CLI Perintah.

## **disassociate-members**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-members`.

### AWS CLI

Untuk memisahkan akun anggota

`disassociate-members` Contoh berikut memisahkan akun anggota yang ditentukan dari akun administrator yang meminta.

```
aws securityhub disassociate-members \  
  --account-ids "123456789111" "123456789222"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [DisassociateMembers](#) di Referensi AWS CLI Perintah.

## **enable-import-findings-for-product**

Contoh kode berikut menunjukkan cara menggunakan `enable-import-findings-for-product`.

### AWS CLI

Untuk mulai menerima temuan dari integrasi produk

`enable-import-findings-for-product` Contoh berikut memungkinkan aliran temuan dari integrasi produk yang ditentukan.

```
aws securityhub enable-import-findings-for-product \  
  --product-arn "arn:aws:securityhub:us-east-1:123456789333:product/crowdstrike/  
crowdstrike-falcon"
```

Output:

```
{  
  "ProductSubscriptionArn": "arn:aws:securityhub:us-east-1:123456789012:product-  
subscription/crowdstrike/crowdstrike-falcon"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola integrasi produk](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [EnableImportFindingsForProduct](#) di Referensi AWS CLI Perintah.

## **enable-organization-admin-account**

Contoh kode berikut menunjukkan cara menggunakan `enable-organization-admin-account`.

### AWS CLI

Untuk menetapkan akun organisasi sebagai akun administrator Security Hub

`enable-organization-admin-account` Contoh berikut menunjuk akun yang ditentukan sebagai akun administrator Security Hub.

```
aws securityhub enable-organization-admin-account \  
  --admin-account-id 777788889999
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menetapkan akun administrator Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [EnableOrganizationAdminAccount](#) di Referensi AWS CLI Perintah.

## **enable-security-hub**

Contoh kode berikut menunjukkan cara menggunakan `enable-security-hub`.

### AWS CLI

Untuk mengaktifkan AWS Security Hub

`enable-security-hub` Contoh berikut memungkinkan AWS Security Hub untuk akun yang meminta. Ini mengkonfigurasi Security Hub untuk mengaktifkan standar default. Untuk sumber daya hub, ia memberikan nilai `Security` ke tag `Department`.

```
aws securityhub enable-security-hub \  
  --enable-default-standards \  
  --tags '{"Department": "Security"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [EnableSecurityHub](#) di Referensi AWS CLI Perintah.

## **get-administrator-account**

Contoh kode berikut menunjukkan cara menggunakan `get-administrator-account`.

### AWS CLI

Untuk mengambil informasi tentang akun administrator

`get-administrator-account` Contoh berikut mengambil informasi tentang akun administrator untuk akun yang meminta.

```
aws securityhub get-administrator-account
```

Output:

```
{
  "Master": {
    "AccountId": "123456789012",
    "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
    "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
    "MemberStatus": "ASSOCIATED"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetAdministratorAccount](#) di Referensi AWS CLI Perintah.

## **get-configuration-policy-association**

Contoh kode berikut menunjukkan cara menggunakan `get-configuration-policy-association`.

AWS CLI

Untuk mendapatkan detail asosiasi konfigurasi untuk target

`get-configuration-policy-association` Contoh berikut mengambil rincian asosiasi untuk target yang ditentukan. Anda dapat memberikan ID akun, ID unit organisasi, atau ID root untuk target.

```
aws securityhub get-configuration-policy-association \
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

Output:

```
{
  "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
```



```
"TargetId": "ou-6hi7-8j91k12m",
"TargetType": "ORGANIZATIONAL_UNIT",
"AssociationType": "APPLIED",
"UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
"AssociationStatus": "SUCCESS",
"AssociationStatusMessage": "Association applied successfully on this target."
}
```

Untuk informasi selengkapnya, lihat [Melihat kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetConfigurationPolicyAssociation](#) di Referensi AWS CLI Perintah.

## get-configuration-policy

Contoh kode berikut menunjukkan cara menggunakan `get-configuration-policy`.

### AWS CLI

Untuk melihat detail kebijakan konfigurasi

`get-configuration-policy` Contoh berikut mengambil rincian tentang kebijakan konfigurasi yang ditentukan.

```
aws securityhub get-configuration-policy \
  --identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
  "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Id": "ce5ed1e7-9639-4e2f-9313-fa87fcef944b",
  "Name": "SampleConfigurationPolicy",
  "Description": "SampleDescription",
  "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
  "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
  "ConfigurationPolicy": {
    "SecurityHub": {
      "ServiceEnabled": true,
      "EnabledStandardIdentifiers": [
```

```

        "arn:aws:securityhub:eu-central-1::standards/aws-foundational-
security-best-practices/v/1.0.0",
        "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"
    ],
    "SecurityControlsConfiguration": {
        "DisabledSecurityControlIdentifiers": [
            "CloudTrail.2"
        ],
        "SecurityControlCustomParameters": [
            {
                "SecurityControlId": "ACM.1",
                "Parameters": {
                    "daysToExpiration": {
                        "ValueType": "CUSTOM",
                        "Value": {
                            "Integer": 15
                        }
                    }
                }
            }
        ]
    }
}

```

Untuk informasi selengkapnya, lihat [Melihat kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetConfigurationPolicy](#) di Referensi AWS CLI Perintah.

## get-enabled-standards

Contoh kode berikut menunjukkan cara menggunakan `get-enabled-standards`.

### AWS CLI

Untuk mengambil informasi tentang standar yang diaktifkan

`get-enabled-standards` Contoh berikut mengambil informasi tentang standar PCI DSS.

```
aws securityhub get-enabled-standards \
```

```
--standards-subscription-arn "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/v/3.2.1"
```

Output:

```
{
  "StandardsSubscriptions": [
    {
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/v/3.2.1",
      "StandardsInput": { },
      "StandardsStatus": "READY",
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/v/3.2.1"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Standar keamanan di AWS Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetEnabledStandards](#) di Referensi AWS CLI Perintah.

## get-finding-aggregator

Contoh kode berikut menunjukkan cara menggunakan `get-finding-aggregator`.

AWS CLI

Untuk mengambil konfigurasi agregasi temuan saat ini

`get-finding-aggregator` Contoh berikut mengambil konfigurasi agregasi temuan saat ini.

```
aws securityhub get-finding-aggregator \
  --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000
```

Output:

```
{
  "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000",
```

```
"FindingAggregationRegion": "us-east-1",
"RegionLinkingMode": "SPECIFIED_REGIONS",
"Regions": "us-west-1,us-west-2"
}
```

Untuk informasi selengkapnya, lihat [Melihat konfigurasi agregasi temuan saat ini](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetFindingAggregator](#) di Referensi AWS CLI Perintah.

## get-finding-history

Contoh kode berikut menunjukkan cara menggunakan `get-finding-history`.

### AWS CLI

Untuk menemukan sejarah

`get-finding-history` Contoh berikut mencapai 90 hari terakhir sejarah untuk temuan yang ditentukan. Dalam contoh ini, hasilnya terbatas pada dua catatan sejarah penemuan.

```
aws securityhub get-finding-history \
  --finding-identifier Id="arn:aws:securityhub:us-
east-1:123456789012:security-control/S3.17/finding/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",ProductArn="arn:aws:securityhub:us-east-1::product/aws/securityhub"
```

Output:

```
{
  "Records": [
    {
      "FindingIdentifier": {
        "Id": "arn:aws:securityhub:us-east-1:123456789012:security-control/
S3.17/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/
securityhub"
      },
      "UpdateTime": "2023-06-02T03:15:25.685000+00:00",
      "FindingCreated": false,
      "UpdateSource": {
        "Type": "BATCH_IMPORT_FINDINGS",
        "Identity": "arn:aws:securityhub:us-east-1::product/aws/securityhub"
      },
    },
  ],
}
```

```

    "Updates": [
      {
        "UpdatedField": "Compliance.RelatedRequirements",
        "OldValue": "[\\"NIST.800-53.r5 SC-12(2)\",\\"NIST.800-53.r5
SC-12(3)\",\\"NIST.800-53.r5 SC-12(6)\",\\"NIST.800-53.r5 CM-3(6)\",\\"NIST.800-53.r5
SC-13\",\\"NIST.800-53.r5 SC-28\",\\"NIST.800-53.r5 SC-28(1)\",\\"NIST.800-53.r5
SC-7(10)\"]",
        "NewValue": "[\\"NIST.800-53.r5 SC-12(2)\",\\"NIST.800-53.r5
CM-3(6)\",\\"NIST.800-53.r5 SC-13\",\\"NIST.800-53.r5 SC-28\",\\"NIST.800-53.r5
SC-28(1)\",\\"NIST.800-53.r5 SC-7(10)\",\\"NIST.800-53.r5 CA-9(1)\",\\"NIST.800-53.r5
SI-7(6)\",\\"NIST.800-53.r5 AU-9\"]"
      },
      {
        "UpdatedField": "LastObservedAt",
        "OldValue": "2023-06-01T09:15:38.587Z",
        "NewValue": "2023-06-02T03:15:22.946Z"
      },
      {
        "UpdatedField": "UpdatedAt",
        "OldValue": "2023-06-01T09:15:31.049Z",
        "NewValue": "2023-06-02T03:15:14.861Z"
      },
      {
        "UpdatedField": "ProcessedAt",
        "OldValue": "2023-06-01T09:15:41.058Z",
        "NewValue": "2023-06-02T03:15:25.685Z"
      }
    ]
  },
  {
    "FindingIdentifier": {
      "Id": "arn:aws:securityhub:us-east-1:123456789012:security-control/
S3.17/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/
securityhub"
    },
    "UpdateTime": "2023-05-23T02:06:51.518000+00:00",
    "FindingCreated": "true",
    "UpdateSource": {
      "Type": "BATCH_IMPORT_FINDINGS",
      "Identity": "arn:aws:securityhub:us-east-1::product/aws/securityhub"
    },
    "Updates": []
  }
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Menemukan riwayat](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetFindingHistory](#) di Referensi AWS CLI Perintah.

## get-findings

Contoh kode berikut menunjukkan cara menggunakan `get-findings`.

### AWS CLI

Contoh 1: Untuk mengembalikan temuan yang dihasilkan untuk standar tertentu

`get-findings` Contoh berikut mengembalikan temuan untuk standar PCI DSS.

```
aws securityhub get-findings \
  --filters '{"GeneratorId":[{"Value": "pci-dss", "Comparison": "PREFIX"}]}' \
  --max-items 1
```

Output:

```
{
  "Findings": [
    {
      "SchemaVersion": "2018-10-08",
      "Id": "arn:aws:securityhub:eu-central-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub",
      "GeneratorId": "pci-dss/v/3.2.1/PCI.Lambda.2",
      "AwsAccountId": "123456789012",
      "Types": [
        "Software and Configuration Checks/Industry and Regulatory Standards/PCI-DSS"
      ],
      "FindingProviderFields": {
        "Severity": {
          "Original": 0,
          "Label": "INFORMATIONAL"
        },
        "Types": [
```

```

        "Software and Configuration Checks/Industry and Regulatory
Standards/PCI-DSS"
    ]
  },
  "FirstObservedAt": "2020-06-02T14:02:49.159Z",
  "LastObservedAt": "2020-06-02T14:02:52.397Z",
  "CreatedAt": "2020-06-02T14:02:49.159Z",
  "UpdatedAt": "2020-06-02T14:02:52.397Z",
  "Severity": {
    "Original": 0,
    "Label": "INFORMATIONAL",
    "Normalized": 0
  },
  "Title": "PCI.Lambda.2 Lambda functions should be in a VPC",
  "Description": "This AWS control checks whether a Lambda function is in
a VPC.",
  "Remediation": {
    "Recommendation": {
      "Text": "For directions on how to fix this issue, please consult
the AWS Security Hub PCI DSS documentation.",
      "Url": "https://docs.aws.amazon.com/console/securityhub/
PCI.Lambda.2/remediation"
    }
  },
  "ProductFields": {
    "StandardsArn": "arn:aws:securityhub::standards/pci-dss/v/3.2.1",
    "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1",
    "ControlId": "PCI.Lambda.2",
    "RecommendationUrl": "https://docs.aws.amazon.com/console/
securityhub/PCI.Lambda.2/remediation",
    "RelatedAWSResources:0/name": "securityhub-lambda-inside-
vpc-0e904a3b",
    "RelatedAWSResources:0/type": "AWS::Config::ConfigRule",
    "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.Lambda.2",
    "aws/securityhub/SeverityLabel": "INFORMATIONAL",
    "aws/securityhub/ProductName": "Security Hub",
    "aws/securityhub/CompanyName": "AWS",
    "aws/securityhub/FindingId": "arn:aws:securityhub:eu-
central-1::product/aws/securityhub/arn:aws:securityhub:eu-
central-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  },
},

```

```

    "Resources": [
      {
        "Type": "AwsAccount",
        "Id": "AWS:::Account:123456789012",
        "Partition": "aws",
        "Region": "us-west-1"
      }
    ],
    "Compliance": {
      "Status": "PASSED",
      "RelatedRequirements": [
        "PCI DSS 1.2.1",
        "PCI DSS 1.3.1",
        "PCI DSS 1.3.2",
        "PCI DSS 1.3.4"
      ]
    },
    "WorkflowState": "NEW",
    "Workflow": {
      "Status": "NEW"
    },
    "RecordState": "ARCHIVED"
  }
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ=="
}

```

Contoh 2: Untuk mengembalikan temuan tingkat keparahan kritis yang memiliki status alur kerja NOTIFIED

get-findings Contoh berikut mengembalikan temuan yang memiliki nilai label keparahan CRITICAL dan status alur kerja NOTIFIED. Hasilnya diurutkan dalam urutan menurun berdasarkan nilai Keyakinan.

```

aws securityhub get-findings \
  --filters '{"SeverityLabel":[{"Value":
"CRITICAL","Comparison":"EQUALS"}],"WorkflowStatus":
[{"Value":"NOTIFIED","Comparison":"EQUALS"}]}' \
  --sort-criteria '{ "Field": "Confidence", "SortOrder": "desc"}' \
  --max-items 1

```

Output:



```
{
  "Findings": [
    {
      "SchemaVersion": "2018-10-08",
      "Id": "arn:aws:securityhub:us-west-1: 123456789012:subscription/cis-aws-
foundations-benchmark/v/1.2.0/1.13/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-2::product/aws/securityhub",
      "GeneratorId": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0/rule/1.13",
      "AwsAccountId": "123456789012",
      "Types": [
        "Software and Configuration Checks/Industry and Regulatory
Standards/CIS AWS Foundations Benchmark"
      ],
      "FindingProviderFields" {
        "Severity": {
          "Original": 90,
          "Label": "CRITICAL"
        },
        "Types": [
          "Software and Configuration Checks/Industry and Regulatory
Standards/CIS AWS Foundations Benchmark"
        ]
      },
      "FirstObservedAt": "2020-05-21T20:16:34.752Z",
      "LastObservedAt": "2020-06-09T08:16:37.171Z",
      "CreatedAt": "2020-05-21T20:16:34.752Z",
      "UpdatedAt": "2020-06-09T08:16:36.430Z",
      "Severity": {
        "Original": 90,
        "Label": "CRITICAL",
        "Normalized": 90
      },
      "Title": "1.13 Ensure MFA is enabled for the \"root\" account",
      "Description": "The root account is the most privileged user in an AWS
account. MFA adds an extra layer of protection on top of a user name and password.
With MFA enabled, when a user signs in to an AWS website, they will be prompted for
their user name and password as well as for an authentication code from their AWS
MFA device.",
      "Remediation": {
        "Recommendation": {
          "Text": "For directions on how to fix this issue, please consult
the AWS Security Hub CIS documentation.",

```

```
        "Url": "https://docs.aws.amazon.com/console/securityhub/
standards-cis-1.13/remediation"
    }
  },
  "ProductFields": {
    "StandardsGuideArn": "arn:aws:securityhub::ruleset/cis-aws-
foundations-benchmark/v/1.2.0",
    "StandardsGuideSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/cis-aws-foundations-benchmark/v/1.2.0",
    "RuleId": "1.13",
    "RecommendationUrl": "https://docs.aws.amazon.com/console/
securityhub/standards-cis-1.13/remediation",
    "RelatedAWSResources:0/name": "securityhub-root-account-mfa-
enabled-5pftha",
    "RelatedAWSResources:0/type": "AWS::Config::ConfigRule",
    "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/cis-aws-foundations-benchmark/v/1.2.0/1.13",
    "aws/securityhub/SeverityLabel": "CRITICAL",
    "aws/securityhub/ProductName": "Security Hub",
    "aws/securityhub/CompanyName": "AWS",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-
west-1::product/aws/securityhub/arn:aws:securityhub:us-
west-1:123456789012:subscription/cis-aws-foundations-benchmark/v/1.2.0/1.13/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  },
  "Resources": [
    {
      "Type": "AwsAccount",
      "Id": "AWS:::Account:123456789012",
      "Partition": "aws",
      "Region": "us-west-1"
    }
  ],
  "Compliance": {
    "Status": "FAILED"
  },
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NOTIFIED"
  },
  "RecordState": "ACTIVE"
}
]
```

```
}
```

Untuk informasi selengkapnya, lihat [Memfilter dan mengelompokkan temuan](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetFindings](#) di Referensi AWS CLI Perintah.

## get-insight-results

Contoh kode berikut menunjukkan cara menggunakan `get-insight-results`.

### AWS CLI

Untuk mengambil hasil untuk wawasan

`get-insight-results` Contoh berikut mengembalikan daftar hasil wawasan untuk wawasan dengan ARN yang ditentukan.

```
aws securityhub get-insight-results \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

### Output:

```
{
  "InsightResults": {
    "GroupByAttribute": "ResourceId",
    "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
    "ResultValues": [
      {
        "Count": 10,
        "GroupByAttributeValue": "AWS:::Account:123456789111"
      },
      {
        "Count": 3,
        "GroupByAttributeValue": "AWS:::Account:123456789222"
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Melihat dan mengambil tindakan terhadap hasil dan temuan wawasan](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetInsightResults](#) di Referensi AWS CLI Perintah.

## get-insights

Contoh kode berikut menunjukkan cara menggunakan `get-insights`.

### AWS CLI

Untuk mengambil detail tentang wawasan

`get-insights` Contoh berikut mengambil detail konfigurasi untuk wawasan dengan ARN yang ditentukan.

```
aws securityhub get-insights \
  --insight-arns "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
  "Insights": [
    {
      "Filters": {
        "ResourceType": [
          {
            "Comparison": "EQUALS",
            "Value": "AwsIamRole"
          }
        ],
        "SeverityLabel": [
          {
            "Comparison": "EQUALS",
            "Value": "CRITICAL"
          }
        ]
      },
      "GroupByAttribute": "ResourceId",
      "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
```

```
        "Name": "Critical role findings"
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [Wawasan di AWS Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetInsights](#) di Referensi AWS CLI Perintah.

## get-invitations-count

Contoh kode berikut menunjukkan cara menggunakan `get-invitations-count`.

### AWS CLI

Untuk mengambil jumlah undangan yang tidak diterima

`get-invitations-count` Contoh berikut mengambil jumlah undangan yang ditolak atau tidak ditanggapi oleh akun yang meminta.

```
aws securityhub get-invitations-count
```

Output:

```
{
  "InvitationsCount": 3
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetInvitationsCount](#) di Referensi AWS CLI Perintah.

## get-master-account

Contoh kode berikut menunjukkan cara menggunakan `get-master-account`.

### AWS CLI

Untuk mengambil informasi tentang akun administrator

`get-master-account` Contoh berikut mengambil informasi tentang akun administrator untuk akun yang meminta.

```
aws securityhub get-master-account
```

Output:

```
{
  "Master": {
    "AccountId": "123456789012",
    "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
    "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
    "MemberStatus": "ASSOCIATED"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetMasterAccount](#) di Referensi AWS CLI Perintah.

## get-members

Contoh kode berikut menunjukkan cara menggunakan `get-members`.

AWS CLI

Untuk mengambil informasi tentang akun anggota yang dipilih

`get-members` Contoh berikut mengambil informasi tentang akun anggota yang ditentukan.

```
aws securityhub get-members \
  --account-ids "444455556666" "777788889999"
```

Output:

```
{
  "Members": [
    {
      "AccountId": "123456789111",
```

```

    "AdministratorId": "123456789012",
    "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
    "MasterId": "123456789012",
    "MemberStatus": "ASSOCIATED",
    "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
  },
  {
    "AccountId": "123456789222",
    "AdministratorId": "123456789012",
    "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
    "MasterId": "123456789012",
    "MemberStatus": "ASSOCIATED",
    "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
  }
],
"UnprocessedAccounts": [ ]
}

```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetMembers](#) di Referensi AWS CLI Perintah.

## get-security-control-definition

Contoh kode berikut menunjukkan cara menggunakan `get-security-control-definition`.

### AWS CLI

Untuk mendapatkan detail definisi kontrol keamanan

`get-security-control-definition` Contoh berikut mengambil rincian definisi untuk kontrol keamanan Security Hub. Detail termasuk judul kontrol, deskripsi, ketersediaan Wilayah, parameter, dan informasi lainnya.

```
aws securityhub get-security-control-definition \
  --security-control-id ACM.1
```

Output:

```
{
  "SecurityControlDefinition": {
```

```

    "SecurityControlId": "ACM.1",
    "Title": "Imported and ACM-issued certificates should be renewed after a
specified time period",
    "Description": "This control checks whether an AWS Certificate Manager
(ACM) certificate is renewed within the specified time period. It checks both
imported certificates and certificates provided by ACM. The control fails if the
certificate isn't renewed within the specified time period. Unless you provide a
custom parameter value for the renewal period, Security Hub uses a default value of
30 days.",
    "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/ACM.1/
remediation",
    "SeverityRating": "MEDIUM",
    "CurrentRegionAvailability": "AVAILABLE",
    "ParameterDefinitions": {
      "daysToExpiration": {
        "Description": "Number of days within which the ACM certificate must
be renewed",
        "ConfigurationOptions": {
          "Integer": {
            "DefaultValue": 30,
            "Min": 14,
            "Max": 365
          }
        }
      }
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Parameter kontrol khusus](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [GetSecurityControlDefinition](#) di Referensi AWS CLI Perintah.

## invite-members

Contoh kode berikut menunjukkan cara menggunakan `invite-members`.

### AWS CLI

Untuk mengirim undangan ke akun anggota

`invite-members` Contoh berikut mengirimkan undangan ke akun anggota yang ditentukan.



```
aws securityhub invite-members \  
  --account-ids "123456789111" "123456789222"
```

Output:

```
{  
  "UnprocessedAccounts": []  
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [InviteMembers](#) di Referensi AWS CLI Perintah.

## list-automation-rules

Contoh kode berikut menunjukkan cara menggunakan `list-automation-rules`.

AWS CLI

Untuk melihat daftar aturan otomatisasi

`list-automation-rules` Contoh berikut mencantumkan aturan otomatisasi untuk AWS akun. Hanya akun administrator Security Hub yang dapat menjalankan perintah ini.

```
aws securityhub list-automation-rules \  
  --max-results 3 \  
  --next-token NULL
```

Output:

```
{  
  "AutomationRulesMetadata": [  
    {  
      "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "RuleStatus": "ENABLED",  
      "RuleOrder": 1,  
      "RuleName": "Suppress informational findings",  
      "Description": "Suppress GuardDuty findings with Informational  
severity",  
      "IsTerminal": false,  
    }  
  ]  
}
```

```
    "CreatedAt": "2023-05-31T17:56:14.837000+00:00",
    "UpdatedAt": "2023-05-31T17:59:38.466000+00:00",
    "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
  },
  {
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "RuleStatus": "ENABLED",
    "RuleOrder": 1,
    "RuleName": "sample rule",
    "Description": "A sample rule",
    "IsTerminal": false,
    "CreatedAt": "2023-07-15T23:37:20.223000+00:00",
    "UpdatedAt": "2023-07-15T23:37:20.223000+00:00",
    "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
  },
  {
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "RuleStatus": "ENABLED",
    "RuleOrder": 1,
    "RuleName": "sample rule",
    "Description": "A sample rule",
    "IsTerminal": false,
    "CreatedAt": "2023-07-15T23:45:25.126000+00:00",
    "UpdatedAt": "2023-07-15T23:45:25.126000+00:00",
    "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Melihat aturan otomatisasi](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListAutomationRules](#) di Referensi AWS CLI Perintah.

## list-configuration-policies

Contoh kode berikut menunjukkan cara menggunakan `list-configuration-policies`.

### AWS CLI

Untuk membuat daftar ringkasan kebijakan konfigurasi

`list-configuration-policies` Contoh berikut mencantumkan ringkasan kebijakan konfigurasi untuk organisasi.

```
aws securityhub list-configuration-policies \  
  --max-items 3
```

Output:

```
{  
  "ConfigurationPolicySummaries": [  
    {  
      "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Name": "SampleConfigurationPolicy1",  
      "Description": "SampleDescription1",  
      "UpdatedAt": "2023-09-26T21:08:36.214000+00:00",  
      "ServiceEnabled": true  
    },  
    {  
      "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Name": "SampleConfigurationPolicy2",  
      "Description": "SampleDescription2"  
      "UpdatedAt": "2023-11-28T19:26:25.207000+00:00",  
      "ServiceEnabled": true  
    },  
    {  
      "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
      "Name": "SampleConfigurationPolicy3",  
      "Description": "SampleDescription3",  
      "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",  
      "ServiceEnabled": true  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListConfigurationPolicies](#) di Referensi AWS CLI Perintah.

## list-configuration-policy-associations

Contoh kode berikut menunjukkan cara menggunakan `list-configuration-policy-associations`.

### AWS CLI

Untuk membuat daftar asosiasi konfigurasi

`list-configuration-policy-associations` Contoh berikut mencantumkan ringkasan asosiasi konfigurasi untuk organisasi. Responsnya mencakup asosiasi dengan kebijakan konfigurasi dan perilaku yang dikelola sendiri.

```
aws securityhub list-configuration-policy-associations \  
  --association-type "APPLIED" \  
  --max-items 4
```

### Output:

```
{  
  "ConfigurationPolicyAssociationSummaries": [  
    {  
      "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "TargetId": "r-1ab2",  
      "TargetType": "ROOT",  
      "AssociationType": "APPLIED",  
      "UpdatedAt": "2023-11-28T19:26:49.417000+00:00",  
      "AssociationStatus": "FAILED",  
      "AssociationStatusMessage": "Policy association failed because 2  
organizational units or accounts under this root failed."  
    },  
    {  
      "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "TargetId": "ou-1ab2-c3de4f5g",  
      "TargetType": "ORGANIZATIONAL_UNIT",  
      "AssociationType": "APPLIED",  
      "UpdatedAt": "2023-09-26T21:14:05.283000+00:00",  
      "AssociationStatus": "FAILED",  
      "AssociationStatusMessage": "One or more children under this target  
failed association."  
    },  
    {  
      "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
```

```

    "TargetId": "ou-6hi7-8j91k12m",
    "TargetType": "ORGANIZATIONAL_UNIT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
    "AssociationStatus": "SUCCESS",
    "AssociationStatusMessage": "Association applied successfully on this
target."
  },
  {
    "ConfigurationPolicyId": "SELF_MANAGED_SECURITY_HUB",
    "TargetId": "111122223333",
    "TargetType": "ACCOUNT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-11-28T22:01:26.409000+00:00",
    "AssociationStatus": "SUCCESS"
  }
}

```

Untuk informasi selengkapnya, lihat [Melihat kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListConfigurationPolicyAssociations](#) di Referensi AWS CLI Perintah.

## list-enabled-products-for-import

Contoh kode berikut menunjukkan cara menggunakan `list-enabled-products-for-import`.

### AWS CLI

Untuk mengembalikan daftar integrasi produk yang diaktifkan

`list-enabled-products-for-import` Contoh berikut mengembalikan daftar ARNS berlangganan untuk integrasi produk yang saat ini diaktifkan.

```
aws securityhub list-enabled-products-for-import
```

### Output:

```

{
  "ProductSubscriptions": [ "arn:aws:securityhub:us-west-1:123456789012:product-
subscription/crowdstrike/crowdstrike-falcon", "arn:aws:securityhub:us-
west-1:123456789012:product-subscription/aws/securityhub" ]
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Mengelola integrasi produk](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListEnabledProductsForImport](#) di Referensi AWS CLI Perintah.

## **list-finding-aggregators**

Contoh kode berikut menunjukkan cara menggunakan `list-finding-aggregators`.

### AWS CLI

Untuk daftar widget yang tersedia

`list-finding-aggregators` Contoh berikut mengembalikan ARN dari konfigurasi agregasi temuan.

```
aws securityhub list-finding-aggregators
```

Output:

```
{
  "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-
aggregator/123e4567-e89b-12d3-a456-426652340000"
}
```

Untuk informasi selengkapnya, lihat [Melihat konfigurasi agregasi temuan saat ini](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListFindingAggregators](#) di Referensi AWS CLI Perintah.

## **list-invitations**

Contoh kode berikut menunjukkan cara menggunakan `list-invitations`.

### AWS CLI

Untuk menampilkan daftar undangan

`list-invitations` Contoh berikut mengambil daftar undangan yang dikirim ke akun yang meminta.

```
aws securityhub list-invitations
```

Output:

```
{
  "Invitations": [
    {
      "AccountId": "123456789012",
      "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
      "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
      "MemberStatus": "ASSOCIATED"
    }
  ],
}
```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListInvitations](#) di Referensi AWS CLI Perintah.

## list-members

Contoh kode berikut menunjukkan cara menggunakan `list-members`.

### AWS CLI

Untuk mengambil daftar akun anggota

`list-members` Contoh berikut mengembalikan daftar akun anggota untuk akun administrator meminta.

```
aws securityhub list-members
```

Output:

```
{
  "Members": [
    {
      "AccountId": "123456789111",
      "AdministratorId": "123456789012",
      "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
```

```

    "MasterId": "123456789012",
    "MemberStatus": "ASSOCIATED",
    "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
  },
  {
    "AccountId": "123456789222",
    "AdministratorId": "123456789012",
    "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
    "MasterId": "123456789012",
    "MemberStatus": "ASSOCIATED",
    "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
  }
],
}

```

Untuk informasi selengkapnya, lihat [Mengelola akun administrator dan anggota](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListMembers](#) di Referensi AWS CLI Perintah.

## list-organization-admin-accounts

Contoh kode berikut menunjukkan cara menggunakan `list-organization-admin-accounts`.

### AWS CLI

Untuk mencantumkan akun administrator Security Hub yang ditunjuk

`list-organization-admin-accounts` Contoh berikut mencantumkan akun administrator Security Hub untuk organisasi.

```
aws securityhub list-organization-admin-accounts
```

Output:

```

{
  AdminAccounts": [
    { "AccountId": "777788889999" },
    { "Status": "ENABLED" }
  ]
}

```



Untuk informasi selengkapnya, lihat [Menetapkan akun administrator Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListOrganizationAdminAccounts](#) di Referensi AWS CLI Perintah.

## list-security-control-definitions

Contoh kode berikut menunjukkan cara menggunakan `list-security-control-definitions`.

### AWS CLI

Contoh 1: Untuk daftar semua kontrol keamanan yang tersedia

`list-security-control-definitions` Contoh berikut mencantumkan kontrol keamanan yang tersedia di semua standar Security Hub. Contoh ini membatasi hasil hingga tiga kontrol.

```
aws securityhub list-security-control-definitions \  
  --max-items 3
```

### Output:

```
{  
  "SecurityControlDefinitions": [  
    {  
      "SecurityControlId": "ACM.1",  
      "Title": "Imported and ACM-issued certificates should be renewed after a  
specified time period",  
      "Description": "This control checks whether an AWS Certificate Manager  
(ACM) certificate is renewed within the specified time period. It checks both  
imported certificates and certificates provided by ACM. The control fails if the  
certificate isn't renewed within the specified time period. Unless you provide a  
custom parameter value for the renewal period, Security Hub uses a default value of  
30 days.",  
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/  
ACM.1/remediation",  
      "SeverityRating": "MEDIUM",  
      "CurrentRegionAvailability": "AVAILABLE",  
      "CustomizableProperties": [  
        "Parameters"  
      ]  
    },  
    {  
      "SecurityControlId": "ACM.2",
```

```

        "Title": "RSA certificates managed by ACM should use a key length of at
least 2,048 bits",
        "Description": "This control checks whether RSA certificates managed by
AWS Certificate Manager use a key length of at least 2,048 bits. The control fails
if the key length is smaller than 2,048 bits.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
ACM.2/remediation",
        "SeverityRating": "HIGH",
        "CurrentRegionAvailability": "AVAILABLE",
        "CustomizableProperties": []
    },
    {
        "SecurityControlId": "APIGateway.1",
        "Title": "API Gateway REST and WebSocket API execution logging should be
enabled",
        "Description": "This control checks whether all stages of an Amazon
API Gateway REST or WebSocket API have logging enabled. The control fails if
the 'loggingLevel' isn't 'ERROR' or 'INFO' for all stages of the API. Unless you
provide custom parameter values to indicate that a specific log type should be
enabled, Security Hub produces a passed finding if the logging level is either
'ERROR' or 'INFO'.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
APIGateway.1/remediation",
        "SeverityRating": "MEDIUM",
        "CurrentRegionAvailability": "AVAILABLE",
        "CustomizableProperties": [
            "Parameters"
        ]
    }
],
"NextToken": "U2FsdGVkX1/UpRCPzxVbkDeHikDXbDxfgJZ1w2RG1XWsFPTMTIQPVE0m/
FduIGxS70bRtAbaUt/8/RCQcg2PU0YXI20hH/GrHo0Tgv+TSm0qvQVFhkJepWmqh
+NYawjocVBeos6xzn/8qnbF9IuwGg=="
}

```

Untuk informasi selengkapnya, lihat [Melihat detail untuk standar](#) di Panduan Pengguna AWS Security Hub.

Contoh 2: Untuk daftar kontrol keamanan yang tersedia untuk standar tertentu

`list-security-control-definitions` Contoh berikut mencantumkan kontrol keamanan yang tersedia untuk CIS AWS Foundations Benchmark v1.4.0. Contoh ini membatasi hasil hingga tiga kontrol.

```
aws securityhub list-security-control-definitions \  
  --standards-arn "arn:aws:securityhub:us-east-1::standards/cis-aws-foundations-  
benchmark/v/1.4.0" \  
  --max-items 3
```

## Output:

```
{  
  "SecurityControlDefinitions": [  
    {  
      "SecurityControlId": "CloudTrail.1",  
      "Title": "CloudTrail should be enabled and configured with at least one  
multi-Region trail that includes read and write management events",  
      "Description": "This AWS control checks that there is at least one  
multi-region AWS CloudTrail trail includes read and write management events.",  
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/  
CloudTrail.1/remediation",  
      "SeverityRating": "HIGH",  
      "CurrentRegionAvailability": "AVAILABLE",  
      "CustomizableProperties": []  
    },  
    {  
      "SecurityControlId": "CloudTrail.2",  
      "Title": "CloudTrail should have encryption at-rest enabled",  
      "Description": "This AWS control checks whether AWS CloudTrail is  
configured to use the server side encryption (SSE) AWS Key Management Service (AWS  
KMS) customer master key (CMK) encryption. The check will pass if the KmsKeyId is  
defined.",  
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/  
CloudTrail.2/remediation",  
      "SeverityRating": "MEDIUM",  
      "CurrentRegionAvailability": "AVAILABLE",  
      "CustomizableProperties": []  
    },  
    {  
      "SecurityControlId": "CloudTrail.4",  
      "Title": "CloudTrail log file validation should be enabled",  
      "Description": "This AWS control checks whether CloudTrail log file  
validation is enabled.",  
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/  
CloudTrail.4/remediation",  
      "SeverityRating": "MEDIUM",  
      "CurrentRegionAvailability": "AVAILABLE",
```

```

    "CustomizableProperties": []
  }
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAzfQ=="
}

```

Untuk informasi selengkapnya, lihat [Melihat detail untuk standar](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [ListSecurityControlDefinitions](#) di Referensi AWS CLI Perintah.

## list-standards-control-associations

Contoh kode berikut menunjukkan cara menggunakan `list-standards-control-associations`.

### AWS CLI

Untuk mendapatkan status pemberdayaan kontrol di setiap standar yang diaktifkan

`list-standards-control-associations` Contoh berikut mencantumkan status pemberdayaan CloudTrail .1 di setiap standar yang diaktifkan.

```

aws securityhub list-standards-control-associations \
  --security-control-id CloudTrail.1

```

### Output:

```

{
  "StandardsControlAssociationSummaries": [
    {
      "StandardsArn": "arn:aws:securityhub:us-east-2::standards/nist-800-53/v/5.0.0",
      "SecurityControlId": "CloudTrail.1",
      "SecurityControlArn": "arn:aws:securityhub:us-east-2:123456789012:security-control/CloudTrail.1",
      "AssociationStatus": "ENABLED",
      "RelatedRequirements": [
        "NIST.800-53.r5 AC-2(4)",
        "NIST.800-53.r5 AC-4(26)",
        "NIST.800-53.r5 AC-6(9)",
        "NIST.800-53.r5 AU-10",
        "NIST.800-53.r5 AU-12",
      ]
    }
  ]
}

```

```

        "NIST.800-53.r5 AU-2",
        "NIST.800-53.r5 AU-3",
        "NIST.800-53.r5 AU-6(3)",
        "NIST.800-53.r5 AU-6(4)",
        "NIST.800-53.r5 AU-14(1)",
        "NIST.800-53.r5 CA-7",
        "NIST.800-53.r5 SC-7(9)",
        "NIST.800-53.r5 SI-3(8)",
        "NIST.800-53.r5 SI-4(20)",
        "NIST.800-53.r5 SI-7(8)",
        "NIST.800-53.r5 SA-8(22)"
    ],
    "UpdatedAt": "2023-05-15T17:52:21.304000+00:00",
    "StandardsControlTitle": "CloudTrail should be enabled and configured
with at least one multi-Region trail that includes read and write management
events",
    "StandardsControlDescription": "This AWS control checks that there is
at least one multi-region AWS CloudTrail trail includes read and write management
events."
  },
  {
    "StandardsArn": "arn:aws:securityhub::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",
    "SecurityControlId": "CloudTrail.1",
    "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
    "AssociationStatus": "ENABLED",
    "RelatedRequirements": [
      "CIS AWS Foundations 2.1"
    ],
    "UpdatedAt": "2020-02-10T21:22:53.998000+00:00",
    "StandardsControlTitle": "Ensure CloudTrail is enabled in all regions",
    "StandardsControlDescription": "AWS CloudTrail is a web service that
records AWS API calls for your account and delivers log files to you. The recorded
information includes the identity of the API caller, the time of the API call,
the source IP address of the API caller, the request parameters, and the response
elements returned by the AWS service."
  },
  {
    "StandardsArn": "arn:aws:securityhub:us-east-2::standards/aws-
foundational-security-best-practices/v/1.0.0",
    "SecurityControlId": "CloudTrail.1",
    "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",

```

```

    "AssociationStatus": "DISABLED",
    "RelatedRequirements": [],
    "UpdatedAt": "2023-05-15T19:31:52.671000+00:00",
    "UpdatedReason": "Alternative compensating controls are in place",
    "StandardsControlTitle": "CloudTrail should be enabled and configured
with at least one multi-Region trail that includes read and write management
events",
    "StandardsControlDescription": "This AWS control checks that there is
at least one multi-region AWS CloudTrail trail includes read and write management
events."
  },
  {
    "StandardsArn": "arn:aws:securityhub:us-east-2::standards/cis-aws-
foundations-benchmark/v/1.4.0",
    "SecurityControlId": "CloudTrail.1",
    "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
    "AssociationStatus": "ENABLED",
    "RelatedRequirements": [
      "CIS AWS Foundations Benchmark v1.4.0/3.1"
    ],
    "UpdatedAt": "2022-11-10T15:40:36.021000+00:00",
    "StandardsControlTitle": "Ensure CloudTrail is enabled in all regions",
    "StandardsControlDescription": "AWS CloudTrail is a web service that
records AWS API calls for your account and delivers log files to you. The recorded
information includes the identity of the API caller, the time of the API call,
the source IP address of the API caller, the request parameters, and the response
elements returned by the AWS service. CloudTrail provides a history of AWS API
calls for an account, including API calls made via the Management Console, SDKs,
command line tools, and higher-level AWS services (such as CloudFormation)."
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mengaktifkan dan menonaktifkan kontrol dalam standar tertentu di Panduan Pengguna AWS Security Hub](#).

- Untuk detail API, lihat [ListStandardsControlAssociations](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk mengambil tag yang ditetapkan ke sumber daya

`list-tags-for-resource` Contoh berikut mengembalikan tag yang ditetapkan ke sumber daya hub tertentu.

```
aws securityhub list-tags-for-resource \  
  --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default"
```

Output:

```
{  
  "Tags": {  
    "Department" : "Operations",  
    "Area" : "USMidwest"  
  }  
}
```

Untuk informasi selengkapnya, lihat [AWS:SecurityHub:::Hub](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **start-configuration-policy-association**

Contoh kode berikut menunjukkan cara menggunakan `start-configuration-policy-association`.

### AWS CLI

Contoh 1: Untuk mengaitkan kebijakan konfigurasi

`start-configuration-policy-association` Contoh berikut mengaitkan kebijakan konfigurasi yang ditentukan dengan unit organisasi yang ditentukan. Konfigurasi dapat dikaitkan dengan akun target, unit organisasi, atau root.

```
aws securityhub start-configuration-policy-association \  
  --configuration-policy-identifier "arn:aws:securityhub:eu-  
central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333" \  
  --target-id "i-12345678901234567890123456789012"
```

```
--target '{"OrganizationalUnitId": "ou-6hi7-8j91k12m"}'
```

### Output:

```
{
  "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
  "TargetId": "ou-6hi7-8j91k12m",
  "TargetType": "ORGANIZATIONAL_UNIT",
  "AssociationType": "APPLIED",
  "UpdatedAt": "2023-11-29T17:40:52.468000+00:00",
  "AssociationStatus": "PENDING"
}
```

Untuk informasi selengkapnya, lihat [Membuat dan mengaitkan kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

Contoh 2: Untuk mengaitkan konfigurasi yang dikelola sendiri

`start-configuration-policy-association` Contoh berikut mengaitkan konfigurasi yang dikelola sendiri dengan akun yang ditentukan.

```
aws securityhub start-configuration-policy-association \
  --configuration-policy-identifier "SELF_MANAGED_SECURITY_HUB" \
  --target '{"OrganizationalUnitId": "123456789012"}'
```

### Output:

```
{
  "ConfigurationPolicyId": "SELF_MANAGED_SECURITY_HUB",
  "TargetId": "123456789012",
  "TargetType": "ACCOUNT",
  "AssociationType": "APPLIED",
  "UpdatedAt": "2023-11-29T17:40:52.468000+00:00",
  "AssociationStatus": "PENDING"
}
```

Untuk informasi selengkapnya, lihat [Membuat dan mengaitkan kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [StartConfigurationPolicyAssociation](#) di Referensi AWS CLI Perintah.



## start-configuration-policy-disassociation

Contoh kode berikut menunjukkan cara menggunakan `start-configuration-policy-disassociation`.

### AWS CLI

Contoh 1: Untuk memisahkan kebijakan konfigurasi

`start-configuration-policy-disassociation` Contoh berikut memisahkan kebijakan konfigurasi dari unit organisasi yang ditentukan. Konfigurasi dapat terlepas dari akun target, unit organisasi, atau root.

```
aws securityhub start-configuration-policy-disassociation \  
  --configuration-policy-identifier "arn:aws:securityhub:eu-  
central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333" \  
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91k12m"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memisahkan konfigurasi dari akun dan OU](#) di Panduan Pengguna AWS Security Hub.

Contoh 2: Untuk memisahkan konfigurasi yang dikelola sendiri

`start-configuration-policy-disassociation` Contoh berikut memisahkan konfigurasi yang dikelola sendiri dari akun yang ditentukan.

```
aws securityhub start-configuration-policy-disassociation \  
  --configuration-policy-identifier "SELF_MANAGED_SECURITY_HUB" \  
  --target '{"AccountId": "123456789012"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memisahkan konfigurasi dari akun dan OU](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [StartConfigurationPolicyDisassociation](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

## AWS CLI

Untuk menetapkan tag ke sumber daya

`tag-resource` Contoh berikut menetapkan nilai untuk tag Departemen dan Area ke sumber daya hub yang ditentukan.

```
aws securityhub tag-resource \  
  --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default" \  
  --tags '{"Department":"Operations", "Area":"USMidwest"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [AWS:SecurityHub:::Hub](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

## AWS CLI

Untuk menghapus nilai tag dari sumber daya

`untag-resource` Contoh berikut menghapus tag Department dari sumber daya hub yang ditentukan.

```
aws securityhub untag-resource \  
  --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default" \  
  --tag-keys "Department"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [AWS:SecurityHub:::Hub](#) di Panduan AWS CloudFormation Pengguna.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-action-target**

Contoh kode berikut menunjukkan cara menggunakan `update-action-target`.

## AWS CLI

Untuk memperbarui tindakan kustom

`update-action-target` Contoh berikut memperbarui nama tindakan kustom yang diidentifikasi oleh ARN yang ditentukan.

```
aws securityhub update-action-target \  
  --action-target-arn "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
Remediation" \  
  --name "Send to remediation"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat tindakan kustom dan mengaitkannya dengan aturan CloudWatch Peristiwa](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateActionTarget](#) di Referensi AWS CLI Perintah.

## **update-configuration-policy**

Contoh kode berikut menunjukkan cara menggunakan `update-configuration-policy`.

## AWS CLI

Untuk memperbarui kebijakan konfigurasi

`update-configuration-policy` Contoh berikut memperbarui kebijakan konfigurasi yang ada untuk menggunakan pengaturan yang ditentukan.

```
aws securityhub update-configuration-policy \  
  --identifier "arn:aws:securityhub:eu-central-1:508236694226:configuration-  
policy/09f37766-57d8-4ede-9d33-5d8b0fecf70e" \  
  --name "SampleConfigurationPolicyUpdated" \  
  --description "SampleDescriptionUpdated" \  
  --configuration-policy '{"SecurityHub": {"ServiceEnabled":  
true, "EnabledStandardIdentifiers": ["arn:aws:securityhub:eu-  
central-1::standards/aws-foundational-security-best-practices/  
v/1.0.0"], "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/  
v/1.2.0"], "SecurityControlsConfiguration": {"DisabledSecurityControlIdentifiers":  
["CloudWatch.1"], "SecurityControlCustomParameters": [{"SecurityControlId":
```

```
"ACM.1", "Parameters": {"daysToExpiration": {"ValueType": "CUSTOM", "Value":
{"Integer": 21}}}}]]}}]' \
--updated-reason "Disabling CloudWatch.1 and changing parameter value"
```

## Output:

```
{
  "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Name": "SampleConfigurationPolicyUpdated",
  "Description": "SampleDescriptionUpdated",
  "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
  "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
  "ConfigurationPolicy": {
    "SecurityHub": {
      "ServiceEnabled": true,
      "EnabledStandardIdentifiers": [
        "arn:aws:securityhub:eu-central-1::standards/aws-foundational-
security-best-practices/v/1.0.0",
        "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"
      ],
      "SecurityControlsConfiguration": {
        "DisabledSecurityControlIdentifiers": [
          "CloudWatch.1"
        ],
        "SecurityControlCustomParameters": [
          {
            "SecurityControlId": "ACM.1",
            "Parameters": {
              "daysToExpiration": {
                "ValueType": "CUSTOM",
                "Value": {
                  "Integer": 21
                }
              }
            }
          }
        ]
      }
    }
  }
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Memperbarui kebijakan konfigurasi Security Hub](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateConfigurationPolicy](#) di Referensi AWS CLI Perintah.

## update-finding-aggregator

Contoh kode berikut menunjukkan cara menggunakan `update-finding-aggregator`.

### AWS CLI

Untuk memperbarui konfigurasi agregasi temuan saat ini

`update-finding-aggregator` Contoh berikut mengubah konfigurasi agregasi temuan untuk menautkan dari Wilayah yang dipilih. Dijalankan dari US East (Virginia), yang merupakan Wilayah agregasi. Ini memilih AS Barat (California N.) dan AS Barat (Oregon) sebagai Wilayah terkait.

```
aws securityhub update-finding-aggregator \
  --region us-east-1 \
  --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-
  aggregator/123e4567-e89b-12d3-a456-426652340000 \
  --region-linking-mode SPECIFIED_REGIONS \
  --regions us-west-1,us-west-2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memperbarui konfigurasi agregasi pencarian](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateFindingAggregator](#) di Referensi AWS CLI Perintah.

## update-insight

Contoh kode berikut menunjukkan cara menggunakan `update-insight`.

### AWS CLI

Contoh 1: Untuk mengubah filter untuk wawasan kustom

`update-insight` Contoh berikut mengubah filter untuk wawasan kustom. Wawasan yang diperbarui mencari temuan dengan tingkat keparahan tinggi yang terkait dengan AWS peran.

```
aws securityhub update-insight \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
  --filters '{"ResourceType": [{ "Comparison": "EQUALS", "Value": "AwsIamRole"}],
"SeverityLabel": [{"Comparison": "EQUALS", "Value": "HIGH"}]}' \
  --name "High severity role findings"
```

Contoh 2: Untuk mengubah atribut pengelompokan untuk wawasan kustom

update-insight Contoh berikut mengubah atribut pengelompokan untuk wawasan kustom dengan ARN yang ditentukan. Atribut pengelompokan baru adalah ID sumber daya.

```
aws securityhub update-insight \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
  --group-by-attribute "ResourceId" \
  --name "Critical role findings"
```

Output:

```
{
  "Insights": [
    {
      "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
      "Name": "Critical role findings",
      "Filters": {
        "SeverityLabel": [
          {
            "Value": "CRITICAL",
            "Comparison": "EQUALS"
          }
        ],
        "ResourceType": [
          {
            "Value": "AwsIamRole",
            "Comparison": "EQUALS"
          }
        ]
      },
      "GroupByAttribute": "ResourceId"
    }
  ]
}
```

```
}  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengelola wawasan khusus](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateInsight](#) di Referensi AWS CLI Perintah.

## update-organization-configuration

Contoh kode berikut menunjukkan cara menggunakan `update-organization-configuration`.

### AWS CLI

Untuk memperbarui cara Security Hub dikonfigurasi untuk organisasi

`update-organization-configuration` Contoh berikut menentukan bahwa Security Hub harus menggunakan konfigurasi pusat untuk mengkonfigurasi organisasi. Setelah menjalankan perintah ini, administrator Security Hub yang didelegasikan dapat membuat dan mengelola kebijakan konfigurasi untuk mengonfigurasi organisasi. Administrator yang didelegasikan juga dapat menggunakan perintah ini untuk beralih dari konfigurasi pusat ke lokal. Jika konfigurasi lokal adalah tipe konfigurasi, administrator yang didelegasikan dapat memilih apakah akan mengaktifkan Security Hub secara otomatis dan standar keamanan default di akun organisasi baru.

```
aws securityhub update-organization-configuration \  
  --no-auto-enable \  
  --organization-configuration '{"ConfigurationType": "CENTRAL"}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola akun dengan AWS Organizations](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateOrganizationConfiguration](#) di Referensi AWS CLI Perintah.

## update-security-control

Contoh kode berikut menunjukkan cara menggunakan `update-security-control`.

## AWS CLI

Untuk memperbarui properti kontrol keamanan

`update-security-control` Contoh berikut menentukan nilai kustom untuk parameter kontrol keamanan Security Hub.

```
aws securityhub update-security-control \  
  --security-control-id ACM.1 \  
  --parameters '{"daysToExpiration": {"ValueType": "CUSTOM", "Value": {"Integer":  
15}}}' \  
  --last-update-reason "Internal compliance requirement"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Parameter kontrol khusus](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateSecurityControl](#) di Referensi AWS CLI Perintah.

## **update-security-hub-configuration**

Contoh kode berikut menunjukkan cara menggunakan `update-security-hub-configuration`.

### AWS CLI

Untuk memperbarui konfigurasi Security Hub

`update-security-hub-configuration` Contoh berikut mengonfigurasi Security Hub untuk secara otomatis mengaktifkan kontrol baru untuk standar yang diaktifkan.

```
aws securityhub update-security-hub-configuration \  
  --auto-enable-controls
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaktifkan kontrol baru secara otomatis](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateSecurityHubConfiguration](#) di Referensi AWS CLI Perintah.



## update-standards-control

Contoh kode berikut menunjukkan cara menggunakan `update-standards-control`.

### AWS CLI

Contoh 1: Untuk menonaktifkan kontrol

`update-standards-control` Contoh berikut menonaktifkan PCI. AutoScaling.1 kontrol.

```
aws securityhub update-standards-control \
  --standards-control-arn "arn:aws:securityhub:us-west-1:123456789012:control/pci-
dss/v/3.2.1/PCI.AutoScaling.1" \
  --control-status "DISABLED" \
  --disabled-reason "Not applicable for my service"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk mengaktifkan kontrol

`update-standards-control` Contoh berikut memungkinkan PCI. AutoScaling.1 kontrol.

```
aws securityhub update-standards-control \
  --standards-control-arn "arn:aws:securityhub:us-west-1:123456789012:control/pci-
dss/v/3.2.1/PCI.AutoScaling.1" \
  --control-status "ENABLED"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menonaktifkan dan mengaktifkan kontrol individual](#) di Panduan Pengguna AWS Security Hub.

- Untuk detail API, lihat [UpdateStandardsControl](#) di Referensi AWS CLI Perintah.

## AWS Serverless Application Repository contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Serverless Application Repository.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **put-application-policy**

Contoh kode berikut menunjukkan cara menggunakan `put-application-policy`.

AWS CLI

Contoh 1: Untuk berbagi aplikasi secara publik

Berikut ini `put-application-policy` membagikan aplikasi secara publik, sehingga siapa pun dapat menemukan dan menyebarkan aplikasi Anda di AWS Serverless Application Repository.

```
aws serverlessrepo put-application-policy \  
  --application-id arn:aws:serverlessrepo:us-east-1:123456789012:applications/my-  
test-application \  
  --statements Principals='*',Actions=Deploy
```

Output:

```
{  
  "Statements": [  
    {  
      "Actions": [  
        "Deploy"  
      ],  
      "Principals": [  
        ""  
      ],  
      "StatementId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"  
    }  
  ]  
}
```

```
]
}
```

Contoh 2: Untuk berbagi aplikasi secara pribadi

Berikut ini `put-application-policy` membagikan aplikasi secara pribadi, sehingga hanya AWS akun tertentu yang dapat menemukan dan menyebarkan aplikasi Anda di AWS Serverless Application Repository.

```
aws serverlessrepo put-application-policy \
  --application-id arn:aws:serverlessrepo:us-east-1:123456789012:applications/my-
test-application \
  --statements Principals=111111111111,222222222222,Actions=Deploy
```

Output:

```
{
  "Statements": [
    {
      "Actions": [
        "Deploy"
      ],
      "Principals": [
        "111111111111",
        "222222222222"
      ],
      "StatementId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Berbagi Aplikasi Melalui Konsol](#) di Panduan Pengembang AWS Repositori Aplikasi Tanpa Server

- Untuk detail API, lihat [PutApplicationPolicy](#) di Referensi AWS CLI Perintah.

## Contoh Service Catalog menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Service Catalog.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **accept-portfolio-share**

Contoh kode berikut menunjukkan cara menggunakan `accept-portfolio-share`.

AWS CLI

Untuk menerima pembagian portofolio

`accept-portfolio-share` Contoh berikut menerima penawaran, yang dibuat oleh pengguna lain, untuk berbagi portofolio yang ditentukan.

```
aws servicecatalog accept-portfolio-share \  
  --portfolio-id port-2s6wuabcdefghijkl
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AcceptPortfolioShare](#) di Referensi AWS CLI Perintah.

### **associate-principal-with-portfolio**

Contoh kode berikut menunjukkan cara menggunakan `associate-principal-with-portfolio`.

AWS CLI

Untuk mengasosiasikan prinsipal dengan portofolio

`associate-principal-with-portfolio` Contoh berikut mengaitkan pengguna dengan portofolio yang ditentukan.

```
aws servicecatalog associate-principal-with-portfolio \  
  --portfolio-id port-2s6abcdefwdh4 \  
  --principal-arn arn:aws:iam::123456789012:user/usertest \  
  --principal-type IAM
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AssociatePrincipalWithPortfolio](#) di Referensi AWS CLI Perintah.

## **associate-product-with-portfolio**

Contoh kode berikut menunjukkan cara menggunakan `associate-product-with-portfolio`.

### AWS CLI

Untuk mengaitkan produk dengan portofolio

`associate-product-with-portfolio` Contoh berikut mengaitkan produk yang diberikan dengan portofolio yang ditentukan.

```
aws servicecatalog associate-product-with-portfolio \  
  --product-id prod-3p5abcdef3oyk \  
  --portfolio-id port-2s6abcdef5wdh4
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AssociateProductWithPortfolio](#) di Referensi AWS CLI Perintah.

## **associate-tag-option-with-resource**

Contoh kode berikut menunjukkan cara menggunakan `associate-tag-option-with-resource`.

### AWS CLI

Untuk mengasosiasikan a TagOption dengan sumber daya

`associate-tag-option-with-resource` Contoh berikut mengaitkan yang ditentukan TagOption dengan sumber daya yang ditentukan.

```
aws servicecatalog associate-tag-option-with-resource \  
  --resource-id port-2s6abcdq5wdh4 \  
  --tag-option-id tag-p3abc2pkpz5qc
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AssociateTagOptionWithResource](#) di Referensi AWS CLI Perintah.

## copy-product

Contoh kode berikut menunjukkan cara menggunakan `copy-product`.

### AWS CLI

Untuk menyalin produk

`copy-product` Contoh berikut membuat salinan dari produk yang ditentukan, menggunakan file JSON untuk meneruskan parameter.

```
aws servicecatalog copy-product --cli-input-json file://copy-product-input.json
```

Isi dari `copy-product-input.json`:

```
{  
  "SourceProductArn": "arn:aws:catalog:us-west-2:123456789012:product/prod-  
tcabcd3syn2xy",  
  "TargetProductName": "copy-of-myproduct",  
  "CopyOptions": [  
    "CopyTags"  
  ]  
}
```

Output:

```
{  
  "CopyProductToken": "copyproduct-abc5defgjdkdji"  
}
```

- Untuk detail API, lihat [CopyProduct](#) di Referensi AWS CLI Perintah.

## create-portfolio-share

Contoh kode berikut menunjukkan cara menggunakan `create-portfolio-share`.

### AWS CLI

Untuk berbagi portofolio dengan akun

`create-portfolio-share` Contoh berikut berbagi portofolio yang ditentukan dengan akun yang ditentukan.

```
aws servicecatalog create-portfolio-share \
  --portfolio-id port-2s6abcdef5wdh4 \
  --account-id 794123456789
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [CreatePortfolioShare](#) di Referensi AWS CLI Perintah.

## create-portfolio

Contoh kode berikut menunjukkan cara menggunakan `create-portfolio`.

### AWS CLI

Untuk membuat portofolio

`create-portfolio` Contoh berikut menciptakan portofolio.

```
aws servicecatalog create-portfolio \
  --provider-name my-provider \
  --display-name my-portfolio
```

Output:

```
{
  "PortfolioDetail": {
    "ProviderName": "my-provider",
    "DisplayName": "my-portfolio",
    "CreatedTime": 1571337221.555,
    "ARN": "arn:aws:catalog:us-east-2:123456789012:portfolio/
port-2s6xmplq5wdh4",
```

```
    "Id": "port-2s6xmplq5wdh4"
  }
}
```

- Untuk detail API, lihat [CreatePortfoliodi Referensi AWS CLI Perintah](#).

## create-product

Contoh kode berikut menunjukkan cara menggunakan `create-product`.

### AWS CLI

Untuk membuat produk

`create-product` Contoh berikut membuat produk, menggunakan file JSON untuk meneruskan parameter.

```
aws servicecatalog create-product \  
  --cli-input-json file://create-product-input.json
```

Isi dari `create-product-input.json`:

```
{  
  "AcceptLanguage": "en",  
  "Name": "test-product",  
  "Owner": "test-owner",  
  "Description": "test-description",  
  "Distributor": "test-distributor",  
  "SupportDescription": "test-support",  
  "SupportEmail": "test@amazon.com",  
  "SupportUrl": "https://aws.amazon.com",  
  "ProductType": "CLOUD_FORMATION_TEMPLATE",  
  "Tags": [  
    {  
      "Key": "region",  
      "Value": "us-east-1"  
    }  
  ],  
  "ProvisioningArtifactParameters": {  
    "Name": "test-version-name",  
    "Description": "test-version-description",  
    "Info": {
```



```
    "LoadTemplateFromURL": "https://s3-us-west-1.amazonaws.com/
cloudformation-templates-us-west-1/my-cfn-template.template"
  },
  "Type": "CLOUD_FORMATION_TEMPLATE"
}
}
```

## Output:

```
{
  "Tags": [
    {
      "Key": "region",
      "Value": "us-east-1"
    }
  ],
  "ProductViewDetail": {
    "CreatedTime": 1576025036.0,
    "ProductARN": "arn:aws:catalog:us-west-2:1234568542028:product/
prod-3p5abcdef3oyk",
    "Status": "CREATED",
    "ProductViewSummary": {
      "Type": "CLOUD_FORMATION_TEMPLATE",
      "Distributor": "test-distributor",
      "SupportUrl": "https://aws.amazon.com",
      "SupportEmail": "test@amazon.com",
      "Id": "prodview-abcd42wvx45um",
      "SupportDescription": "test-support",
      "ShortDescription": "test-description",
      "Owner": "test-owner",
      "Name": "test-product2",
      "HasDefaultPath": false,
      "ProductId": "prod-3p5abcdef3oyk"
    }
  },
  "ProvisioningArtifactDetail": {
    "CreatedTime": 1576025036.0,
    "Active": true,
    "Id": "pa-pq3p5lil12a34",
    "Description": "test-version-description",
    "Name": "test-version-name",
    "Type": "CLOUD_FORMATION_TEMPLATE"
  }
}
```

```
}
```

- Untuk detail API, lihat [CreateProduct](#) di Referensi AWS CLI Perintah.

## create-provisioning-artifact

Contoh kode berikut menunjukkan cara menggunakan `create-provisioning-artifact`.

### AWS CLI

Untuk membuat artefak penyediaan

`create-provisioning-artifact` Contoh berikut membuat artefak penyediaan, menggunakan file JSON untuk meneruskan parameter.

```
aws servicecatalog create-provisioning-artifact \  
  --cli-input-json file://create-provisioning-artifact-input.json
```

Isi dari `create-provisioning-artifact-input.json`:

```
{  
  "ProductId": "prod-nfi2abcdefghi",  
  "Parameters": {  
    "Name": "test-provisioning-artifact",  
    "Description": "test description",  
    "Info": {  
      "LoadTemplateFromURL": "https://s3-us-west-1.amazonaws.com/  
cloudformation-templates-us-west-1/my-cfn-template.template"  
    },  
    "Type": "CLOUD_FORMATION_TEMPLATE"  
  }  
}
```

Output:

```
{  
  "Info": {  
    "TemplateUrl": "https://s3-us-west-1.amazonaws.com/cloudformation-templates-  
us-west-1/my-cfn-template.template"  
  },  
  "Status": "CREATING",  
  "ProvisioningArtifactDetail": {
```

```
    "Id": "pa-bb4abcdefwnaio",
    "Name": "test-provisioning-artifact",
    "Description": "test description",
    "Active": true,
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "CreatedTime": 1576022545.0
  }
}
```

- Untuk detail API, lihat [CreateProvisioningArtifact](#) di Referensi AWS CLI Perintah.

## create-tag-option

Contoh kode berikut menunjukkan cara menggunakan `create-tag-option`.

### AWS CLI

Untuk membuat `TagOption`

`create-tag-option` Contoh berikut menciptakan a `TagOption`.

```
aws servicecatalog create-tag-option
  --key 1234
  --value name
```

Output:

```
{
  "TagOptionDetail": {
    "Id": "tag-iabcdn4fzjjms",
    "Value": "name",
    "Active": true,
    "Key": "1234"
  }
}
```

- Untuk detail API, lihat [CreateTagOption](#) di Referensi AWS CLI Perintah.

## delete-portfolio-share

Contoh kode berikut menunjukkan cara menggunakan `delete-portfolio-share`.

## AWS CLI

Untuk berhenti berbagi portofolio dengan akun

`delete-portfolio-share` Contoh berikut berhenti berbagi portofolio dengan akun yang ditentukan.

```
aws servicecatalog delete-portfolio-share \  
  --portfolio-id port-2s6abcdq5wdh4 \  
  --account-id 123456789012
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePortfolioShare](#) di Referensi AWS CLI Perintah.

## **delete-portfolio**

Contoh kode berikut menunjukkan cara menggunakan `delete-portfolio`.

### AWS CLI

Untuk menghapus portofolio

`delete-portfolio` Contoh berikut menghapus portofolio yang ditentukan.

```
aws servicecatalog delete-portfolio \  
  --id port-abcdlx4gox4do
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePortfolio](#) di Referensi AWS CLI Perintah.

## **delete-product**

Contoh kode berikut menunjukkan cara menggunakan `delete-product`.

### AWS CLI

Untuk menghapus produk

`delete-product` Contoh berikut menghapus produk yang ditentukan.

```
aws servicecatalog delete-product \  
  --id prod-abcdcek6yhbxi
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteProduct](#) di Referensi AWS CLI Perintah.

## **delete-provisioning-artifact**

Contoh kode berikut menunjukkan cara menggunakan `delete-provisioning-artifact`.

### AWS CLI

Untuk menghapus artefak penyediaan

`delete-provisioning-artifact` Contoh berikut menghapus artefak penyediaan yang ditentukan.

```
aws servicecatalog delete-provisioning-artifact \  
  --product-id prod-abc2uebuplcpw \  
  --provisioning-artifact-id pa-pqabddii7ouc
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteProvisioningArtifact](#) di Referensi AWS CLI Perintah.

## **delete-tag-option**

Contoh kode berikut menunjukkan cara menggunakan `delete-tag-option`.

### AWS CLI

Untuk menghapus TagOption

`delete-tag-option` Contoh berikut menghapus yang ditentukan TagOption.

```
aws servicecatalog delete-tag-option \  
  --id tag-iabcdn4fzjjms
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteTagOption](#) di Referensi AWS CLI Perintah.

## describe-copy-product-status

Contoh kode berikut menunjukkan cara menggunakan `describe-copy-product-status`.

### AWS CLI

Untuk menggambarkan status operasi produk salinan

`describe-copy-product-status` Contoh berikut menampilkan status saat ini dari operasi produk salinan asinkron yang ditentukan.

```
aws servicecatalog describe-copy-product-status \  
  --copy-product-token copyproduct-znn5tf5abcd3w
```

Output:

```
{  
  "CopyProductStatus": "SUCCEEDED",  
  "TargetProductId": "prod-os6hog7abcdt2"  
}
```

- Untuk detail API, lihat [DescribeCopyProductStatus](#) di Referensi AWS CLI Perintah.

## describe-portfolio

Contoh kode berikut menunjukkan cara menggunakan `describe-portfolio`.

### AWS CLI

Untuk menggambarkan portofolio

`describe-portfolio` Contoh berikut menampilkan rincian untuk portofolio yang ditentukan.

```
aws servicecatalog describe-portfolio \  
  --id port-2s6abcdq5wdh4
```

Output:

```
{  
  "TagOptions": [],
```

```

    "PortfolioDetail": {
      "ARN": "arn:aws:catalog:us-west-2:687558541234:portfolio/
port-2s6abcdq5wdh4",
      "Id": "port-2s6wuzyyq5wdh4",
      "CreatedTime": 1571337221.555,
      "DisplayName": "my-portfolio",
      "ProviderName": "my-provider"
    },
    "Tags": []
  }

```

- Untuk detail API, lihat [DescribePortfoliodi Referensi AWS CLI Perintah](#).

## describe-product-as-admin

Contoh kode berikut menunjukkan cara menggunakan `describe-product-as-admin`.

### AWS CLI

Untuk menggambarkan suatu produk sebagai administrator

`describe-product-as-admin` Contoh berikut menampilkan rincian untuk produk yang ditentukan menggunakan hak administrator.

```

aws servicecatalog describe-product-as-admin \
  --id prod-abcdcek6yhbxi

```

### Output:

```

{
  "TagOptions": [],
  "ProductViewDetail": {
    "ProductARN": "arn:aws:catalog:us-west-2:687558542028:product/prod-
abcdcek6yhbxi",
    "ProductViewSummary": {
      "SupportEmail": "test@amazon.com",
      "Type": "CLOUD_FORMATION_TEMPLATE",
      "Distributor": "test-distributor",
      "ShortDescription": "test-description",
      "Owner": "test-owner",
      "Id": "prodview-wi3l2j4abc6vc",
      "SupportDescription": "test-support",

```

```

        "ProductId": "prod-abcdcek6yhbxix",
        "HasDefaultPath": false,
        "Name": "test-product3",
        "SupportUrl": "https://aws.amazon.com"
    },
    "CreatedTime": 1577136715.0,
    "Status": "CREATED"
},
"ProvisioningArtifactSummaries": [
  {
    "CreatedTime": 1577136715.0,
    "Description": "test-version-description",
    "ProvisioningArtifactMetadata": {
      "SourceProvisioningArtifactId": "pa-abcdxkkiv5fcm"
    },
    "Name": "test-version-name-3",
    "Id": "pa-abcdxkkiv5fcm"
  }
],
"Tags": [
  {
    "Value": "iad",
    "Key": "region"
  }
]
}

```

- Untuk detail API, lihat [DescribeProductAsAdmin](#) di Referensi AWS CLI Perintah.

## describe-provisioned-product

Contoh kode berikut menunjukkan cara menggunakan `describe-provisioned-product`.

### AWS CLI

Untuk menggambarkan produk yang disediakan

`describe-provisioned-product` Contoh berikut menampilkan rincian untuk produk yang disediakan tertentu.

```
aws servicecatalog describe-provisioned-product \
  --id pp-dpom27bm4abcd
```



**Output:**

```
{
  "ProvisionedProductDetail": {
    "Status": "ERROR",
    "CreatedTime": 1577222793.358,
    "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/mytestppname3/pp-dpom27bm4abcd",
    "Id": "pp-dpom27bm4abcd",
    "StatusMessage": "AmazonCloudFormationException Parameters: [KeyName] must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code: ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
    "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
    "LastRecordId": "rec-tfuawdjovzxge",
    "Type": "CFN_STACK",
    "Name": "mytestppname3"
  },
  "CloudWatchDashboards": []
}
```

- Untuk detail API, lihat [DescribeProvisionedProduct](#) di Referensi AWS CLI Perintah.

**describe-provisioning-artifact**

Contoh kode berikut menunjukkan cara menggunakan `describe-provisioning-artifact`.

**AWS CLI**

Untuk menggambarkan artefak penyediaan

`describe-provisioning-artifact` Contoh berikut menampilkan rincian untuk artefak penyediaan tertentu.

```
aws servicecatalog describe-provisioning-artifact \
  --provisioning-artifact-id pa-pcz347abcdcfm \
  --product-id prod-abcdfz3syn2rg
```

**Output:**

```
{
  "Info": {
```

```
    "TemplateUrl": "https://awsdocs.s3.amazonaws.com/servicecatalog/
myexampledevelopment-environment.template"
  },
  "ProvisioningArtifactDetail": {
    "Id": "pa-pcz347abcdcfm",
    "Active": true,
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "Description": "updated description",
    "CreatedTime": 1562097906.0,
    "Name": "updated name"
  },
  "Status": "AVAILABLE"
}
```

- Untuk detail API, lihat [DescribeProvisioningArtifact](#) di Referensi AWS CLI Perintah.

## describe-tag-option

Contoh kode berikut menunjukkan cara menggunakan `describe-tag-option`.

### AWS CLI

Untuk menggambarkan TagOption

`describe-tag-option` Contoh berikut menampilkan rincian untuk yang ditentukan TagOption.

```
aws servicecatalog describe-tag-option \
  --id tag-p3tej2abcd5qc
```

Output:

```
{
  "TagOptionDetail": {
    "Active": true,
    "Id": "tag-p3tej2abcd5qc",
    "Value": "value-3",
    "Key": "1234"
  }
}
```

- Untuk detail API, lihat [DescribeTagOption](#) di Referensi AWS CLI Perintah.

## **disassociate-principal-from-portfolio**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-principal-from-portfolio`.

### AWS CLI

Untuk memisahkan kepala sekolah dari portofolio

`disassociate-principal-from-portfolio` Contoh berikut memisahkan prinsipal yang ditentukan dari portofolio.

```
aws servicecatalog disassociate-principal-from-portfolio \  
  --portfolio-id port-2s6abcdq5wdh4 \  
  --principal-arn arn:aws:iam::123456789012:group/myendusers
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DisassociatePrincipalFromPortfolio](#) di Referensi AWS CLI Perintah.

## **disassociate-product-from-portfolio**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-product-from-portfolio`.

### AWS CLI

Untuk memisahkan produk dari portofolio

`disassociate-product-from-portfolio` Contoh berikut memisahkan produk yang ditentukan dari portofolio.

```
aws servicecatalog disassociate-product-from-portfolio \  
  --product-id prod-3p5abcdmu3oyk \  
  --portfolio-id port-2s6abcdq5wdh4
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DisassociateProductFromPortfolio](#) di Referensi AWS CLI Perintah.

## disassociate-tag-option-from-resource

Contoh kode berikut menunjukkan cara menggunakan `disassociate-tag-option-from-resource`.

### AWS CLI

Untuk memisahkan a TagOption dari sumber daya

`disassociate-tag-option-from-resource` Contoh berikut memisahkan yang ditentukan TagOption dari sumber daya.

```
aws servicecatalog disassociate-tag-option-from-resource \  
  --resource-id port-2s6abcdq5wdh4 \  
  --tag-option-id tag-p3abc2pkpz5qc
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DisassociateTagOptionFromResource](#) di Referensi AWS CLI Perintah.

## list-accepted-portfolio-shares

Contoh kode berikut menunjukkan cara menggunakan `list-accepted-portfolio-shares`.

### AWS CLI

Untuk mencantumkan saham portofolio yang diterima

`list-accepted-portfolio-shares` Contoh berikut mencantumkan semua portofolio yang berbagi diterima oleh akun ini, termasuk hanya portofolio Service Catalog default.

```
aws servicecatalog list-accepted-portfolio-shares \  
  --portfolio-share-type "AWS_SERVICECATALOG"
```

Output:

```
{  
  "PortfolioDetails": [  
    {  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/port-  
d2abcd5dpkuma",
```

```

        "Description": "AWS Service Catalog Reference blueprints for often-used
AWS services such as EC2, S3, RDS, VPC and EMR.",
        "CreatedTime": 1574456190.687,
        "ProviderName": "AWS Service Catalog",
        "DisplayName": "Reference Architectures",
        "Id": "port-d2abcd5dpkuma"
    },
    {
        "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/port-
abcdefaua7zpu",
        "Description": "AWS well-architected blueprints for high reliability
applications.",
        "CreatedTime": 1574461496.092,
        "ProviderName": "AWS Service Catalog",
        "DisplayName": "High Reliability Architectures",
        "Id": "port-abcdefaua7zpu"
    }
]
}

```

- Untuk detail API, lihat [ListAcceptedPortfolioShares](#) di Referensi AWS CLI Perintah.

## list-portfolio-access

Contoh kode berikut menunjukkan cara menggunakan `list-portfolio-access`.

### AWS CLI

Untuk membuat daftar akun dengan akses ke portofolio

`list-portfolio-access` Contoh berikut mencantumkan AWS akun yang memiliki akses ke portofolio yang ditentukan.

```
aws servicecatalog list-portfolio-access \
  --portfolio-id port-2s6abcdq5wdh4
```

### Output:

```
{
  "AccountIds": [
    "123456789012"
  ]
}
```

```
}
```

- Untuk detail API, lihat [ListPortfolioAccess](#) di Referensi AWS CLI Perintah.

## list-portfolios-for-product

Contoh kode berikut menunjukkan cara menggunakan `list-portfolios-for-product`.

### AWS CLI

Untuk daftar portofolio yang terkait dengan produk

`list-portfolios-for-product` Contoh berikut mencantumkan portofolio yang terkait dengan produk yang ditentukan.

```
aws servicecatalog list-portfolios-for-product \  
  --product-id prod-abcdefz3syn2rg
```

### Output:

```
{  
  "PortfolioDetails": [  
    {  
      "CreatedTime": 1571337221.555,  
      "Id": "port-2s6abcdq5wdh4",  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/  
port-2s6abcdq5wdh4",  
      "DisplayName": "my-portfolio",  
      "ProviderName": "my-provider"  
    },  
    {  
      "CreatedTime": 1559665256.348,  
      "Id": "port-5abcd3e5st4ei",  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/  
port-5abcd3e5st4ei",  
      "DisplayName": "test",  
      "ProviderName": "provider-name"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListPortfoliosForProduct](#) di Referensi AWS CLI Perintah.

## list-portfolios

Contoh kode berikut menunjukkan cara menggunakan `list-portfolios`.

### AWS CLI

Untuk daftar portofolio

`list-portfolios` Contoh berikut mencantumkan portofolio Service Catalog di Wilayah saat ini.

```
aws servicecatalog list-portfolios
```

Output:

```
{
  "PortfolioDetails": [
    {
      "CreatedTime": 1559665256.348,
      "ARN": "arn:aws:catalog:us-east-2:123456789012:portfolio/
port-5pzcxmlst4ei",
      "DisplayName": "my-portfolio",
      "Id": "port-5pzcxmlst4ei",
      "ProviderName": "my-user"
    }
  ]
}
```

- Untuk detail API, lihat [ListPortfolios](#) di Referensi AWS CLI Perintah.

## list-principals-for-portfolio

Contoh kode berikut menunjukkan cara menggunakan `list-principals-for-portfolio`.

### AWS CLI

Untuk daftar semua prinsipal untuk portofolio

`list-principals-for-portfolio` Contoh berikut mencantumkan semua prinsipal untuk portofolio yang ditentukan.

```
aws servicecatalog list-principals-for-portfolio \
```

```
--portfolio-id port-2s6abcdq5wdh4
```

Output:

```
{
  "Principals": [
    {
      "PrincipalARN": "arn:aws:iam::123456789012:user/usertest",
      "PrincipalType": "IAM"
    }
  ]
}
```

- Untuk detail API, lihat [ListPrincipalsForPortfolio](#) di Referensi AWS CLI Perintah.

## list-provisioning-artifacts

Contoh kode berikut menunjukkan cara menggunakan `list-provisioning-artifacts`.

AWS CLI

Untuk mencantumkan semua artefak penyediaan untuk suatu produk

`list-provisioning-artifacts` Contoh berikut mencantumkan semua artefak penyediaan untuk produk yang ditentukan.

```
aws servicecatalog list-provisioning-artifacts \
  --product-id prod-nfi2abcdefgcpw
```

Output:

```
{
  "ProvisioningArtifactDetails": [
    {
      "Id": "pa-abcdef54ipm6z",
      "Description": "test-version-description",
      "Type": "CLOUD_FORMATION_TEMPLATE",
      "CreatedTime": 1576021147.0,
      "Active": true,
      "Name": "test-version-name"
    },
    {
```



```

    "Id": "pa-bb4zyxwwnaio",
    "Description": "test description",
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "CreatedTime": 1576022545.0,
    "Active": true,
    "Name": "test-provisioning-artifact-2"
  }
]
}

```

- Untuk detail API, lihat [ListProvisioningArtifacts](#) di Referensi AWS CLI Perintah.

## list-resources-for-tag-option

Contoh kode berikut menunjukkan cara menggunakan `list-resources-for-tag-option`.

### AWS CLI

Untuk membuat daftar sumber daya yang terkait dengan `TagOption`

`list-resources-for-tag-option` Contoh berikut mencantumkan sumber daya yang terkait dengan yang ditentukan `TagOption`.

```

aws servicecatalog list-resources-for-tag-option \
  --tag-option-id tag-p3tej2abcd5qc

```

### Output:

```

{
  "ResourceDetails": [
    {
      "ARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
abcdfz3syn2rg",
      "Name": "my product",
      "Description": "description",
      "CreatedTime": 1562097906.0,
      "Id": "prod-abcdfz3syn2rg"
    }
  ]
}

```

- Untuk detail API, lihat [ListResourcesForTagOption](#) di Referensi AWS CLI Perintah.

## list-tag-options

Contoh kode berikut menunjukkan cara menggunakan `list-tag-options`.

### AWS CLI

`list-tag-options` Contoh berikut mencantumkan semua nilai untuk `TagOptions`.

```
aws servicecatalog list-tag-options
```

### Output:

```
{
  "TagOptionDetails": [
    {
      "Value": "newvalue",
      "Active": true,
      "Id": "tag-iabcdn4fzjjms",
      "Key": "1234"
    },
    {
      "Value": "value1",
      "Active": true,
      "Id": "tag-e3abcdvmwvrzy",
      "Key": "key"
    }
  ]
}
```

- Untuk detail API, lihat [ListTagOptions](#) di Referensi AWS CLI Perintah.

## provision-product

Contoh kode berikut menunjukkan cara menggunakan `provision-product`.

### AWS CLI

Untuk menyediakan produk

`provision-product` Contoh berikut menyediakan produk yang ditentukan menggunakan artefak penyediaan yang ditentukan.

```
aws servicecatalog provision-product \  
  --product-id prod-abcdefz3syn2rg \  
  --provisioning-artifact-id pa-abc347pcscfm \  
  --provisioned-product-name "mytestppname3"
```

Output:

```
{  
  "RecordDetail": {  
    "RecordId": "rec-tfuawdabcdege",  
    "CreatedTime": 1577222793.362,  
    "ProvisionedProductId": "pp-abcd27bm4mldq",  
    "PathId": "lpv2-abcdg3jp6t5k6",  
    "RecordErrors": [],  
    "ProductId": "prod-abcdefz3syn2rg",  
    "UpdatedTime": 1577222793.362,  
    "RecordType": "PROVISION_PRODUCT",  
    "ProvisionedProductName": "mytestppname3",  
    "ProvisioningArtifactId": "pa-pcz347abcdcfm",  
    "RecordTags": [],  
    "Status": "CREATED",  
    "ProvisionedProductType": "CFN_STACK"  
  }  
}
```

- Untuk detail API, lihat [ProvisionProduct](#) di Referensi AWS CLI Perintah.

## reject-portfolio-share

Contoh kode berikut menunjukkan cara menggunakan `reject-portfolio-share`.

### AWS CLI

Untuk menolak pembagian portofolio

`reject-portfolio-share` Contoh berikut menolak pangsa portofolio untuk portofolio yang diberikan.

```
aws servicecatalog reject-portfolio-share \  
  --portfolio-id port-2s6wuabcdefghijk
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RejectPortfolioShared](#) di Referensi AWS CLI Perintah.

## scan-provisioned-products

Contoh kode berikut menunjukkan cara menggunakan `scan-provisioned-products`.

### AWS CLI

Untuk mencantumkan semua produk yang tersedia

`scan-provisioned-products` Contoh berikut mencantumkan produk yang disediakan yang tersedia.

```
aws servicecatalog scan-provisioned-products
```

Output:

```
{
  "ProvisionedProducts": [
    {
      "Status": "ERROR",
      "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/mytestppname3/pp-abcd27bm4mldq",
      "StatusMessage": "AmazonCloudFormationException Parameters: [KeyName] must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code: ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
      "Id": "pp-abcd27bm4mldq",
      "Type": "CFN_STACK",
      "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
      "CreatedTime": 1577222793.358,
      "Name": "mytestppname3",
      "LastRecordId": "rec-tfuawdabcdxge"
    }
  ]
}
```

- Untuk detail API, lihat [ScanProvisionedProducts](#) di Referensi AWS CLI Perintah.

## search-products-as-admin

Contoh kode berikut menunjukkan cara menggunakan `search-products-as-admin`.

## AWS CLI

Untuk mencari produk dengan hak administrator

`search-products-as-admin` Contoh berikut mencari produk dengan hak istimewa admin, menggunakan ID portofolio sebagai filter.

```
aws servicecatalog search-products-as-admin \  
  --portfolio-id port-5abcd3e5st4ei
```

Output:

```
{  
  "ProductViewDetails": [  
    {  
      "ProductViewSummary": {  
        "Name": "my product",  
        "Owner": "owner name",  
        "Type": "CLOUD_FORMATION_TEMPLATE",  
        "ProductId": "prod-abcdefz3syn2rg",  
        "HasDefaultPath": false,  
        "Id": "prodview-abcdmyuzv2dlu",  
        "ShortDescription": "description"  
      },  
      "ProductARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-  
abcdefz3syn2rg",  
      "CreatedTime": 1562097906.0,  
      "Status": "CREATED"  
    }  
  ]  
}
```

- Untuk detail API, lihat [SearchProductsAsAdmin](#) di Referensi AWS CLI Perintah.

## **search-provisioned-products**

Contoh kode berikut menunjukkan cara menggunakan `search-provisioned-products`.

## AWS CLI

Untuk mencari produk yang disediakan

search-provisioned-products Contoh berikut mencari produk yang disediakan yang cocok dengan ID produk yang ditentukan, menggunakan file JSON untuk meneruskan parameter.

```
aws servicecatalog search-provisioned-products \  
  --cli-input-json file://search-provisioned-products-input.json
```

Isi dari search-provisioned-products-input.json:

```
{  
  "Filters": {  
    "SearchQuery": [  
      "prod-tcjevz3syn2rg"  
    ]  
  }  
}
```

Output:

```
{  
  "ProvisionedProducts": [  
    {  
      "ProvisioningArtifactId": "pa-pcz347abcdcfm",  
      "Name": "mytestppname3",  
      "CreatedTime": 1577222793.358,  
      "Id": "pp-abcd27bm4mldq",  
      "Status": "ERROR",  
      "UserArn": "arn:aws:iam::123456789012:user/cliuser",  
      "StatusMessage": "AmazonCloudFormationException Parameters: [KeyName]  
must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code:  
ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",  
      "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/  
mytestppname3/pp-abcd27bm4mldq",  
      "Tags": [  
        {  
          "Value": "arn:aws:catalog:us-west-2:123456789012:product/prod-  
abcdfz3syn2rg",  
          "Key": "aws:servicecatalog:productArn"  
        },  
        {  
          "Value": "arn:aws:iam::123456789012:user/cliuser",  
          "Key": "aws:servicecatalog:provisioningPrincipalArn"  
        }  
      ],  
    }  
  ]  
}
```

```

        {
            "Value": "value-3",
            "Key": "1234"
        },
        {
            "Value": "pa-pcz347abcdcfm",
            "Key": "aws:servicecatalog:provisioningArtifactIdentifier"
        },
        {
            "Value": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-2s6abcdq5wdh4",
            "Key": "aws:servicecatalog:portfolioArn"
        },
        {
            "Value": "arn:aws:servicecatalog:us-west-2:123456789012:stack/
mytestppname3/pp-abcd27bm4mldq",
            "Key": "aws:servicecatalog:provisionedProductArn"
        }
    ],
    "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
    "UserArnSession": "arn:aws:iam::123456789012:user/cliuser",
    "Type": "CFN_STACK",
    "LastRecordId": "rec-tfuawdabcdxge",
    "ProductId": "prod-abcdfz3syn2rg"
}
],
"TotalResultsCount": 1
}

```

- Untuk detail API, lihat [SearchProvisionedProducts](#) di Referensi AWS CLI Perintah.

## update-portfolio

Contoh kode berikut menunjukkan cara menggunakan `update-portfolio`.

### AWS CLI

Untuk memperbarui portofolio

`update-portfolio` Contoh berikut memperbarui nama portofolio yang ditentukan.

```
aws servicecatalog update-portfolio \
  --id port-5abcd3e5st4ei \
```

```
--display-name "New portfolio name"
```

Output:

```
{
  "PortfolioDetail": {
    "DisplayName": "New portfolio name",
    "ProviderName": "provider",
    "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-5abcd3e5st4ei",
    "Id": "port-5abcd3e5st4ei",
    "CreatedTime": 1559665256.348
  },
  "Tags": []
}
```

- Untuk detail API, lihat [UpdatePortfolio](#) di Referensi AWS CLI Perintah.

## update-product

Contoh kode berikut menunjukkan cara menggunakan `update-product`.

### AWS CLI

Untuk memperbarui produk

`update-product` Contoh berikut memperbarui nama dan pemilik produk yang ditentukan.

```
aws servicecatalog update-product \
  --id prod-os6abc7drqlt2 \
  --name "New product name" \
  --owner "Updated product owner"
```

Output:

```
{
  "Tags": [
    {
      "Value": "iad",
      "Key": "region"
    }
  ],
}
```



```

"ProductViewDetail": {
  "ProductViewSummary": {
    "Owner": "Updated product owner",
    "ProductId": "prod-os6abc7drqlt2",
    "Distributor": "test-distributor",
    "SupportUrl": "https://aws.amazon.com",
    "Name": "New product name",
    "ShortDescription": "test-description",
    "HasDefaultPath": false,
    "Id": "prodview-6abcdgrfhvidy",
    "SupportDescription": "test-support",
    "SupportEmail": "test@amazon.com",
    "Type": "CLOUD_FORMATION_TEMPLATE"
  },
  "Status": "CREATED",
  "ProductARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
os6abc7drqlt2",
  "CreatedTime": 1577136255.0
}
}

```

- Untuk detail API, lihat [UpdateProduct](#) di Referensi AWS CLI Perintah.

## update-provisioning-artifact

Contoh kode berikut menunjukkan cara menggunakan `update-provisioning-artifact`.

### AWS CLI

Untuk memperbarui artefak penyediaan

`update-provisioning-artifact` Contoh berikut memperbarui nama dan deskripsi artefak penyediaan yang ditentukan, menggunakan file JSON untuk meneruskan parameter.

```

aws servicecatalog update-provisioning-artifact \
  --cli-input-json file://update-provisioning-artifact-input.json

```

Isi dari `update-provisioning-artifact-input.json`:

```

{
  "ProductId": "prod-abcdefz3syn2rg",
  "ProvisioningArtifactId": "pa-pcz347abcdcfm",

```

```
"Name": "updated name",
"Description": "updated description"
}
```

### Output:

```
{
  "Info": {
    "TemplateUrl": "https://awsdocs.s3.amazonaws.com/servicecatalog/
myexampledevelopment-environment.template"
  },
  "Status": "AVAILABLE",
  "ProvisioningArtifactDetail": {
    "Active": true,
    "Description": "updated description",
    "Id": "pa-pcz347abcdcfm",
    "Name": "updated name",
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "CreatedTime": 1562097906.0
  }
}
```

- Untuk detail API, lihat [UpdateProvisioningArtifact](#) di Referensi AWS CLI Perintah.

## update-tag-option

Contoh kode berikut menunjukkan cara menggunakan `update-tag-option`.

### AWS CLI

Untuk memperbarui `TagOption`

`update-tag-option` Contoh berikut memperbarui nilai `TagOption`, menggunakan file JSON yang ditentukan.

```
aws servicecatalog update-tag-option --cli-input-json file://update-tag-option-
input.json
```

Isi dari `update-tag-option-input.json`:

```
{
  "Id": "tag-iabcdn4fzjjms",
```

```
"Value": "newvalue",  
"Active": true  
}
```

Output:

```
{  
  "TagOptionDetail": {  
    "Value": "newvalue",  
    "Key": "1234",  
    "Active": true,  
    "Id": "tag-iabcdn4fzjjms"  
  }  
}
```

- Untuk detail API, lihat [UpdateTagOption](#) di Referensi AWS CLI Perintah.

## Contoh Service Quotas menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan Service Quotas AWS Command Line Interface with.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **get-aws-default-service-quota**

Contoh kode berikut menunjukkan cara menggunakan `get-aws-default-service-quota`.

## AWS CLI

Untuk mendeskripsikan kuota layanan default

`get-aws-default-service-quota` Contoh berikut menampilkan rincian untuk kuota yang ditentukan.

```
aws service-quotas get-aws-default-service-quota \  
  --service-code ec2 \  
  --quota-code L-1216C47A
```

Output:

```
{  
  "Quota": {  
    "ServiceCode": "ec2",  
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",  
    "QuotaArn": "arn:aws:servicequotas:us-east-2::ec2/L-1216C47A",  
    "QuotaCode": "L-1216C47A",  
    "QuotaName": "Running On-Demand Standard (A, C, D, H, I, M, R, T, Z)  
instances",  
    "Value": 5.0,  
    "Unit": "None",  
    "Adjustable": true,  
    "GlobalQuota": false,  
    "UsageMetric": {  
      "MetricNamespace": "AWS/Usage",  
      "MetricName": "ResourceCount",  
      "MetricDimensions": {  
        "Class": "Standard/OnDemand",  
        "Resource": "vCPU",  
        "Service": "EC2",  
        "Type": "Resource"  
      },  
      "MetricStatisticRecommendation": "Maximum"  
    }  
  }  
}
```

- Untuk detail API, lihat [GetAwsDefaultServiceQuota](#) di Referensi AWS CLI Perintah.

## get-requested-service-quota-change

Contoh kode berikut menunjukkan cara menggunakan `get-requested-service-quota-change`.

### AWS CLI

Untuk menjelaskan permintaan peningkatan kuota layanan

`get-requested-service-quota-change` Contoh berikut menjelaskan permintaan peningkatan kuota yang ditentukan.

```
aws service-quotas get-requested-service-quota-change \  
--request-id d187537d15254312a9609aa51bbf7624u7W49tP0
```

### Output:

```
{  
  "RequestedQuota": {  
    "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",  
    "CaseId": "6780195351",  
    "ServiceCode": "ec2",  
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",  
    "QuotaCode": "L-20F13EBD",  
    "QuotaName": "Running Dedicated c5n Hosts",  
    "DesiredValue": 2.0,  
    "Status": "CASE_OPENED",  
    "Created": 1580446904.067,  
    "LastUpdated": 1580446953.265,  
    "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":  
\\\"arn:aws:iam::123456789012:root\\\"}\",  
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-20F13EBD",  
    "GlobalQuota": false,  
    "Unit": "None"  
  }  
}
```

- Untuk detail API, lihat [GetRequestedServiceQuotaChange](#) di Referensi AWS CLI Perintah.

## get-service-quota

Contoh kode berikut menunjukkan cara menggunakan `get-service-quota`.

## AWS CLI

Untuk mendeskripsikan kuota layanan

`get-service-quota` Contoh berikut menampilkan rincian tentang kuota yang ditentukan.

```
aws service-quotas get-service-quota \  
  --service-code ec2 \  
  --quota-code L-1216C47A
```

Output:

```
{  
  "Quota": {  
    "ServiceCode": "ec2",  
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",  
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-1216C47A",  
    "QuotaCode": "L-1216C47A",  
    "QuotaName": "Running On-Demand Standard (A, C, D, H, I, M, R, T, Z)  
instances",  
    "Value": 1920.0,  
    "Unit": "None",  
    "Adjustable": true,  
    "GlobalQuota": false,  
    "UsageMetric": {  
      "MetricNamespace": "AWS/Usage",  
      "MetricName": "ResourceCount",  
      "MetricDimensions": {  
        "Class": "Standard/OnDemand",  
        "Resource": "vCPU",  
        "Service": "EC2",  
        "Type": "Resource"  
      },  
      "MetricStatisticRecommendation": "Maximum"  
    }  
  }  
}
```

- Untuk detail API, lihat [GetServiceQuota](#) di Referensi AWS CLI Perintah.

## list-aws-default-service-quotas

Contoh kode berikut menunjukkan cara menggunakan `list-aws-default-service-quotas`.

### AWS CLI

Untuk membuat daftar kuota default untuk layanan

`list-aws-default-service-quotas` Contoh berikut mencantumkan nilai default untuk kuota untuk layanan yang ditentukan.

```
aws service-quotas list-aws-default-service-quotas \
  --service-code xray
```

Output:

```
{
  "Quotas": [
    {
      "ServiceCode": "xray",
      "ServiceName": "AWS X-Ray",
      "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-C6B6F05D",
      "QuotaCode": "L-C6B6F05D",
      "QuotaName": "Indexed annotations per trace",
      "Value": 50.0,
      "Unit": "None",
      "Adjustable": false,
      "GlobalQuota": false
    },
    {
      "ServiceCode": "xray",
      "ServiceName": "AWS X-Ray",
      "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-D781C0FD",
      "QuotaCode": "L-D781C0FD",
      "QuotaName": "Segment document size",
      "Value": 64.0,
      "Unit": "Kilobytes",
      "Adjustable": false,
      "GlobalQuota": false
    },
    {
      "ServiceCode": "xray",
      "ServiceName": "AWS X-Ray",
```

```

    "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-998BFF16",
    "QuotaCode": "L-998BFF16",
    "QuotaName": "Trace and service graph retention in days",
    "Value": 30.0,
    "Unit": "None",
    "Adjustable": false,
    "GlobalQuota": false
  }
]
}

```

- Untuk detail API, lihat [ListAwsDefaultServiceQuotas](#) di Referensi AWS CLI Perintah.

## list-requested-service-quota-change-history-by-quota

Contoh kode berikut menunjukkan cara menggunakan `list-requested-service-quota-change-history-by-quota`.

### AWS CLI

Untuk membuat daftar permintaan peningkatan kuota

`list-requested-service-quota-change-history-by-quota` Contoh berikut mencantumkan permintaan peningkatan kuota untuk kuota yang ditentukan.

```

aws service-quotas list-requested-service-quota-change-history-by-quota \
  --service-code ec2 \
  --quota-code L-20F13EBD

```

Output:

```

{
  "RequestedQuotas": [
    {
      "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",
      "CaseId": "6780195351",
      "ServiceCode": "ec2",
      "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
      "QuotaCode": "L-20F13EBD",
      "QuotaName": "Running Dedicated c5n Hosts",
      "DesiredValue": 2.0,
      "Status": "CASE_OPENED",
    }
  ]
}

```



```

    "Created": 1580446904.067,
    "LastUpdated": 1580446953.265,
    "Requester": "{\"accountId\":\"123456789012\", \"callerArn\":
\\\"arn:aws:iam::123456789012:root\\\"}\",
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/
L-20F13EBD",
    "GlobalQuota": false,
    "Unit": "None"
  }
]
}

```

- Untuk detail API, lihat [ListRequestedServiceQuotaChangeHistoryByQuota](#) di Referensi AWS CLI Perintah.

## list-requested-service-quota-change-history

Contoh kode berikut menunjukkan cara menggunakan `list-requested-service-quota-change-history`.

### AWS CLI

Untuk membuat daftar permintaan peningkatan kuota

`list-requested-service-quota-change-history` Contoh berikut mencantumkan permintaan peningkatan kuota untuk layanan yang ditentukan.

```
aws service-quotas list-requested-service-quota-change-history \
--service-code ec2
```

### Output:

```

{
  "RequestedQuotas": [
    {
      "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",
      "CaseId": "6780195351",
      "ServiceCode": "ec2",
      "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
      "QuotaCode": "L-20F13EBD",
      "QuotaName": "Running Dedicated c5n Hosts",
      "DesiredValue": 2.0,

```

```

        "Status": "CASE_OPENED",
        "Created": 1580446904.067,
        "LastUpdated": 1580446953.265,
        "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":
\\\"arn:aws:iam::123456789012:root\\\"}\",
        "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/
L-20F13EBD",
        "GlobalQuota": false,
        "Unit": "None"
    }
]
}

```

- Untuk detail API, lihat [ListRequestedServiceQuotaChangeHistory](#) di Referensi AWS CLI Perintah.

## list-service-quotas

Contoh kode berikut menunjukkan cara menggunakan `list-service-quotas`.

### AWS CLI

Untuk membuat daftar kuota untuk layanan

`list-service-quotas` Contoh berikut menampilkan rincian tentang kuota untuk AWS CloudFormation.

```
aws service-quotas list-service-quotas \
--service-code cloudformation
```

Output:

```

{
  "Quotas": [
    {
      "ServiceCode": "cloudformation",
      "ServiceName": "AWS CloudFormation",
      "QuotaArn": "arn:aws:servicequotas:us-
east-2:123456789012:cloudformation/L-87D14FB7",
      "QuotaCode": "L-87D14FB7",
      "QuotaName": "Output count in CloudFormation template",
      "Value": 60.0,

```

```
        "Unit": "None",
        "Adjustable": false,
        "GlobalQuota": false
    },
    {
        "ServiceCode": "cloudformation",
        "ServiceName": "AWS CloudFormation",
        "QuotaArn": "arn:aws:servicequotas:us-
east-2:123456789012:cloudformation/L-0485CB21",
        "QuotaCode": "L-0485CB21",
        "QuotaName": "Stack count",
        "Value": 200.0,
        "Unit": "None",
        "Adjustable": true,
        "GlobalQuota": false
    }
]
}
```

- Untuk detail API, lihat [ListServiceQuotas](#) di Referensi AWS CLI Perintah.

## list-services

Contoh kode berikut menunjukkan cara menggunakan `list-services`.

### AWS CLI

Untuk daftar layanan yang tersedia

Perintah berikut mencantumkan layanan yang tersedia di Service Quotas.

```
aws service-quotas list-services
```

Output:

```
{
  "Services": [
    {
      "ServiceCode": "AWSCloudMap",
      "ServiceName": "AWS Cloud Map"
    },
    {
```

```
    "ServiceCode": "access-analyzer",
    "ServiceName": "Access Analyzer"
  },
  {
    "ServiceCode": "acm",
    "ServiceName": "AWS Certificate Manager (ACM)"
  },
  ...truncated...
  {
    "ServiceCode": "xray",
    "ServiceName": "AWS X-Ray"
  }
]
```

Anda dapat menambahkan `--query` parameter untuk memfilter tampilan ke informasi yang Anda minati. Contoh berikut hanya menampilkan kode layanan.

```
aws service-quotas list-services \
  --query Services[*].ServiceCode
```

Output:

```
[
  "AWSCloudMap",
  "access-analyzer",
  "acm",
  "acm-pca",
  "amplify",
  "apigateway",
  "application-autoscaling",
  ...truncated...
  "xray"
]
```

- Untuk detail API, lihat [ListServices](#) di Referensi AWS CLI Perintah.

## **request-service-quota-increase**

Contoh kode berikut menunjukkan cara menggunakan `request-service-quota-increase`.

## AWS CLI

Meminta peningkatan kuota layanan

`request-service-quota-increase` Contoh berikut meminta peningkatan kuota layanan yang ditentukan.

```
aws service-quotas request-service-quota-increase \  
  --service-code ec2 \  
  --quota-code L-20F13EBD \  
  --desired-value 2
```

Output:

```
{  
  "RequestedQuota": {  
    "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",  
    "ServiceCode": "ec2",  
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",  
    "QuotaCode": "L-20F13EBD",  
    "QuotaName": "Running Dedicated c5n Hosts",  
    "DesiredValue": 2.0,  
    "Status": "PENDING",  
    "Created": 1580446904.067,  
    "Requester": "{\"accountId\":\"123456789012\", \"callerArn\":  
  \":\"arn:aws:iam::123456789012:root\"}",  
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-20F13EBD",  
    "GlobalQuota": false,  
    "Unit": "None"  
  }  
}
```

- Untuk detail API, lihat [RequestServiceQuotaIncreased](#) di Referensi AWS CLI Perintah.

## Amazon SES contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon SES.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **delete-identity**

Contoh kode berikut menunjukkan cara menggunakan `delete-identity`.

AWS CLI

Untuk menghapus identitas

Contoh berikut menggunakan `delete-identity` perintah untuk menghapus identitas dari daftar identitas yang diverifikasi dengan Amazon SES:

```
aws ses delete-identity --identity user@example.com
```

Untuk informasi selengkapnya tentang identitas terverifikasi, lihat [Memverifikasi Alamat Email dan Domain di Amazon SES](#) di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [DeleteIdentity](#) di Referensi AWS CLI Perintah.

### **get-identity-dkim-attributes**

Contoh kode berikut menunjukkan cara menggunakan `get-identity-dkim-attributes`.

AWS CLI

Untuk mendapatkan atribut Amazon SES Easy DKIM untuk daftar identitas

Contoh berikut menggunakan `get-identity-dkim-attributes` perintah untuk mengambil atribut Amazon SES Easy DKIM untuk daftar identitas:

```
aws ses get-identity-dkim-attributes --identities "example.com" "user@example.com"
```

**Output:**

```
{
  "DkimAttributes": {
    "example.com": {
      "DkimTokens": [
        "EXAMPLEjcs5xoyqytjsotsijas7236gr",
        "EXAMPLEjr76cvoc6mysspnioorxsn6ep",
        "EXAMPLEkbnkqkhlm2lyz77ppkulerm4k"
      ],
      "DkimEnabled": true,
      "DkimVerificationStatus": "Success"
    },
    "user@example.com": {
      "DkimEnabled": false,
      "DkimVerificationStatus": "NotStarted"
    }
  }
}
```

Jika Anda memanggil perintah ini dengan identitas yang belum pernah Anda kirimkan untuk verifikasi, identitas itu tidak akan muncul di output.

Untuk informasi selengkapnya tentang Easy DKIM, lihat Easy DKIM di Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [GetIdentityDkimAttributes](#) di Referensi AWS CLI Perintah.

**get-identity-notification-attributes**

Contoh kode berikut menunjukkan cara menggunakan `get-identity-notification-attributes`.

**AWS CLI**

Untuk mendapatkan atribut notifikasi Amazon SES untuk daftar identitas

Contoh berikut menggunakan `get-identity-notification-attributes` perintah untuk mengambil atribut notifikasi Amazon SES untuk daftar identitas:

```
aws ses get-identity-notification-attributes --identities "user1@example.com"
"user2@example.com"
```

## Output:

```
{
  "NotificationAttributes": {
    "user1@example.com": {
      "ForwardingEnabled": false,
      "ComplaintTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic",
      "BounceTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic",
      "DeliveryTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic"
    },
    "user2@example.com": {
      "ForwardingEnabled": true
    }
  }
}
```

Perintah ini mengembalikan status penerusan umpan balik email dan, jika berlaku, Nama Sumber Daya Amazon (ARN) dari topik Amazon SNS yang terpental, keluhan, dan pemberitahuan pengiriman dikirim.

Jika Anda memanggil perintah ini dengan identitas yang belum pernah Anda kirimkan untuk verifikasi, identitas itu tidak akan muncul di output.

Untuk informasi selengkapnya tentang notifikasi, lihat Menggunakan Pemberitahuan Dengan Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [GetIdentityNotificationAttributes](#) di Referensi AWS CLI Perintah.

## get-identity-verification-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-identity-verification-attributes`.

### AWS CLI

Untuk mendapatkan status verifikasi Amazon SES untuk daftar identitas

Contoh berikut menggunakan `get-identity-verification-attributes` perintah untuk mengambil status verifikasi Amazon SES untuk daftar identitas:

```
aws ses get-identity-verification-attributes --identities "user1@example.com"
"user2@example.com"
```



**Output:**

```
{
  "VerificationAttributes": {
    "user1@example.com": {
      "VerificationStatus": "Success"
    },
    "user2@example.com": {
      "VerificationStatus": "Pending"
    }
  }
}
```

Jika Anda memanggil perintah ini dengan identitas yang belum pernah Anda kirimkan untuk verifikasi, identitas itu tidak akan muncul di output.

Untuk informasi selengkapnya tentang identitas terverifikasi, lihat Memverifikasi Alamat Email dan Domain di Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [GetIdentityVerificationAttributes](#) di Referensi AWS CLI Perintah.

**get-send-quota**

Contoh kode berikut menunjukkan cara menggunakan `get-send-quota`.

**AWS CLI**

Untuk mendapatkan batas pengiriman Amazon SES Anda

Contoh berikut menggunakan `get-send-quota` perintah untuk mengembalikan batas pengiriman Amazon SES Anda:

```
aws ses get-send-quota
```

**Output:**

```
{
  "Max24HourSend": 200.0,
  "SentLast24Hours": 1.0,
  "MaxSendRate": 1.0
}
```

Max24 HourSend adalah kuota pengiriman Anda, yaitu jumlah email maksimum yang dapat Anda kirim dalam jangka waktu 24 jam. Kuota pengiriman mencerminkan periode waktu bergulir. Setiap kali Anda mencoba mengirim email, Amazon SES memeriksa berapa banyak email yang Anda kirim dalam 24 jam sebelumnya. Selama jumlah email yang Anda kirim kurang dari kuota Anda, permintaan kirim Anda akan diterima dan email Anda akan terkirim.

SentLast24Hours adalah jumlah email yang telah Anda kirim dalam 24 jam sebelumnya.

MaxSendRate adalah jumlah maksimum email yang dapat Anda kirim per detik.

Perhatikan bahwa batas pengiriman didasarkan pada penerima, bukan pada pesan. Misalnya, email yang memiliki 10 penerima dihitung sebagai 10 terhadap kuota pengiriman Anda.

Untuk informasi selengkapnya, lihat Mengelola Batas Pengiriman Amazon SES Anda di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [GetSendQuota](#) di Referensi AWS CLI Perintah.

## get-send-statistics

Contoh kode berikut menunjukkan cara menggunakan `get-send-statistics`.

### AWS CLI

Untuk mendapatkan statistik pengiriman Amazon SES Anda

Contoh berikut menggunakan `get-send-statistics` perintah untuk mengembalikan statistik pengiriman Amazon SES Anda

```
aws ses get-send-statistics
```

Output:

```
{
  "SendDataPoints": [
    {
      "Complaints": 0,
      "Timestamp": "2013-06-12T19:32:00Z",
      "DeliveryAttempts": 2,
      "Bounces": 0,
      "Rejects": 0
    },
  ],
}
```

```
{
  "Complaints": 0,
  "Timestamp": "2013-06-12T00:47:00Z",
  "DeliveryAttempts": 1,
  "Bounces": 0,
  "Rejects": 0
}
]
```

Hasilnya adalah daftar titik data, yang mewakili dua minggu terakhir aktivitas pengiriman. Setiap titik data dalam daftar berisi statistik untuk interval 15 menit.

Dalam contoh ini, hanya ada dua titik data karena satu-satunya email yang dikirim pengguna dalam dua minggu terakhir jatuh dalam dua interval 15 menit.

Untuk informasi selengkapnya, lihat [Memantau Statistik Penggunaan Amazon SES Anda di Panduan Pengembang Layanan Email Sederhana Amazon](#).

- Untuk detail API, lihat [GetSendStatistics](#) di Referensi AWS CLI Perintah.

## list-identities

Contoh kode berikut menunjukkan cara menggunakan `list-identities`.

### AWS CLI

Untuk mencantumkan semua identitas (alamat email dan domain) untuk akun tertentu AWS

Contoh berikut menggunakan `list-identities` perintah untuk mencantumkan semua identitas yang telah dikirimkan untuk verifikasi dengan Amazon SES:

```
aws ses list-identities
```

Output:

```
{
  "Identities": [
    "user@example.com",
    "example.com"
  ]
}
```

Daftar yang dikembalikan berisi semua identitas terlepas dari status verifikasi (verifikasi, verifikasi tertunda, kegagalan, dll.).

Dalam contoh ini, alamat email dan domain dikembalikan karena kami tidak menentukan parameter tipe identitas.

Untuk informasi selengkapnya tentang verifikasi, lihat Memverifikasi Alamat Email dan Domain di Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [ListIdentities](#) di Referensi AWS CLI Perintah.

## send-email

Contoh kode berikut menunjukkan cara menggunakan `send-email`.

### AWS CLI

Untuk mengirim email yang diformat menggunakan Amazon SES

Contoh berikut menggunakan `send-email` perintah untuk mengirim email yang diformat:

```
aws ses send-email --from sender@example.com --destination file://destination.json
--message file://message.json
```

Output:

```
{
  "MessageId": "EXAMPLEf3a5efcd1-51adec81-d2a4-4e3f-9fe2-5d85c1b23783-000000"
}
```

Tujuan dan pesannya adalah struktur data JSON yang disimpan dalam `file.json` di direktori saat ini. File-file ini adalah sebagai berikut:

`destination.json`:

```
{
  "ToAddresses": ["recipient1@example.com", "recipient2@example.com"],
  "CcAddresses": ["recipient3@example.com"],
  "BccAddresses": []
}
```

`message.json`:

```
{
  "Subject": {
    "Data": "Test email sent using the AWS CLI",
    "Charset": "UTF-8"
  },
  "Body": {
    "Text": {
      "Data": "This is the message body in text format.",
      "Charset": "UTF-8"
    },
    "Html": {
      "Data": "This message body contains HTML formatting. It can, for example,
contain links like this one: <a class=\"uLink\" href=\"http://docs.aws.amazon.com/
ses/latest/DeveloperGuide\" target=\"_blank\">Amazon SES Developer Guide</a>.",
      "Charset": "UTF-8"
    }
  }
}
```

Ganti alamat email pengirim dan penerima dengan yang ingin Anda gunakan. Perhatikan bahwa alamat email pengirim harus diverifikasi dengan Amazon SES. Sampai Anda diberikan akses produksi ke Amazon SES, Anda juga harus memverifikasi alamat email setiap penerima kecuali penerima adalah simulator kotak surat Amazon SES. Untuk informasi selengkapnya tentang verifikasi, lihat Memverifikasi Alamat Email dan Domain di Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

ID Pesan dalam output menunjukkan bahwa panggilan ke kirim email berhasil.

Jika Anda tidak menerima email, centang kotak Sampah Anda.

Untuk informasi selengkapnya tentang mengirim email yang diformat, lihat Mengirim Email Berformat Menggunakan Amazon SES API di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [SendEmail](#) di Referensi AWS CLI Perintah.

## **send-raw-email**

Contoh kode berikut menunjukkan cara menggunakan `send-raw-email`.

## AWS CLI

Untuk mengirim email mentah menggunakan Amazon SES

Contoh berikut menggunakan `send-raw-email` perintah untuk mengirim email dengan lampiran TXT:

```
aws ses send-raw-email --raw-message file://message.json
```

Output:

```
{
  "MessageId": "EXAMPLEf3f73d99b-c63fb06f-d263-41f8-a0fb-d0dc67d56c07-000000"
}
```

Pesan mentah adalah struktur data JSON yang disimpan dalam file bernama `message.json` di direktori saat ini. Ini berisi yang berikut:

```
{
  "Data": "From: sender@example.com\nTo: recipient@example.com\nSubject: Test email sent using the AWS CLI (contains an attachment)\nMIME-Version: 1.0\nContent-type: Multipart/Mixed; boundary=\"NextPart\"\n\n--NextPart\nContent-Type: text/plain\n\nThis is the message body.\n\n--NextPart\nContent-Type: text/plain;\nContent-Disposition: attachment; filename=\"attachment.txt\"\n\nThis is the text in the attachment.\n\n--NextPart--"
}
```

Seperti yang Anda lihat, “Data” adalah satu string panjang yang berisi seluruh konten email mentah dalam format MIME, termasuk lampiran yang disebut `attachment.txt`.

Ganti `sender@example.com` dan `recipient@example.com` dengan alamat yang ingin Anda gunakan. Perhatikan bahwa alamat email pengirim harus diverifikasi dengan Amazon SES. Sampai Anda diberikan akses produksi ke Amazon SES, Anda juga harus memverifikasi alamat email penerima kecuali penerima adalah simulator kotak surat Amazon SES. Untuk informasi selengkapnya tentang verifikasi, lihat [Memverifikasi Alamat Email dan Domain di Amazon SES](#) di [Panduan Pengembang Layanan Email Sederhana Amazon](#).

ID Pesan dalam output menunjukkan bahwa panggilan ke `send-raw-email` berhasil.

Jika Anda tidak menerima email, centang kotak Sampah Anda.

Untuk informasi selengkapnya tentang mengirim email mentah, lihat [Mengirim Email Mentah Menggunakan Amazon SES API](#) di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [SendRawEmail](#) di Referensi AWS CLI Perintah.

## **set-identity-dkim-enabled**

Contoh kode berikut menunjukkan cara menggunakan `set-identity-dkim-enabled`.

### AWS CLI

Untuk mengaktifkan atau menonaktifkan Easy DKIM untuk identitas terverifikasi Amazon SES

Contoh berikut menggunakan `set-identity-dkim-enabled` perintah untuk menonaktifkan DKIM untuk alamat email terverifikasi:

```
aws ses set-identity-dkim-enabled --identity user@example.com --no-dkim-enabled
```

Untuk informasi selengkapnya tentang Easy DKIM, lihat [Easy DKIM di Amazon SES](#) di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [SetIdentityDkimEnabled](#) di Referensi AWS CLI Perintah.

## **set-identity-feedback-forwarding-enabled**

Contoh kode berikut menunjukkan cara menggunakan `set-identity-feedback-forwarding-enabled`.

### AWS CLI

Untuk mengaktifkan atau menonaktifkan penerusan umpan balik email pentalan dan keluhan untuk identitas terverifikasi Amazon SES

Contoh berikut menggunakan `set-identity-feedback-forwarding-enabled` perintah untuk mengaktifkan alamat email terverifikasi untuk menerima pemberitahuan bouncing dan keluhan melalui email:

```
aws ses set-identity-feedback-forwarding-enabled --identity user@example.com --forwarding-enabled
```

Anda diminta untuk menerima pemberitahuan pentalan dan keluhan melalui Amazon SNS atau penerusan umpan balik email, sehingga Anda hanya dapat menonaktifkan penerusan umpan balik email jika Anda memilih topik Amazon SNS untuk pemberitahuan bouncing dan keluhan.

Untuk informasi selengkapnya tentang notifikasi, lihat Menggunakan Pemberitahuan Dengan Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [SetIdentityFeedbackForwardingEnabled](#) di Referensi AWS CLI Perintah.

## **set-identity-notification-topic**

Contoh kode berikut menunjukkan cara menggunakan `set-identity-notification-topic`.

### AWS CLI

Untuk mengatur topik Amazon SNS yang Amazon SES akan mempublikasikan pemberitahuan pentalan, keluhan, dan/atau pengiriman untuk identitas terverifikasi

Contoh berikut menggunakan `set-identity-notification-topic` perintah untuk menentukan topik Amazon SNS yang alamat email terverifikasi akan menerima pemberitahuan bouncing:

```
aws ses set-identity-notification-topic --identity user@example.com --notification-type Bounce --sns-topic arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic
```

Untuk informasi selengkapnya tentang notifikasi, lihat Menggunakan Pemberitahuan Dengan Amazon SES di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [SetIdentityNotificationTopic](#) di Referensi AWS CLI Perintah.

## **verify-domain-dkim**

Contoh kode berikut menunjukkan cara menggunakan `verify-domain-dkim`.

### AWS CLI

Untuk menghasilkan token DKIM domain terverifikasi untuk penandatanganan DKIM dengan Amazon SES

Contoh berikut menggunakan `verify-domain-dkim` perintah untuk menghasilkan token DKIM untuk domain yang telah diverifikasi dengan Amazon SES:



```
aws ses verify-domain-dkim --domain example.com
```

Output:

```
{
  "DkimTokens": [
    "EXAMPLEq76owjnks3lnluwg65scbemvw",
    "EXAMPLEi3dnsj67hstzaj673klariwx2",
    "EXAMPLEwfbtcukvimehexktmdtaz6naj"
  ]
}
```

Untuk menyiapkan DKIM, Anda harus menggunakan token DKIM yang dikembalikan untuk memperbarui pengaturan DNS domain Anda dengan catatan CNAME yang mengarah ke kunci publik DKIM yang dihosting oleh Amazon SES. Untuk informasi selengkapnya, lihat [DKIM Mudah di Amazon SES](#) di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [VerifyDomainDkim](#) di Referensi AWS CLI Perintah.

## verify-domain-identity

Contoh kode berikut menunjukkan cara menggunakan `verify-domain-identity`.

### AWS CLI

Untuk memverifikasi domain dengan Amazon SES

Contoh berikut menggunakan `verify-domain-identity` perintah untuk memverifikasi domain:

```
aws ses verify-domain-identity --domain example.com
```

Output:

```
{
  "VerificationToken": "eoEmxw+YaYhb3h3iVJHuXMJXqeu1q1/wmvjuEXAMPLE"
}
```

Untuk menyelesaikan verifikasi domain, Anda harus menambahkan catatan TXT dengan token verifikasi yang dikembalikan ke pengaturan DNS domain Anda. Untuk informasi selengkapnya, lihat [Memverifikasi Domain di Amazon SES](#) di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [VerifyDomainIdentity](#) di Referensi AWS CLI Perintah.

## verify-email-identity

Contoh kode berikut menunjukkan cara menggunakan `verify-email-identity`.

### AWS CLI

Untuk memverifikasi alamat email dengan Amazon SES

Contoh berikut menggunakan `verify-email-identity` perintah untuk memverifikasi alamat email:

```
aws ses verify-email-identity --email-address user@example.com
```

Sebelum Anda dapat mengirim email menggunakan Amazon SES, Anda harus memverifikasi alamat atau domain tempat Anda mengirim email untuk membuktikan bahwa Anda memilikinya. Jika Anda belum memiliki akses produksi, Anda juga perlu memverifikasi alamat email apa pun yang Anda kirim email kecuali alamat email yang disediakan oleh simulator kotak surat Amazon SES.

Setelah `verify-email-identity` dipanggil, alamat email akan menerima email verifikasi. Pengguna harus mengklik tautan di email untuk menyelesaikan proses verifikasi.

Untuk informasi selengkapnya, lihat [Memverifikasi Alamat Email di Amazon SES](#) di Panduan Pengembang Layanan Email Sederhana Amazon.

- Untuk detail API, lihat [VerifyEmailIdentity](#) di Referensi AWS CLI Perintah.

## Contoh Shield menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Shield.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## **associate-drt-log-bucket**

Contoh kode berikut menunjukkan cara menggunakan `associate-drt-log-bucket`.

AWS CLI

Untuk mengotorisasi DRT untuk mengakses bucket Amazon S3

`associate-drt-log-bucket` Contoh berikut membuat asosiasi antara DRT dan bucket S3 yang ditentukan. Ini memungkinkan DRT untuk mengakses bucket atas nama akun. :

```
aws shield associate-drt-log-bucket \  
  --log-bucket flow-logs-for-website-lb
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengotorisasi Tim Respons DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [AssociateDrtLogBucket](#) di Referensi AWS CLI Perintah.

## **associate-drt-role**

Contoh kode berikut menunjukkan cara menggunakan `associate-drt-role`.

AWS CLI

Untuk mengotorisasi DRT untuk mengurangi potensi serangan atas nama Anda

`associate-drt-role` Contoh berikut menciptakan hubungan antara DRT dan peran yang ditentukan. DRT dapat menggunakan peran untuk mengakses dan mengelola akun.

```
aws shield associate-drt-role \  
  --role arn:aws:iam::123456789012:role/MyRole
```

```
--role-arn arn:aws:iam::123456789012:role/service-role/DrtRole
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengotorisasi Tim Respons DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [AssociateDrtRole](#) di Referensi AWS CLI Perintah.

## create-protection

Contoh kode berikut menunjukkan cara menggunakan `create-protection`.

### AWS CLI

Untuk mengaktifkan perlindungan AWS Shield Advanced untuk satu AWS sumber daya

`create-protection` Contoh berikut memungkinkan perlindungan Shield Advanced untuk AWS CloudFront distribusi yang ditentukan.

```
aws shield create-protection \  
  --name "Protection for CloudFront distribution" \  
  --resource-arn arn:aws:cloudfront::123456789012:distribution/E198WC25FX0WY8
```

Output:

```
{  
  "ProtectionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Untuk informasi selengkapnya, lihat [Menentukan Sumber Daya Anda untuk Dilindungi](#) di Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [CreateProtection](#) di Referensi AWS CLI Perintah.

## create-subscription

Contoh kode berikut menunjukkan cara menggunakan `create-subscription`.

### AWS CLI

Untuk mengaktifkan perlindungan AWS Shield Advanced untuk akun

`create-subscription` Contoh berikut memungkinkan perlindungan Shield Advanced untuk akun.

```
aws shield create-subscription
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai dengan AWS Shield Advanced](#) di Panduan Pengembang AWS Shield Advanced.

- Untuk detail API, lihat [CreateSubscription](#) di Referensi AWS CLI Perintah.

## **delete-protection**

Contoh kode berikut menunjukkan cara menggunakan `delete-protection`.

### AWS CLI

Untuk menghapus perlindungan AWS Shield Advanced dari AWS sumber daya

`delete-protection` Contoh berikut menghapus perlindungan AWS Shield Advanced yang ditentukan.

```
aws shield delete-protection \  
  --protection-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menghapus AWS Shield Advanced dari AWS Sumber Daya](#) di Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DeleteProtection](#) di Referensi AWS CLI Perintah.

## **describe-attack**

Contoh kode berikut menunjukkan cara menggunakan `describe-attack`.

### AWS CLI

Untuk mengambil deskripsi rinci tentang serangan

`describe-attack` Contoh berikut menampilkan rincian tentang serangan DDoS dengan ID serangan yang ditentukan. Anda dapat memperoleh ID serangan dengan menjalankan `list-attacks` perintah.

```
aws shield describe-attack --attack-id a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
  "Attack": {
    "AttackId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "ResourceArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/testElb",
    "SubResources": [
      {
        "Type": "IP",
        "Id": "192.0.2.2",
        "AttackVectors": [
          {
            "VectorType": "SYN_FLOOD",
            "VectorCounters": [
              {
                "Name": "SYN_FLOOD_BPS",
                "Max": 982184.0,
                "Average": 982184.0,
                "Sum": 11786208.0,
                "N": 12,
                "Unit": "BPS"
              }
            ]
          }
        ]
      },
      {
        "Type": "IP",
        "Id": "192.0.2.3",
        "AttackVectors": [
          {
            "VectorType": "SYN_FLOOD",
            "VectorCounters": [
              {
                "Name": "SYN_FLOOD_BPS",
```

```
        "Max": 982184.0,
        "Average": 982184.0,
        "Sum": 9821840.0,
        "N": 10,
        "Unit": "BPS"
      }
    ]
  },
  "Counters": []
},
{
  "Type": "IP",
  "Id": "192.0.2.4",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 7857472.0,
          "N": 8,
          "Unit": "BPS"
        }
      ]
    }
  ]
},
  "Counters": []
},
{
  "Type": "IP",
  "Id": "192.0.2.5",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 1964368.0,
          "N": 2,
```

```
        "Unit": "BPS"
      }
    ]
  },
  "Counters": []
},
{
  "Type": "IP",
  "Id": "2001:DB8::bcde:4321:8765:0:0",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 1964368.0,
          "N": 2,
          "Unit": "BPS"
        }
      ]
    }
  ]
},
{
  "Type": "IP",
  "Id": "192.0.2.6",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 1964368.0,
          "N": 2,
          "Unit": "BPS"
        }
      ]
    }
  ]
}
```



```
    ],
    "Counters": []
  }
],
"StartTime": 1576024927.457,
"EndTime": 1576025647.457,
"AttackCounters": [],
"AttackProperties": [
  {
    "AttackLayer": "NETWORK",
    "AttackPropertyIdentifier": "SOURCE_IP_ADDRESS",
    "TopContributors": [
      {
        "Name": "198.51.100.5",
        "Value": 2024475682
      },
      {
        "Name": "198.51.100.8",
        "Value": 1311380863
      },
      {
        "Name": "203.0.113.4",
        "Value": 900599855
      },
      {
        "Name": "198.51.100.4",
        "Value": 769417366
      },
      {
        "Name": "203.1.113.13",
        "Value": 757992847
      }
    ]
  },
  {
    "AttackLayer": "NETWORK",
    "AttackPropertyIdentifier": "SOURCE_COUNTRY",
    "TopContributors": [
      {
        "Name": "United States",
        "Value": 80938161764
      }
    ]
  }
],
```

```
    {
      "Name": "Brazil",
      "Value": 9929864330
    },
    {
      "Name": "Netherlands",
      "Value": 1635009446
    },
    {
      "Name": "Mexico",
      "Value": 144832971
    },
    {
      "Name": "Japan",
      "Value": 45369000
    }
  ],
  "Unit": "BYTES",
  "Total": 92773354841
},
{
  "AttackLayer": "NETWORK",
  "AttackPropertyIdentifier": "SOURCE_ASN",
  "TopContributors": [
    {
      "Name": "12345",
      "Value": 74953625841
    },
    {
      "Name": "12346",
      "Value": 4440087595
    },
    {
      "Name": "12347",
      "Value": 1635009446
    },
    {
      "Name": "12348",
      "Value": 1221230000
    },
    {
      "Name": "12349",
      "Value": 1199425294
    }
  ]
}
```

```
        ],
        "Unit": "BYTES",
        "Total": 92755479921
      }
    ],
    "Mitigations": []
  }
}
```

Untuk informasi selengkapnya, lihat [Meninjau Insiden DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [DescribeAttack](#) di Referensi AWS CLI Perintah.

## describe-drt-access

Contoh kode berikut menunjukkan cara menggunakan `describe-drt-access`.

### AWS CLI

Untuk mengambil deskripsi otorisasi DRT harus mengurangi serangan atas nama Anda

`describe-drt-access` Contoh berikut mengambil peran dan otorisasi bucket S3 yang dimiliki DRT, yang memungkinkannya merespons potensi serangan atas nama Anda.

```
aws shield describe-drt-access
```

Output:

```
{
  "RoleArn": "arn:aws:iam::123456789012:role/service-role/DrtRole",
  "LogBucketList": [
    "flow-logs-for-website-lb"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengotorisasi Tim Respons DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [DescribeDrtAccess](#) di Referensi AWS CLI Perintah.

## describe-emergency-contact-settings

Contoh kode berikut menunjukkan cara menggunakan `describe-emergency-contact-settings`.

### AWS CLI

Untuk mengambil alamat e-mail darurat yang Anda miliki pada file dengan DRT

`describe-emergency-contact-settings` Contoh berikut mengambil alamat e-mail yang ada di file dengan DRT untuk account. Ini adalah alamat yang harus dihubungi DRT ketika menanggapi serangan yang dicurigai.

```
aws shield describe-emergency-contact-settings
```

Output:

```
{
  "EmergencyContactList": [
    {
      "EmailAddress": "ops@example.com"
    },
    {
      "EmailAddress": "ddos-notifications@example.com"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat Cara Kerja AWS Shield < <https://docs.aws.amazon.com/waf/latest/developerguide/ddos-overview.html> > di Panduan Pengembang AWS Shield Advanced.

- Untuk detail API, lihat [DescribeEmergencyContactSettings](#) di Referensi AWS CLI Perintah.

## describe-protection

Contoh kode berikut menunjukkan cara menggunakan `describe-protection`.

### AWS CLI

Untuk mengambil detail untuk AWS perlindungan Shield Advanced

`describe-protection` Contoh berikut menampilkan detail tentang perlindungan Shield Advanced dengan ID yang ditentukan. Anda dapat memperoleh ID perlindungan dengan menjalankan `list-protections` perintah.

```
aws shield describe-protection \  
  --protection-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{  
  "Protection": {  
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "Name": "1.2.3.4",  
    "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:eip-allocation/  
eipalloc-0ac1537af40742a6d"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Menentukan Sumber Daya Anda untuk Dilindungi](#) di Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DescribeProtection](#) di Referensi AWS CLI Perintah.

## **describe-subscription**

Contoh kode berikut menunjukkan cara menggunakan `describe-subscription`.

### AWS CLI

Untuk mengambil rincian perlindungan AWS Shield Advanced untuk akun

`describe-subscription` Contoh berikut menampilkan detail tentang perlindungan Shield Advanced yang disediakan untuk akun. :

```
aws shield describe-subscription
```

Output:

```
{  
  "Subscription": {  
    "StartTime": 1534368978.0,  
    "EndTime": 1597613778.0,
```

```
"TimeCommitmentInSeconds": 63244800,
"AutoRenew": "ENABLED",
"Limits": [
  {
    "Type": "GLOBAL_ACCELERATOR",
    "Max": 1000
  },
  {
    "Type": "ROUTE53_HOSTED_ZONE",
    "Max": 1000
  },
  {
    "Type": "CF_DISTRIBUTION",
    "Max": 1000
  },
  {
    "Type": "ELB_LOAD_BALANCER",
    "Max": 1000
  },
  {
    "Type": "EC2_ELASTIC_IP_ALLOCATION",
    "Max": 1000
  }
]
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja AWS Shield](#) di Panduan Pengembang AWS Shield Advanced.

- Untuk detail API, lihat [DescribeSubscription](#) di Referensi AWS CLI Perintah.

## **disassociate-drt-log-bucket**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-drt-log-bucket`.

### AWS CLI

Untuk menghapus otorisasi DRT untuk mengakses bucket Amazon S3 atas nama Anda

`disassociate-drt-log-bucket` Contoh berikut menghapus hubungan antara DRT dan bucket S3 yang ditentukan. Setelah perintah ini selesai, DRT tidak dapat lagi mengakses bucket atas nama akun.

```
aws shield disassociate-drt-log-bucket \  
  --log-bucket flow-logs-for-website-lb
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengotorisasi Tim Respons DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [DisassociateDrtLogBucket](#) di Referensi AWS CLI Perintah.

## **disassociate-drt-role**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-drt-role`.

### AWS CLI

Untuk menghapus otorisasi DRT untuk mengurangi potensi serangan atas nama Anda

`disassociate-drt-role` Contoh berikut menghapus hubungan antara DRT dan akun. Setelah panggilan ini, DRT tidak dapat lagi mengakses atau mengelola akun Anda.

```
aws shield disassociate-drt-role
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengotorisasi Tim Respons DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [DisassociateDrtRole](#) di Referensi AWS CLI Perintah.

## **get-subscription-state**

Contoh kode berikut menunjukkan cara menggunakan `get-subscription-state`.

### AWS CLI

Untuk mengambil status langganan AWS Shield Advanced akun saat ini

`get-subscription-state` Contoh berikut mengambil status perlindungan Shield Advanced untuk akun tersebut.

```
aws shield get-subscription-state
```

Output:

```
{
  "SubscriptionState": "ACTIVE"
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja AWS Shield](#) di Panduan Pengembang AWS Shield Advanced.

- Untuk detail API, lihat [GetSubscriptionState](#) di Referensi AWS CLI Perintah.

## list-attacks

Contoh kode berikut menunjukkan cara menggunakan `list-attacks`.

### AWS CLI

Untuk mengambil ringkasan serangan dari Shield Advanced AWS

`list-attacks` Contoh berikut mengambil ringkasan serangan untuk AWS CloudFront distribusi yang ditentukan selama periode waktu yang ditentukan. Responsnya mencakup ID serangan yang dapat Anda berikan kepada `describe-attack` perintah untuk informasi rinci tentang serangan.

```
aws shield list-attacks \
  --resource-arns arn:aws:cloudfront::12345678910:distribution/E1PXMP22ZVFAOR \
  --start-time FromInclusive=1529280000,ToExclusive=1529300000
```

Output:

```
{
  "AttackSummaries": [
    {
      "AttackId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ResourceArn": "arn:aws:cloudfront::123456789012:distribution/
E1PXMP22ZVFAOR",
      "StartTime": 1529280000.0,
      "EndTime": 1529449200.0,
      "AttackVectors": [
```



```
    {
      "VectorType": "SYN_FLOOD"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Meninjau Insiden DDoS di Panduan Pengembang AWS Shield Advanced](#).

- Untuk detail API, lihat [ListAttacks](#) di Referensi AWS CLI Perintah.

## list-protections

Contoh kode berikut menunjukkan cara menggunakan `list-protections`.

### AWS CLI

Untuk mengambil ringkasan perlindungan dari Shield Advanced AWS

`list-protections` Contoh berikut mengambil ringkasan perlindungan yang diaktifkan untuk akun.

```
aws shield list-protections
```

Output:

```
{
  "Protections": [
    {
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Name": "Protection for CloudFront distribution",
      "ResourceArn": "arn:aws:cloudfront::123456789012:distribution/
E198WC25FX0WY8"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menentukan Sumber Daya Anda untuk Dilindungi](#) di Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [ListProtections](#) di Referensi AWS CLI Perintah.

## update-emergency-contact-settings

Contoh kode berikut menunjukkan cara menggunakan `update-emergency-contact-settings`.

### AWS CLI

Untuk menentukan alamat email darurat yang ada di file dengan DRT

`update-emergency-contact-settings` Contoh berikut mendefinisikan dua alamat email yang harus dihubungi DRT ketika menanggapi serangan yang dicurigai.

```
aws shield update-emergency-contact-settings \  
    --emergency-contact-list EmailAddress=ops@example.com EmailAddress=ddos-  
notifications@example.com
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Kerja AWS Shield](#) di Panduan Pengembang AWS Shield Advanced.

- Untuk detail API, lihat [UpdateEmergencyContactSettings](#) di Referensi AWS CLI Perintah.

## update-subscription

Contoh kode berikut menunjukkan cara menggunakan `update-subscription`.

### AWS CLI

Untuk mengubah langganan AWS Shield Advanced akun

`update-subscription` Contoh berikut memungkinkan perpanjangan otomatis langganan AWS Shield Advanced untuk akun tersebut.

```
aws shield update-subscription \  
    --auto-renew ENABLED
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Cara Kerja AWS Shield](#) di Panduan Pengembang AWS Shield Advanced.

- Untuk detail API, lihat [UpdateSubscription](#) di Referensi AWS CLI Perintah.

## Contoh penandatanganan menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Signer.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **cancel-signing-profile**

Contoh kode berikut menunjukkan cara menggunakan `cancel-signing-profile`.

AWS CLI

Untuk menghapus profil penandatanganan

`cancel-signing-profile` Contoh berikut menghapus profil penandatanganan yang ada dari AWS Penandatanganan.

```
aws signer cancel-signing-profile \  
  --profile-name MyProfile1
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [CancelSigningProfile](#) di Referensi AWS CLI Perintah.

## describe-signing-job

Contoh kode berikut menunjukkan cara menggunakan `describe-signing-job`.

### AWS CLI

Untuk menampilkan detail tentang pekerjaan penandatanganan

`describe-signing-job` Contoh berikut menampilkan rincian tentang pekerjaan penandatanganan yang ditentukan.

```
aws signer describe-signing-job \  
  --job-id 2065c468-73e2-4385-a6c9-0123456789abc
```

Output:

```
{  
  "status": "Succeeded",  
  "completedAt": 1568412037,  
  "platformId": "AmazonFreeRTOS-Default",  
  "signingMaterial": {  
    "certificateArn": "arn:aws:acm:us-  
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"  
  },  
  "statusReason": "Signing Succeeded",  
  "jobId": "2065c468-73e2-4385-a6c9-0123456789abc",  
  "source": {  
    "s3": {  
      "version": "PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4",  
      "bucketName": "signer-source",  
      "key": "MyCode.rb"  
    }  
  },  
  "profileName": "MyProfile2",  
  "signedObject": {  
    "s3": {  
      "bucketName": "signer-destination",  
      "key": "signed-2065c468-73e2-4385-a6c9-0123456789abc"  
    }  
  },  
  "requestedBy": "arn:aws:iam::123456789012:user/maria",  
  "createdAt": 1568412036  
}
```

- Untuk detail API, lihat [DescribeSigningJob](#) di Referensi AWS CLI Perintah.

## get-signing-platform

Contoh kode berikut menunjukkan cara menggunakan `get-signing-platform`.

### AWS CLI

Untuk menampilkan detail tentang platform penandatanganan

`get-signing-platform` Contoh berikut menampilkan rincian tentang platform penandatanganan yang ditentukan.

```
aws signer get-signing-platform \  
  --platform-id AmazonFreeRTOS-TI-CC3220SF
```

### Output:

```
{  
  "category": "AWS",  
  "displayName": "Amazon FreeRTOS SHA1-RSA CC3220SF-Format",  
  "target": "SHA1-RSA-TISHA1",  
  "platformId": "AmazonFreeRTOS-TI-CC3220SF",  
  "signingConfiguration": {  
    "encryptionAlgorithmOptions": {  
      "defaultValue": "RSA",  
      "allowedValues": [  
        "RSA"  
      ]  
    },  
    "hashAlgorithmOptions": {  
      "defaultValue": "SHA1",  
      "allowedValues": [  
        "SHA1"  
      ]  
    }  
  },  
  "maxSizeInMB": 16,  
  "partner": "AmazonFreeRTOS",  
  "signingImageFormat": {  
    "defaultFormat": "JSONEmbedded",  
    "supportedFormats": [  

```

```
        "JSONEmbedded"  
      ]  
    }  
  }
```

- Untuk detail API, lihat [GetSigningPlatform](#) di Referensi AWS CLI Perintah.

## get-signing-profile

Contoh kode berikut menunjukkan cara menggunakan `get-signing-profile`.

### AWS CLI

Untuk menampilkan detail tentang profil penandatanganan

`get-signing-profile` Contoh berikut menampilkan rincian tentang profil penandatanganan yang ditentukan.

```
aws signer get-signing-profile \  
  --profile-name MyProfile3
```

Output:

```
{  
  "platformId": "AmazonFreeRTOS-TI-CC3220SF",  
  "profileName": "MyProfile3",  
  "status": "Active",  
  "signingMaterial": {  
    "certificateArn": "arn:aws:acm:us-  
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"  
  }  
}
```

- Untuk detail API, lihat [GetSigningProfile](#) di Referensi AWS CLI Perintah.

## list-signing-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-signing-jobs`.

### AWS CLI

Untuk membuat daftar semua pekerjaan penandatanganan

`list-signing-jobs` Contoh berikut menampilkan detail tentang semua pekerjaan penandatanganan untuk akun.

```
aws signer list-signing-jobs
```

Dalam contoh ini, dua pekerjaan dikembalikan, satu berhasil, dan satu gagal.

```
{
  "jobs": [
    {
      "status": "Succeeded",
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      },
      "jobId": "2065c468-73e2-4385-a6c9-0123456789abc",
      "source": {
        "s3": {
          "version": "PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4",
          "bucketName": "signer-source",
          "key": "MyCode.rb"
        }
      },
      "signedObject": {
        "s3": {
          "bucketName": "signer-destination",
          "key": "signed-2065c468-73e2-4385-a6c9-0123456789abc"
        }
      },
      "createdAt": 1568412036
    },
    {
      "status": "Failed",
      "source": {
        "s3": {
          "version": "PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4",
          "bucketName": "signer-source",
          "key": "MyOtherCode.rb"
        }
      },
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      }
    }
  ]
}
```

```
    },
    "createdAt": 1568402690,
    "jobId": "74d9825e-22fc-4a0d-b962-0123456789abc"
  }
]
}
```

- Untuk detail API, lihat [ListSigningJobs](#) di Referensi AWS CLI Perintah.

## list-signing-platforms

Contoh kode berikut menunjukkan cara menggunakan `list-signing-platforms`.

### AWS CLI

Untuk membuat daftar semua platform penandatanganan

`list-signing-platforms` Contoh berikut menampilkan detail tentang semua platform penandatanganan yang tersedia.

```
aws signer list-signing-platforms
```

### Output:

```
{
  "platforms": [
    {
      "category": "AWS",
      "displayName": "AWS IoT Device Management SHA256-ECDSA ",
      "target": "SHA256-ECDSA",
      "platformId": "AWSIoTDeviceManagement-SHA256-ECDSA",
      "signingConfiguration": {
        "encryptionAlgorithmOptions": {
          "defaultValue": "ECDSA",
          "allowedValues": [
            "ECDSA"
          ]
        },
      },
      "hashAlgorithmOptions": {
        "defaultValue": "SHA256",
        "allowedValues": [
          "SHA256"
        ]
      }
    }
  ]
}
```



```
    ]
  }
},
"maxSizeInMB": 2048,
"partner": "AWSIoTDeviceManagement",
"signingImageFormat": {
  "defaultFormat": "JSONDetached",
  "supportedFormats": [
    "JSONDetached"
  ]
}
},
{
  "category": "AWS",
  "displayName": "Amazon FreeRTOS SHA1-RSA CC3220SF-Format",
  "target": "SHA1-RSA-TISHA1",
  "platformId": "AmazonFreeRTOS-TI-CC3220SF",
  "signingConfiguration": {
    "encryptionAlgorithmOptions": {
      "defaultValue": "RSA",
      "allowedValues": [
        "RSA"
      ]
    },
    "hashAlgorithmOptions": {
      "defaultValue": "SHA1",
      "allowedValues": [
        "SHA1"
      ]
    }
  },
  "maxSizeInMB": 16,
  "partner": "AmazonFreeRTOS",
  "signingImageFormat": {
    "defaultFormat": "JSONEmbedded",
    "supportedFormats": [
      "JSONEmbedded"
    ]
  }
},
{
  "category": "AWS",
  "displayName": "Amazon FreeRTOS SHA256-ECDSA",
  "target": "SHA256-ECDSA",
```

```

    "platformId": "AmazonFreeRTOS-Default",
    "signingConfiguration": {
      "encryptionAlgorithmOptions": {
        "defaultValue": "ECDSA",
        "allowedValues": [
          "ECDSA"
        ]
      },
      "hashAlgorithmOptions": {
        "defaultValue": "SHA256",
        "allowedValues": [
          "SHA256"
        ]
      }
    },
    "maxSizeInMB": 16,
    "partner": "AmazonFreeRTOS",
    "signingImageFormat": {
      "defaultFormat": "JSONEmbedded",
      "supportedFormats": [
        "JSONEmbedded"
      ]
    }
  }
]
}

```

- Untuk detail API, lihat [ListSigningPlatforms](#) di Referensi AWS CLI Perintah.

## list-signing-profiles

Contoh kode berikut menunjukkan cara menggunakan `list-signing-profiles`.

### AWS CLI

Untuk membuat daftar semua profil penandatanganan

`list-signing-profiles` Contoh berikut menampilkan rincian tentang semua profil penandatanganan untuk akun.

```
aws signer list-signing-profiles
```

Output:

```
{
  "profiles": [
    {
      "platformId": "AmazonFreeRTOS-TI-CC3220SF",
      "profileName": "MyProfile4",
      "status": "Active",
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      }
    },
    {
      "platformId": "AWSIoTDeviceManagement-SHA256-ECDSA",
      "profileName": "MyProfile5",
      "status": "Active",
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      }
    }
  ]
}
```

- Untuk detail API, lihat [ListSigningProfiles](#) di Referensi AWS CLI Perintah.

## put-signing-profile

Contoh kode berikut menunjukkan cara menggunakan `put-signing-profile`.

### AWS CLI

Untuk membuat profil penandatanganan

`put-signing-profile` Contoh berikut membuat profil penandatanganan menggunakan sertifikat dan platform yang ditentukan.

```
aws signer put-signing-profile \
  --profile-name MyProfile6 \
  --signing-material certificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc \
  --platform AmazonFreeRTOS-TI-CC3220SF
```

**Output:**

```
{
  "arn": "arn:aws:signer:us-west-2:123456789012:/signing-profiles/MyProfile6"
}
```

- Untuk detail API, lihat [PutSigningProfile](#) di Referensi AWS CLI Perintah.

**start-signing-job**

Contoh kode berikut menunjukkan cara menggunakan `start-signing-job`.

**AWS CLI**

Untuk memulai pekerjaan penandatanganan

`start-signing-job` Contoh berikut memulai pekerjaan penandatanganan pada kode yang ditemukan di sumber yang ditentukan. Ini menggunakan profil yang ditentukan untuk melakukan penandatanganan dan menempatkan kode yang ditandatangani di tujuan yang ditentukan.

```
aws signer start-signing-job \
  --source 's3={bucketName=signer-
source,key=MyCode.rb,version=PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4}' \
  --destination 's3={bucketName=signer-destination,prefix=signed-}' \
  --profile-name MyProfile7
```

Outputnya adalah ID dari pekerjaan penandatanganan.

```
{
  "jobId": "2065c468-73e2-4385-a6c9-0123456789abc"
}
```

- Untuk detail API, lihat [StartSigningJob](#) di Referensi AWS CLI Perintah.

**Contoh Snowball menggunakan AWS CLI**

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Snowball.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **get-snowball-usage**

Contoh kode berikut menunjukkan cara menggunakan `get-snowball-usage`.

AWS CLI

Untuk mendapatkan informasi tentang batas layanan Snowball untuk akun Anda

`get-snowball-usage` Contoh berikut menampilkan informasi tentang batas layanan Snowball untuk akun Anda, dan juga jumlah Bola Salju yang digunakan akun Anda.

```
aws snowball get-snowball-usage
```

Output:

```
{
  "SnowballLimit": 1,
  "SnowballsInUse": 0
}
```

UNTUK informasi lebih lanjut, lihat [Batas Tepi AWS Snowball di Panduan Pengembang AWS Snowball](#).

- Untuk detail API, lihat [GetSnowballUsage](#) di Referensi AWS CLI Perintah.

## list-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-jobs`.

### AWS CLI

Untuk membuat daftar pekerjaan Snowball saat ini di akun Anda

`list-jobs` Contoh berikut menampilkan array `JobListEntry` objek. Dalam contoh ini, satu pekerjaan terdaftar.

```
aws snowball list-jobs
```

Output:

```
{
  "JobListEntries": [
    {
      "CreationDate": 2016-09-27T14:50Z,
      "Description": "Important Photos 2016-08-11",
      "IsMaster": TRUE,
      "JobId": "ABCd1e324fe-022f-488e-a98b-3b0566063db1",
      "JobState": "Complete",
      "JobType": "IMPORT",
      "SnowballType": "EDGE"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Pekerjaan untuk perangkat AWS Snowball Edge di Panduan Pengembang AWS Snowball](#).

- Untuk detail API, lihat [ListJobs](#) di Referensi AWS CLI Perintah.

## Contoh Amazon SNS menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface dengan Amazon SNS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)
- [Skenario](#)

Tindakan

### **add-permission**

Contoh kode berikut menunjukkan cara menggunakan `add-permission`.

AWS CLI

Untuk menambahkan izin ke topik

`add-permission` Contoh berikut menambahkan izin untuk AWS akun 987654321098 untuk menggunakan `Publish` tindakan dengan topik yang ditentukan di bawah AWS akun 123456789012.

```
aws sns add-permission \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --label Publish-Permission \  
  --aws-account-id 987654321098 \  
  --action-name Publish
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AddPermission](#) di Referensi AWS CLI Perintah.

### **check-if-phone-number-is-opted-out**

Contoh kode berikut menunjukkan cara menggunakan `check-if-phone-number-is-opted-out`.

AWS CLI

Untuk memeriksa pesan SMS opt-out untuk nomor telepon

`check-if-phone-number-is-opted-out` Contoh berikut memeriksa apakah nomor telepon yang ditentukan dipilih untuk tidak menerima pesan SMS dari AWS akun saat ini.

```
aws sns check-if-phone-number-is-opted-out \  
  --phone-number +1555550100
```

Output:

```
{  
  "isOptedOut": false  
}
```

- Untuk detail API, lihat [CheckIfPhoneNumbersIsOptedOut](#) di Referensi AWS CLI Perintah.

## **confirm-subscription**

Contoh kode berikut menunjukkan cara menggunakan `confirm-subscription`.

AWS CLI

Untuk mengonfirmasi langganan

`confirm-subscription` Perintah berikut menyelesaikan proses konfirmasi yang dimulai saat Anda berlangganan topik SNS bernama `my-topic`. Parameter `--token` berasal dari pesan konfirmasi yang dikirim ke titik akhir notifikasi yang ditentukan dalam panggilan berlangganan.

```
aws sns confirm-subscription \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic \  
  --token  
  2336412f37fb687f5d51e6e241d7700ae02f7124d8268910b858cb4db727ceeb2474bb937929d3bdd7ce5d0cce1
```

Output:

```
{  
  "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-  
  topic:8a21d249-4329-4871-acc6-7be709c6ea7f"  
}
```

- Untuk detail API, lihat [ConfirmSubscription](#) di Referensi AWS CLI Perintah.



## create-platform-application

Contoh kode berikut menunjukkan cara menggunakan `create-platform-application`.

### AWS CLI

Untuk membuat aplikasi platform

`create-platform-application` Contoh berikut membuat aplikasi platform Google Firebase menggunakan kredensi platform yang ditentukan.

```
aws sns create-platform-application \  
  --name MyApplication \  
  --platform GCM \  
  --attributes PlatformCredential=EXAMPLEabcd12345jklm67890stuv12345bcdef
```

Output:

```
{  
  "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/GCM/  
  MyApplication"  
}
```

- Untuk detail API, lihat [CreatePlatformApplication](#) di Referensi AWS CLI Perintah.

## create-topic

Contoh kode berikut menunjukkan cara menggunakan `create-topic`.

### AWS CLI

Untuk membuat topik SNS

`create-topic` Contoh berikut membuat topik SNS bernama `my-topic`.

```
aws sns create-topic \  
  --name my-topic
```

Output:

```
{
```

```
"ResponseMetadata": {
  "RequestId": "1469e8d7-1642-564e-b85d-a19b4b341f83"
},
"TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Antarmuka Baris AWS Perintah dengan Amazon SQS dan Amazon SNS](#) di Panduan Pengguna Antarmuka Baris AWS Perintah.

- Untuk detail API, lihat [CreateTopic](#) di Referensi AWS CLI Perintah.

## delete-endpoint

Contoh kode berikut menunjukkan cara menggunakan `delete-endpoint`.

### AWS CLI

Untuk menghapus titik akhir aplikasi platform

`delete-endpoint` Contoh berikut menghapus endpoint aplikasi platform yang ditentukan.

```
aws sns delete-endpoint \
  --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteEndpoint](#) di Referensi AWS CLI Perintah.

## delete-platform-application

Contoh kode berikut menunjukkan cara menggunakan `delete-platform-application`.

### AWS CLI

Untuk menghapus aplikasi platform

`delete-platform-application` Contoh berikut menghapus aplikasi platform yang ditentukan.

```
aws sns delete-platform-application \
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/ADM/
MyApplication
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeletePlatformApplication](#) di Referensi AWS CLI Perintah.

## **delete-topic**

Contoh kode berikut menunjukkan cara menggunakan `delete-topic`.

### AWS CLI

Untuk menghapus topik SNS

`delete-topic` Contoh berikut menghapus topik SNS yang ditentukan.

```
aws sns delete-topic \  
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteTopic](#) di Referensi AWS CLI Perintah.

## **get-endpoint-attributes**

Contoh kode berikut menunjukkan cara menggunakan `get-endpoint-attributes`.

### AWS CLI

Untuk membuat daftar atribut endpoint aplikasi platform

`get-endpoint-attributes` Contoh berikut mencantumkan atribut untuk endpoint aplikasi platform tertentu.

```
aws sns get-endpoint-attributes \  
  --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/  
MyApplication/12345678-abcd-9012-efgh-345678901234
```

Output:

```
{  
  "Attributes": {  
    "Enabled": "true",
```

```
    "Token": "EXAMPLE12345..."
  }
}
```

- Untuk detail API, lihat [GetEndpointAttributes](#) di Referensi AWS CLI Perintah.

## get-platform-application-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-platform-application-attributes`.

### AWS CLI

Untuk membuat daftar atribut aplikasi platform

`get-platform-application-attributes` Contoh berikut mencantumkan atribut untuk aplikasi platform tertentu.

```
aws sns get-platform-application-attributes \
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/MPNS/
  MyApplication
```

Output:

```
{
  "Attributes": {
    "Enabled": "true",
    "SuccessFeedbackSampleRate": "100"
  }
}
```

- Untuk detail API, lihat [GetPlatformApplicationAttributes](#) di Referensi AWS CLI Perintah.

## get-sms-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-sms-attributes`.

### AWS CLI

Untuk mencantumkan atribut pesan SMS default

`get-sms-attributes` Contoh berikut mencantumkan atribut default untuk mengirim pesan SMS.

```
aws sns get-sms-attributes
```

Output:

```
{
  "attributes": {
    "DefaultSenderId": "MyName"
  }
}
```

- Untuk detail API, lihat [GetSmSatTributes](#) di Referensi Perintah.AWS CLI

## get-subscription-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-subscription-attributes`.

AWS CLI

Untuk mengambil atribut langganan untuk topik

Berikut ini `get-subscription-attributes` menampilkan atribut langganan yang ditentukan. Anda bisa mendapatkan `subscription-arn` dari output `list-subscriptions` perintah.

```
aws sns get-subscription-attributes \
  --subscription-arn "arn:aws:sns:us-west-2:123456789012:my-
  topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
```

Output:

```
{
  "Attributes": {
    "Endpoint": "my-email@example.com",
    "Protocol": "email",
    "RawMessageDelivery": "false",
    "ConfirmationWasAuthenticated": "false",
    "Owner": "123456789012",
    "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
    topic:8a21d249-4329-4871-acc6-7be709c6ea7f",
```

```

    "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
  }
}

```

- Untuk detail API, lihat [GetSubscriptionAttributes](#) di Referensi AWS CLI Perintah.

## get-topic-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-topic-attributes`.

### AWS CLI

Untuk mengambil atribut topik

`get-topic-attributes` Contoh berikut menampilkan atribut untuk topik yang ditentukan.

```

aws sns get-topic-attributes \
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"

```

### Output:

```

{
  "Attributes": {
    "SubscriptionsConfirmed": "1",
    "DisplayName": "my-topic",
    "SubscriptionsDeleted": "0",
    "EffectiveDeliveryPolicy": "{\"http\":{\"defaultHealthyRetryPolicy\":{\"minDelayTarget\":20,\"maxDelayTarget\":20,\"numRetries\":3,\"numMaxDelayRetries\":0,\"numNoDelayRetries\":0,\"numMinDelayRetries\":0,\"backoffFunction\":\"linear\"},\"disableSubscriptionOverrides\":false}}",
    "Owner": "123456789012",
    "Policy": "{\"Version\":\"2008-10-17\",\"Id\":\"__default_policy_ID\",\"Statement\":[{\"Sid\":\"__default_statement_ID\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":[\"SNS:Subscribe\",\"SNS:ListSubscriptionsByTopic\",\"SNS>DeleteTopic\",\"SNS:GetTopicAttributes\",\"SNS:Publish\",\"SNS:RemovePermission\",\"SNS:AddPermission\",\"SNS:SetTopicAttributes\"],\"Resource\":\"arn:aws:sns:us-west-2:123456789012:my-topic\",\"Condition\":{\"StringEquals\":{\"AWS:SourceOwner\":\"0123456789012\"}}}]}",
    "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
    "SubscriptionsPending": "0"
  }
}

```

- Untuk detail API, lihat [GetTopicAttributes](#) di Referensi AWS CLI Perintah.

## **list-endpoints-by-platform-application**

Contoh kode berikut menunjukkan cara menggunakan `list-endpoints-by-platform-application`.

### AWS CLI

Untuk membuat daftar titik akhir untuk aplikasi platform

`list-endpoints-by-platform-application` Contoh berikut mencantumkan titik akhir dan atribut endpoint untuk aplikasi platform tertentu.

```
aws sns list-endpoints-by-platform-application \
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/
  MyApplication
```

Output:

```
{
  "Endpoints": [
    {
      "Attributes": {
        "Token": "EXAMPLE12345...",
        "Enabled": "true"
      },
      "EndpointArn": "arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234"
    }
  ]
}
```

- Untuk detail API, lihat [ListEndpointsByPlatformApplication](#) di Referensi AWS CLI Perintah.

## **list-phone-numbers-opted-out**

Contoh kode berikut menunjukkan cara menggunakan `list-phone-numbers-opted-out`.

### AWS CLI

Untuk membuat daftar opt-out pesan SMS

`list-phone-numbers-opted-out` Contoh berikut mencantumkan nomor telepon yang dipilih untuk tidak menerima pesan SMS.

```
aws sns list-phone-numbers-opted-out
```

Output:

```
{
  "phoneNumbers": [
    "+15555550100"
  ]
}
```

- Untuk detail API, lihat [ListPhoneNumbersOptedOut](#) di Referensi AWS CLI Perintah.

## `list-platform-applications`

Contoh kode berikut menunjukkan cara menggunakan `list-platform-applications`.

AWS CLI

Untuk membuat daftar aplikasi platform

`list-platform-applications` Contoh berikut mencantumkan aplikasi platform untuk ADM dan MPNS.

```
aws sns list-platform-applications
```

Output:

```
{
  "PlatformApplications": [
    {
      "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/ADM/MyApplication",
      "Attributes": {
        "SuccessFeedbackSampleRate": "100",
        "Enabled": "true"
      }
    },
    {
```



```
    "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/MPNS/MyOtherApplication",
    "Attributes": {
      "SuccessFeedbackSampleRate": "100",
      "Enabled": "true"
    }
  ]
}
```

- Untuk detail API, lihat [ListPlatformApplications](#) di Referensi AWS CLI Perintah.

## list-subscriptions-by-topic

Contoh kode berikut menunjukkan cara menggunakan `list-subscriptions-by-topic`.

### AWS CLI

Untuk membuat daftar langganan yang terkait dengan topik

Berikut ini `list-subscriptions-by-topic` mengambil daftar langganan SNS yang terkait dengan topik yang ditentukan.

```
aws sns list-subscriptions-by-topic \
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Output:

```
{
  "Subscriptions": [
    {
      "Owner": "123456789012",
      "Endpoint": "my-email@example.com",
      "Protocol": "email",
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
      "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
    }
  ]
}
```

- Untuk detail API, lihat [ListSubscriptionsByTopic](#) di Referensi AWS CLI Perintah.

## list-subscriptions

Contoh kode berikut menunjukkan cara menggunakan `list-subscriptions`.

### AWS CLI

Untuk membuat daftar langganan SNS Anda

`list-subscriptions` Contoh berikut menampilkan daftar langganan SNS di akun Anda AWS .

```
aws sns list-subscriptions
```

Output:

```
{
  "Subscriptions": [
    {
      "Owner": "123456789012",
      "Endpoint": "my-email@example.com",
      "Protocol": "email",
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
      "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
    }
  ]
}
```

- Untuk detail API, lihat [ListSubscriptions](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag untuk suatu topik

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk topik Amazon SNS yang ditentukan.

```
aws sns list-tags-for-resource \
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic
```

**Output:**

```
{
  "Tags": [
    {
      "Key": "Team",
      "Value": "Alpha"
    }
  ]
}
```

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

**list-topics**

Contoh kode berikut menunjukkan cara menggunakan `list-topics`.

**AWS CLI**

Untuk membuat daftar topik SNS Anda

`list-topics` Contoh berikut mencantumkan semua topik SNS di AWS akun Anda.

```
aws sns list-topics
```

**Output:**

```
{
  "Topics": [
    {
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
    }
  ]
}
```

- Untuk detail API, lihat [ListTopics](#) di Referensi AWS CLI Perintah.

**opt-in-phone-number**

Contoh kode berikut menunjukkan cara menggunakan `opt-in-phone-number`.

## AWS CLI

Untuk ikut serta dalam pesan SMS

`opt-in-phone-number` Contoh berikut memilih nomor telepon yang ditentukan untuk menerima pesan SMS.

```
aws sns opt-in-phone-number \  
  --phone-number +15555550100
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [OptInPhoneNumber](#) di Referensi AWS CLI Perintah.

## publish

Contoh kode berikut menunjukkan cara menggunakan `publish`.

## AWS CLI

Contoh 1: Untuk mempublikasikan pesan ke topik

`publish` Contoh berikut menerbitkan pesan yang ditentukan ke topik SNS yang ditentukan. Pesan berasal dari file teks, yang memungkinkan Anda untuk memasukkan jeda baris.

```
aws sns publish \  
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic" \  
  --message file://message.txt
```

Isi dari `message.txt`:

```
Hello World  
Second Line
```

Output:

```
{  
  "MessageId": "123a45b6-7890-12c3-45d6-111122223333"  
}
```

## Contoh 2: Untuk mempublikasikan pesan SMS ke nomor telepon

publishContoh berikut menerbitkan pesan Hello world! ke nomor +1-555-555-0100 telepon.

```
aws sns publish \  
  --message "Hello world!" \  
  --phone-number +1-555-555-0100
```

### Output:

```
{  
  "MessageId": "123a45b6-7890-12c3-45d6-333322221111"  
}
```

- Untuk detail API, lihat [Menerbitkan](#) di Referensi AWS CLI Perintah.

## put-data-protection-policy

Contoh kode berikut menunjukkan cara menggunakanput-data-protection-policy.

### AWS CLI

Untuk menetapkan kebijakan perlindungan data

Contoh 1: Untuk menolak penerbit mempublikasikan pesan dengan CreditCardNumber

put-data-protection-policyContoh berikut menyangkal penerbit dari mempublikasikan pesan dengan. CreditCardNumber

```
aws sns put-data-protection-policy \  
  --resource-arn arn:aws:sns:us-east-1:123456789012:mytopic \  
  --data-protection-policy "{\"Name\": \"data_protection_policy\", \"Description\  
\": \"Example data protection policy\", \"Version\": \"2021-06-01\", \"Statement\  
\": [{\"DataDirection\": \"Inbound\", \"Principal\": [\"*\"], \"DataIdentifier\":  
[\"arn:aws:dataprotection::aws:data-identifier/CreditCardNumber\"], \"Operation\":  
{\"Deny\": {}}}]}"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk memuat parameter dari file

Berikut ini `put-data-protection-policy` memuat parameter dari file.

```
aws sns put-data-protection-policy \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --data-protection-policy file://policy.json
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutDataProtectionPolicy](#) di Referensi AWS CLI Perintah.

## **remove-permission**

Contoh kode berikut menunjukkan cara menggunakan `remove-permission`.

### AWS CLI

Untuk menghapus izin dari topik

`remove-permission` Contoh berikut menghapus izin `Publish-Permission` dari topik yang ditentukan.

```
aws sns remove-permission \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --label Publish-Permission
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RemovePermission](#) di Referensi AWS CLI Perintah.

## **set-endpoint-attributes**

Contoh kode berikut menunjukkan cara menggunakan `set-endpoint-attributes`.

### AWS CLI

Untuk mengatur atribut titik akhir

`set-endpoint-attributes` Contoh berikut menonaktifkan endpoint aplikasi platform yang ditentukan.

```
aws sns set-endpoint-attributes \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --platform-attributes {\"
```

```
--endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/MyApplication/12345678-abcd-9012-efgh-345678901234 \  
--attributes Enabled=false
```

Output:

```
{  
  "Attributes": {  
    "Enabled": "false",  
    "Token": "EXAMPLE12345..."  
  }  
}
```

- Untuk detail API, lihat [SetEndpointAttributes](#) di Referensi AWS CLI Perintah.

## set-platform-application-attributes

Contoh kode berikut menunjukkan cara menggunakan `set-platform-application-attributes`.

### AWS CLI

Untuk mengatur atribut aplikasi platform

`set-platform-application-attributes` Contoh berikut menetapkan `EventDeliveryFailure` atribut untuk aplikasi platform yang ditentukan ke ARN dari topik Amazon SNS yang ditentukan.

```
aws sns set-platform-application-attributes \  
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/MyApplication \  
  --attributes EventDeliveryFailure=arn:aws:sns:us-west-2:123456789012:AnotherTopic
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetPlatformApplicationAttributes](#) di Referensi AWS CLI Perintah.

## set-sms-attributes

Contoh kode berikut menunjukkan cara menggunakan `set-sms-attributes`.

## AWS CLI

Untuk mengatur atribut pesan SMS

`set-sms-attributes` Contoh berikut menetapkan ID pengirim default untuk pesan SMS ke `MyName`.

```
aws sns set-sms-attributes \  
  --attributes DefaultSenderId=MyName
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetSmSatTributes](#) di Referensi Perintah.AWS CLI

## `set-subscription-attributes`

Contoh kode berikut menunjukkan cara menggunakan `set-subscription-attributes`.

## AWS CLI

Untuk mengatur atribut langganan

`set-subscription-attributes` Contoh berikut menetapkan `RawMessageDelivery` atribut ke langganan SQS.

```
aws sns set-subscription-attributes \  
  --subscription-arn arn:aws:sns:us-  
east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \  
  --attribute-name RawMessageDelivery \  
  --attribute-value true
```

Perintah ini tidak menghasilkan output.

`set-subscription-attributes` Contoh berikut menetapkan `FilterPolicy` atribut ke langganan SQS.

```
aws sns set-subscription-attributes \  
  --subscription-arn arn:aws:sns:us-  
east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \  
  --attribute-name FilterPolicy \  
  --attribute-value "{ \"anyMandatoryKey\": [\"any\", \"of\", \"these\"] }"
```



Perintah ini tidak menghasilkan output.

`set-subscription-attributes` Contoh berikut menghapus `FilterPolicy` atribut dari langganan SQS.

```
aws sns set-subscription-attributes \  
  --subscription-arn arn:aws:sns:us-  
east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \  
  --attribute-name FilterPolicy \  
  --attribute-value "{}"
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetSubscriptionAttributes](#) di Referensi AWS CLI Perintah.

## **set-topic-attributes**

Contoh kode berikut menunjukkan cara menggunakan `set-topic-attributes`.

### AWS CLI

Untuk menetapkan atribut untuk topik

`set-topic-attributes` Contoh berikut menetapkan `DisplayName` atribut untuk topik yang ditentukan.

```
aws sns set-topic-attributes \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --attribute-name DisplayName \  
  --attribute-value MyTopicDisplayName
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [SetTopicAttributes](#) di Referensi AWS CLI Perintah.

## **subscribe**

Contoh kode berikut menunjukkan cara menggunakan `subscribe`.

### AWS CLI

Untuk berlangganan topik

**subscribe** Perintah berikut berlangganan alamat email ke topik yang ditentukan.

```
aws sns subscribe \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic \  
  --protocol email \  
  --notification-endpoint my-email@example.com
```

Output:

```
{  
  "SubscriptionArn": "pending confirmation"  
}
```

- Untuk detail API, lihat [Berlangganan](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke topik

`tag-resource` Contoh berikut menambahkan tag metadata ke topik Amazon SNS yang ditentukan.

```
aws sns tag-resource \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --tags Key=Team,Value=Alpha
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## unsubscribe

Contoh kode berikut menunjukkan cara menggunakan `unsubscribe`.

### AWS CLI

Untuk berhenti berlangganan dari suatu topik

`unsubscribe` Contoh berikut menghapus langganan yang ditentukan dari suatu topik.

```
aws sns unsubscribe \  
  --subscription-arn arn:aws:sns:us-west-2:0123456789012:my-  
  topic:8a21d249-4329-4871-acc6-7be709c6ea7f
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [Berhenti berlangganan](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari topik

`untag-resource` Contoh berikut menghapus tag apa pun dengan kunci yang ditentukan dari topik Amazon SNS yang ditentukan.

```
aws sns untag-resource \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --tag-keys Team
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## Skenario

Buat titik akhir platform untuk pemberitahuan push

Contoh kode berikut menunjukkan cara membuat titik akhir platform untuk notifikasi push Amazon SNS.

### AWS CLI

Untuk membuat endpoint aplikasi platform

`create-platform-endpoint` Contoh berikut membuat titik akhir untuk aplikasi platform tertentu menggunakan token yang ditentukan.

```
aws sns create-platform-endpoint \  
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/  
MyApplication \  
  --token EXAMPLE12345...
```

Output:

```
{  
  "EndpointArn": "arn:aws:sns:us-west-2:1234567890:endpoint/GCM/  
MyApplication/12345678-abcd-9012-efgh-345678901234"  
}
```

## Contoh Amazon SQS menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon SQS. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-permission**

Contoh kode berikut menunjukkan cara menggunakan `add-permission`.

AWS CLI

Untuk menambahkan izin ke antrian

Contoh ini memungkinkan AWS akun yang ditentukan untuk mengirim pesan ke antrian yang ditentukan.

Perintah:

```
aws sqs add-permission --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --label SendMessagesFromMyQueue --aws-account-ids 12345EXAMPLE --actions SendMessage
```

Output:

```
None.
```

- Untuk detail API, lihat [AddPermission](#) di Referensi AWS CLI Perintah.

## cancel-message-move-task

Contoh kode berikut menunjukkan cara menggunakan `cancel-message-move-task`.

AWS CLI

Untuk membatalkan tugas pemindahan pesan

`cancel-message-move-task` Contoh berikut membatalkan tugas pemindahan pesan yang ditentukan.

```
aws sqs cancel-message-move-task \
  --task-handle AQEB6nR4...Hz1vZQ==
```

Output:

```
{
  "ApproximateNumberOfMessagesMoved": 102
}
```

Untuk informasi selengkapnya, lihat [Izin Amazon SQS API: Tindakan dan referensi sumber daya di Panduan Pengembang](#).

- Untuk detail API, lihat [CancelMessageMoveTask](#) di Referensi AWS CLI Perintah.

## change-message-visibility-batch

Contoh kode berikut menunjukkan cara menggunakan `change-message-visibility-batch`.

### AWS CLI

Untuk mengubah visibilitas batas waktu beberapa pesan sebagai batch

Contoh ini mengubah visibilitas batas waktu 2 pesan yang ditentukan menjadi 10 jam (10 jam\* 60 menit\* 60 detik).

Perintah:

```
aws sqs change-message-visibility-batch --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://change-message-visibility-batch.json
```

Berkas masukan (`change-message-visibility-batch.json`):

```
[
  {
    "Id": "FirstMessage",
    "ReceiptHandle": "AQEBhz2q...Jf3kaw==",
    "VisibilityTimeout": 36000
  },
  {
    "Id": "SecondMessage",
    "ReceiptHandle": "AQEBkTUH...HifSnw==",
    "VisibilityTimeout": 36000
  }
]
```

Output:

```
{
  "Successful": [
    {
      "Id": "SecondMessage"
    },
    {
      "Id": "FirstMessage"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [ChangeMessageVisibilityBatch](#) di Referensi AWS CLI Perintah.

## change-message-visibility

Contoh kode berikut menunjukkan cara menggunakan `change-message-visibility`.

### AWS CLI

Untuk mengubah visibilitas batas waktu pesan

Contoh ini mengubah visibilitas batas waktu pesan yang ditentukan menjadi 10 jam (10 jam \* 60 menit \* 60 detik).

Perintah:

```
aws sqs change-message-visibility --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --receipt-handle AQEBTpyI...t6HyQg== --
visibility-timeout 36000
```

Output:

```
None.
```

- Untuk detail API, lihat [ChangeMessageVisibility](#) di Referensi AWS CLI Perintah.

## create-queue

Contoh kode berikut menunjukkan cara menggunakan `create-queue`.

### AWS CLI

Untuk membuat antrian

Contoh ini membuat antrian dengan nama yang ditentukan, menetapkan periode penyimpanan pesan menjadi 3 hari (3 hari \* 24 jam \* 60 menit \* 60 detik), dan mengatur antrian surat mati antrian ke antrian yang ditentukan dengan jumlah penerimaan maksimum 1.000 pesan.

Perintah:

```
aws sqs create-queue --queue-name MyQueue --attributes file://create-queue.json
```

File masukan (buat-antrean.json):

```
{
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-east-1:80398EXAMPLE:MyDeadLetterQueue\", \"maxReceiveCount\":\"1000\"}\",
  "MessageRetentionPeriod": "259200"
}
```

Output:

```
{
  "QueueUrl": "https://queue.amazonaws.com/80398EXAMPLE/MyQueue"
}
```

- Untuk detail API, lihat [CreateQueue](#) di Referensi AWS CLI Perintah.

## delete-message-batch

Contoh kode berikut menunjukkan cara menggunakan `delete-message-batch`.

### AWS CLI

Untuk menghapus beberapa pesan sebagai batch

Contoh ini menghapus pesan yang ditentukan.

Perintah:

```
aws sqs delete-message-batch --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://delete-message-batch.json
```

Berkas masukan (delete-message-batch.json):

```
[
  {
    "Id": "FirstMessage",
    "ReceiptHandle": "AQEB1mg1...Z4GuLw=="
  },
]
```



```
{
  "Id": "SecondMessage",
  "ReceiptHandle": "AQEBLsYM...VQubAA=="
}
]
```

Output:

```
{
  "Successful": [
    {
      "Id": "FirstMessage"
    },
    {
      "Id": "SecondMessage"
    }
  ]
}
```

- Untuk detail API, lihat [DeleteMessageBatch](#) di Referensi AWS CLI Perintah.

## delete-message

Contoh kode berikut menunjukkan cara menggunakan `delete-message`.

### AWS CLI

Untuk menghapus pesan

Contoh ini menghapus pesan yang ditentukan.

Perintah:

```
aws sqs delete-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --receipt-handle AQEBRXTo...q2doVA==
```

Output:

```
None.
```

- Untuk detail API, lihat [DeleteMessage](#) di Referensi AWS CLI Perintah.

## delete-queue

Contoh kode berikut menunjukkan cara menggunakan `delete-queue`.

### AWS CLI

Untuk menghapus antrian

Contoh ini menghapus antrian yang ditentukan.

Perintah:

```
aws sqs delete-queue --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewerQueue
```

Output:

```
None.
```

- Untuk detail API, lihat [DeleteQueue](#) di Referensi AWS CLI Perintah.

## get-queue-attributes

Contoh kode berikut menunjukkan cara menggunakan `get-queue-attributes`.

### AWS CLI

Untuk mendapatkan atribut antrian

Contoh ini mendapatkan semua atribut antrian yang ditentukan.

Perintah:

```
aws sqs get-queue-attributes --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names All
```

Output:

```
{
  "Attributes": {
    "ApproximateNumberOfMessagesNotVisible": "0",
```

```

    "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-
east-1:80398EXAMPLE:MyDeadLetterQueue\", \"maxReceiveCount\":1000}\",
    \"MessageRetentionPeriod\": \"345600\",
    \"ApproximateNumberOfMessagesDelayed\": \"0\",
    \"MaximumMessageSize\": \"262144\",
    \"CreatedTimestamp\": \"1442426968\",
    \"ApproximateNumberOfMessages\": \"0\",
    \"ReceiveMessageWaitTimeSeconds\": \"0\",
    \"DelaySeconds\": \"0\",
    \"VisibilityTimeout\": \"30\",
    \"LastModifiedTimestamp\": \"1442426968\",
    \"QueueArn\": \"arn:aws:sqs:us-east-1:80398EXAMPLE:MyNewQueue\"
  }
}

```

Contoh ini hanya mendapatkan ukuran pesan maksimum antrian yang ditentukan dan atribut batas waktu visibilitas.

Perintah:

```

aws sqs get-queue-attributes --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyNewQueue --attribute-names MaximumMessageSize
VisibilityTimeout

```

Output:

```

{
  \"Attributes\": {
    \"VisibilityTimeout\": \"30\",
    \"MaximumMessageSize\": \"262144\"
  }
}

```

- Untuk detail API, lihat [GetQueueAttributes](#) di Referensi AWS CLI Perintah.

## get-queue-url

Contoh kode berikut menunjukkan cara menggunakan `get-queue-url`.

### AWS CLI

Untuk mendapatkan URL antrian

Contoh ini mendapatkan URL antrian yang ditentukan.

Perintah:

```
aws sqs get-queue-url --queue-name MyQueue
```

Output:

```
{
  "QueueUrl": "https://queue.amazonaws.com/80398EXAMPLE/MyQueue"
}
```

- Untuk detail API, lihat [GetQueueUrl](#) di Referensi AWS CLI Perintah.

## **list-dead-letter-source-queues**

Contoh kode berikut menunjukkan cara menggunakan `list-dead-letter-source-queues`.

AWS CLI

Untuk membuat daftar antrian sumber surat mati

Contoh ini mencantumkan antrian yang terkait dengan antrian sumber huruf mati yang ditentukan.

Perintah:

```
aws sqs list-dead-letter-source-queues --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue
```

Output:

```
{
  "queueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue"
  ]
}
```

- Untuk detail API, lihat [ListDeadLetterSourceQueues](#) di Referensi AWS CLI Perintah.

## list-message-move-tasks

Contoh kode berikut menunjukkan cara menggunakan `list-message-move-tasks`.

### AWS CLI

Untuk mencantumkan tugas pemindahan pesan

`list-message-move-tasks` Contoh berikut mencantumkan 2 tugas pemindahan pesan terbaru dalam antrian yang ditentukan.

```
aws sqs list-message-move-tasks \  
  --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue \  
  --max-results 2
```

Output:

```
{  
  "Results": [  
    {  
      "TaskHandle": "AQEB6nR4...HzlvZQ==",  
      "Status": "RUNNING",  
      "SourceArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1",  
      "DestinationArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2",  
      "MaxNumberOfMessagesPerSecond": 50,  
      "ApproximateNumberOfMessagesMoved": 203,  
      "ApproximateNumberOfMessagesToMove": 30,  
      "StartedTimestamp": 1442428276921  
    },  
    {  
      "Status": "COMPLETED",  
      "SourceArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1",  
      "DestinationArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2",  
      "ApproximateNumberOfMessagesMoved": 29,  
      "ApproximateNumberOfMessagesToMove": 0,  
      "StartedTimestamp": 1342428272093  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [izin Amazon SQS API: Tindakan dan referensi sumber daya di Panduan Pengembang](#).

- Untuk detail API, lihat [ListMessageMoveTasks](#) di Referensi AWS CLI Perintah.

## list-queue-tags

Contoh kode berikut menunjukkan cara menggunakan `list-queue-tags`.

### AWS CLI

Untuk mencantumkan semua tag alokasi biaya untuk antrian

`list-queue-tags` Contoh berikut menampilkan semua tag alokasi biaya yang terkait dengan antrian yang ditentukan.

```
aws sqs list-queue-tags \
  --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue
```

Output:

```
{
  "Tags": {
    "Team": "Alpha"
  }
}
```

Untuk informasi selengkapnya, lihat [Daftar Tag Alokasi Biaya di Panduan](#) Pengembang Layanan Antrian Sederhana Amazon.

- Untuk detail API, lihat [ListQueueTags](#) di Referensi AWS CLI Perintah.

## list-queues

Contoh kode berikut menunjukkan cara menggunakan `list-queues`.

### AWS CLI

Untuk daftar antrian

Contoh ini mencantumkan semua antrian.

Perintah:

```
aws sqs list-queues
```

**Output:**

```
{
  "QueueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/TestQueue1",
    "https://queue.amazonaws.com/80398EXAMPLE/TestQueue2"
  ]
}
```

Contoh ini hanya mencantumkan antrian yang dimulai dengan “My”.

**Perintah:**

```
aws sqs list-queues --queue-name-prefix My
```

**Output:**

```
{
  "QueueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue"
  ]
}
```

- Untuk detail API, lihat [ListQueues](#) di Referensi AWS CLI Perintah.

**purge-queue**

Contoh kode berikut menunjukkan cara menggunakan `purge-queue`.

**AWS CLI**

Untuk membersihkan antrian

Contoh ini menghapus semua pesan dalam antrian yang ditentukan.

**Perintah:**

```
aws sqs purge-queue --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue
```

Output:

```
None.
```

- Untuk detail API, lihat [PurgeQueue](#) di Referensi AWS CLI Perintah.

## receive-message

Contoh kode berikut menunjukkan cara menggunakan `receive-message`.

### AWS CLI

Untuk menerima pesan

Contoh ini menerima hingga 10 pesan yang tersedia, mengembalikan semua atribut yang tersedia.

Perintah:

```
aws sqs receive-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names All --message-attribute-names All --max-number-of-messages 10
```

Output:

```
{
  "Messages": [
    {
      "Body": "My first message.",
      "ReceiptHandle": "AQEBzbVv...fqNzFw==",
      "MD5ofBody": "1000f835...a35411fa",
      "MD5ofMessageAttributes": "9424c491...26bc3ae7",
      "MessageId": "d6790f8d-d575-4f01-bc51-40122EXAMPLE",
      "Attributes": {
        "ApproximateFirstReceiveTimestamp": "1442428276921",
        "SenderId": "AIDAIKMSNQ7EXAMPLE",
        "ApproximateReceiveCount": "5",
        "SentTimestamp": "1442428276921"
      }
    },
  ],
}
```



```
"MessageAttributes": {
  "PostalCode": {
    "DataType": "String",
    "StringValue": "ABC123"
  },
  "City": {
    "DataType": "String",
    "StringValue": "Any City"
  }
}
]
```

Contoh ini menerima pesan berikutnya yang tersedia, hanya mengembalikan `SentTimestamp` atribut `SenderId` dan serta atribut `PostalCode` pesan.

Perintah:

```
aws sqs receive-message --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names SenderId SentTimestamp
--message-attribute-names PostalCode
```

Output:

```
{
  "Messages": [
    {
      "Body": "My first message.",
      "ReceiptHandle": "AQEB6nR4...HzlvZQ==",
      "MD5ofBody": "1000f835...a35411fa",
      "MD5ofMessageAttributes": "b8e89563...e088e74f",
      "MessageId": "d6790f8d-d575-4f01-bc51-40122EXAMPLE",
      "Attributes": {
        "SenderId": "AIDAIIAZKMSNQ7TEXAMPLE",
        "SentTimestamp": "1442428276921"
      },
      "MessageAttributes": {
        "PostalCode": {
          "DataType": "String",
          "StringValue": "ABC123"
        }
      }
    }
  ]
}
```

```
}  
]  
}
```

- Untuk detail API, lihat [ReceiveMessage](#) di Referensi AWS CLI Perintah.

## **remove-permission**

Contoh kode berikut menunjukkan cara menggunakan `remove-permission`.

### AWS CLI

Untuk menghapus izin

Contoh ini menghapus izin dengan label yang ditentukan dari antrian yang ditentukan.

Perintah:

```
aws sqs remove-permission --queue-url https://sqs.us-  
east-1.amazonaws.com/80398EXAMPLE/MyQueue --label SendMessageFromMyQueue
```

Output:

```
None.
```

- Untuk detail API, lihat [RemovePermission](#) di Referensi AWS CLI Perintah.

## **send-message-batch**

Contoh kode berikut menunjukkan cara menggunakan `send-message-batch`.

### AWS CLI

Untuk mengirim beberapa pesan sebagai batch

Contoh ini mengirimkan 2 pesan dengan badan pesan tertentu, periode tunda, dan atribut pesan, ke antrian yang ditentukan.

Perintah:

```
aws sqs send-message-batch --queue-url https://sqs.us-  
east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://send-message-batch.json
```

## Berkas masukan (send-message-batch.json):

```
[
  {
    "Id": "FuelReport-0001-2015-09-16T140731Z",
    "MessageBody": "Fuel report for account 0001 on 2015-09-16 at 02:07:31 PM.",
    "DelaySeconds": 10,
    "MessageAttributes": {
      "SellerName": {
        "DataType": "String",
        "StringValue": "Example Store"
      },
      "City": {
        "DataType": "String",
        "StringValue": "Any City"
      },
      "Region": {
        "DataType": "String",
        "StringValue": "WA"
      },
      "PostalCode": {
        "DataType": "String",
        "StringValue": "99065"
      },
      "PricePerGallon": {
        "DataType": "Number",
        "StringValue": "1.99"
      }
    }
  },
  {
    "Id": "FuelReport-0002-2015-09-16T140930Z",
    "MessageBody": "Fuel report for account 0002 on 2015-09-16 at 02:09:30 PM.",
    "DelaySeconds": 10,
    "MessageAttributes": {
      "SellerName": {
        "DataType": "String",
        "StringValue": "Example Fuels"
      },
      "City": {
        "DataType": "String",
        "StringValue": "North Town"
      },
      "Region": {
```

```
        "DataType": "String",
        "StringValue": "WA"
    },
    "PostalCode": {
        "DataType": "String",
        "StringValue": "99123"
    },
    "PricePerGallon": {
        "DataType": "Number",
        "StringValue": "1.87"
    }
}
]
```

Output:

```
{
  "Successful": [
    {
      "MD5ofMessageBody": "203c4a38...7943237e",
      "MD5ofMessageAttributes": "10809b55...baf283ef",
      "Id": "FuelReport-0001-2015-09-16T140731Z",
      "MessageId": "d175070c-d6b8-4101-861d-adeb3EXAMPLE"
    },
    {
      "MD5ofMessageBody": "2cf0159a...c1980595",
      "MD5ofMessageAttributes": "55623928...ae354a25",
      "Id": "FuelReport-0002-2015-09-16T140930Z",
      "MessageId": "f9b7d55d-0570-413e-b9c5-a9264EXAMPLE"
    }
  ]
}
```

- Untuk detail API, lihat [SendMessageBatch](#) di Referensi AWS CLI Perintah.

## send-message

Contoh kode berikut menunjukkan cara menggunakan `send-message`.

### AWS CLI

Untuk mengirim pesan

Contoh ini mengirimkan pesan dengan isi pesan tertentu, periode tunda, dan atribut pesan, ke antrian yang ditentukan.

Perintah:

```
aws sqs send-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --message-body "Information about the largest city in Any Region." --delay-seconds 10 --message-attributes file://send-message.json
```

File masukan (send-message.json):

```
{
  "City": {
    "DataType": "String",
    "StringValue": "Any City"
  },
  "Greeting": {
    "DataType": "Binary",
    "BinaryValue": "Hello, World!"
  },
  "Population": {
    "DataType": "Number",
    "StringValue": "1250800"
  }
}
```

Output:

```
{
  "MD5ofMessageBody": "51b0a325...39163aa0",
  "MD5ofMessageAttributes": "00484c68...59e48f06",
  "MessageId": "da68f62c-0c07-4bee-bf5f-7e856EXAMPLE"
}
```

- Untuk detail API, lihat [SendMessage](#) di Referensi AWS CLI Perintah.

## set-queue-attributes

Contoh kode berikut menunjukkan cara menggunakan `set-queue-attributes`.

## AWS CLI

Untuk mengatur atribut antrian

Contoh ini menetapkan antrian yang ditentukan ke penundaan pengiriman 10 detik, ukuran pesan maksimum 128 KB (128 KB \* 1.024 byte), periode penyimpanan pesan 3 hari (3 hari \* 24 jam \* 60 menit \* 60 detik), waktu tunggu pesan terima 20 detik, dan batas waktu visibilitas default 60 detik. Contoh ini juga mengaitkan antrian surat mati yang ditentukan dengan jumlah penerimaan maksimum 1.000 pesan.

Perintah:

```
aws sqs set-queue-attributes --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue --attributes file://set-queue-attributes.json
```

Berkas masukan (set-queue-attributes.json):

```
{
  "DelaySeconds": "10",
  "MaximumMessageSize": "131072",
  "MessageRetentionPeriod": "259200",
  "ReceiveMessageWaitTimeSeconds": "20",
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-east-1:80398EXAMPLE:MyDeadLetterQueue\",\"maxReceiveCount\":\"1000\"}",
  "VisibilityTimeout": "60"
}
```

Output:

```
None.
```

- Untuk detail API, lihat [SetQueueAttributes](#) di Referensi AWS CLI Perintah.

## start-message-move-task

Contoh kode berikut menunjukkan cara menggunakan `start-message-move-task`.

## AWS CLI

Contoh 1: \*Untuk memulai tugas pemindahan pesan\*

`start-message-move-task` Contoh berikut memulai tugas pemindahan pesan untuk menggerakkan ulang pesan dari antrian huruf mati yang ditentukan ke antrian sumber.

```
aws sqs start-message-move-task \  
  --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue
```

Output:

```
{  
  "TaskHandle": "AQEB6nR4...Hz1vZQ=="  
}
```

Untuk informasi selengkapnya, lihat [Ini adalah judul topik](#) di Nama panduan Anda.

Contoh 2: \*Untuk memulai tugas pemindahan pesan dengan tingkat maksimum\*

`start-message-move-task` Contoh berikut memulai tugas pemindahan pesan untuk menggerakkan ulang pesan dari antrian huruf mati yang ditentukan ke antrian tujuan yang ditentukan dengan kecepatan maksimum 50 pesan per detik.

```
aws sqs start-message-move-task \  
  --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1 \  
  --destination-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2 \  
  --max-number-of-messages-per-second 50
```

Output:

```
{  
  "TaskHandle": "AQEB6nR4...Hz1vZQ=="  
}
```

Untuk informasi selengkapnya, lihat [Izin Amazon SQS API: Tindakan dan referensi sumber daya di Panduan Pengembang](#).

- Untuk detail API, lihat [StartMessageMoveTask](#) di Referensi AWS CLI Perintah.

## tag-queue

Contoh kode berikut menunjukkan cara menggunakan `tag-queue`.

## AWS CLI

Untuk menambahkan tag alokasi biaya ke antrian

`tag-queue` Contoh berikut menambahkan tag alokasi biaya ke antrian Amazon SQS yang ditentukan.

```
aws sqs tag-queue \  
  --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue \  
  --tags Priority=Highest
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan Tag Alokasi Biaya](#) di Panduan Pengembang Layanan Antrian Sederhana Amazon.

- Untuk detail API, lihat [TagQueue](#) di Referensi AWS CLI Perintah.

## untag-queue

Contoh kode berikut menunjukkan cara menggunakan `untag-queue`.

## AWS CLI

Untuk menghapus tag alokasi biaya dari antrian

`untag-queue` Contoh berikut menghapus tag alokasi biaya dari antrian Amazon SQS yang ditentukan.

```
aws sqs untag-queue \  
  --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue \  
  --tag-keys "Priority"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menambahkan Tag Alokasi Biaya](#) di Panduan Pengembang Layanan Antrian Sederhana Amazon.

- Untuk detail API, lihat [UntagQueue](#) di Referensi AWS CLI Perintah.



## Contoh Storage Gateway menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Storage Gateway.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **describe-gateway-information**

Contoh kode berikut menunjukkan cara menggunakan `describe-gateway-information`.

AWS CLI

Untuk menggambarkan gateway

`describe-gateway-information` Perintah berikut mengembalikan metadata tentang gateway yang ditentukan. Untuk menentukan gateway mana yang akan dijelaskan, gunakan Amazon Resource Name (ARN) dari gateway dalam perintah.

Contoh ini menentukan gateway dengan id `sgw-12A3456B` di akun `123456789012`:

```
aws storagegateway describe-gateway-information --gateway-arn
"arn:aws:storagegateway:us-west-2:123456789012:gateway/sgw-12A3456B"
```

Perintah ini mengeluarkan blok JSON yang berisi metadata tentang gateway seperti namanya, antarmuka jaringan, zona waktu yang dikonfigurasi, dan status (apakah gateway sedang berjalan atau tidak).

- Untuk detail API, lihat [DescribeGatewayInformation](#) di Referensi AWS CLI Perintah.

## list-file-shares

Contoh kode berikut menunjukkan cara menggunakan `list-file-shares`.

### AWS CLI

Untuk daftar berbagi file

command-name Contoh berikut mencantumkan widget yang tersedia di AWS akun Anda.

```
aws storagegateway list-file-shares \  
  --gateway-arn arn:aws:storagegateway:us-east-1:209870788375:gateway/sgw-FB02E292
```

Output:

```
{  
  "FileShareInfoList": [  
    {  
      "FileShareType": "NFS",  
      "FileShareARN": "arn:aws:storagegateway:us-east-1:111122223333:share/  
share-2FA12345",  
      "FileShareId": "share-2FA12345",  
      "FileShareStatus": "AVAILABLE",  
      "GatewayARN": "arn:aws:storagegateway:us-east-1:111122223333:gateway/  
sgw-FB0AAAAA"  
    }  
  ],  
  "Marker": null  
}
```

Untuk informasi selengkapnya, lihat [ListFileShares](#) di Referensi API Layanan AWS Storage Gateway.

- Untuk detail API, lihat [ListFileShares](#) di Referensi AWS CLI Perintah.

## list-gateways

Contoh kode berikut menunjukkan cara menggunakan `list-gateways`.

### AWS CLI

Untuk membuat daftar gateway untuk akun

`list-gateways` Perintah berikut mencantumkan semua gateway yang ditentukan untuk akun:

```
aws storagegateway list-gateways
```

Perintah ini mengeluarkan blok JSON yang berisi daftar gateway Amazon Resource Names (ARN).

- Untuk detail API, lihat [ListGateways](#) di Referensi AWS CLI Perintah.

## **list-volumes**

Contoh kode berikut menunjukkan cara menggunakan `list-volumes`.

### AWS CLI

Untuk membuat daftar volume yang dikonfigurasi untuk gateway

`list-volumes` Perintah berikut mengembalikan daftar volume dikonfigurasi untuk gateway yang ditentukan. Untuk menentukan gateway mana yang akan dijelaskan, gunakan Amazon Resource Name (ARN) dari gateway dalam perintah.

Contoh ini menentukan gateway dengan id `sgw-12A3456B` di akun `123456789012`:

```
aws storagegateway list-volumes --gateway-arn "arn:aws:storagegateway:us-west-2:123456789012:gateway/sgw-12A3456B"
```

Perintah ini mengeluarkan blok JSON yang daftar volume yang mencakup tipe dan ARN untuk setiap volume.

- Untuk detail API, lihat [ListVolumes](#) di Referensi AWS CLI Perintah.

## **refresh-cache**

Contoh kode berikut menunjukkan cara menggunakan `refresh-cache`.

### AWS CLI

Untuk me-refresh cache berbagi file

`refresh-cache` Contoh berikut menyegarkan cache untuk berbagi file yang ditentukan.

```
aws storagegateway refresh-cache \  
  --file-share-arn arn:aws:storagegateway:us-east-1:111122223333:share/  
share-2FA12345
```

Output:

```
{  
  "FileShareARN": "arn:aws:storagegateway:us-east-1:111122223333:share/  
share-2FA12345",  
  "NotificationId": "4954d4b1-abcd-ef01-1234-97950a7d3483"  
}
```

Untuk informasi selengkapnya, lihat [ListFileShares](#) di Referensi API Layanan AWS Storage Gateway.

- Untuk detail API, lihat [RefreshCached](#) di Referensi AWS CLI Perintah.

## AWS STS contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS STS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **assume-role-with-saml**

Contoh kode berikut menunjukkan cara menggunakan `assume-role-with-saml`.

## AWS CLI

Untuk mendapatkan kredensi jangka pendek untuk peran yang diautentikasi dengan SAFL

`assume-role-with-saml` Perintah berikut mengambil satu set kredensi jangka pendek untuk peran IAM. `TestSaml` Permintaan dalam contoh ini diautentikasi dengan menggunakan pernyataan SAFL yang disediakan oleh penyedia identitas Anda saat Anda mengautentikasi.

```
aws sts assume-role-with-saml \
  --role-arn arn:aws:iam::123456789012:role/TestSaml \
  --principal-arn arn:aws:iam::123456789012:saml-provider/SAML-test \
  --saml-assertion
"VERYLONGENCODEDASSERTIONEXAMPLExzYW1s0kF1ZG11bmN1PmJsYW5rPC9zYW1s0kF1ZG11bmN1Pjwvc2FtbDpBd
+PHNhbWw6TmFtZULEIEZvcmlhdD0idXJu0m9hc2lz0m5hbWVz0nRj01NBTUw6Mi4w0m5hbWVpZC1mb3JtYXQ6dHJhbnN
+PHNhbWw6U3ViamVjdENvbMzpcmlhdGlvbiBNZXRob2Q9InVybjpvYXNpczpuYW1lczp0YzpqTQU1M0jIuMDpjbTpiZWw"
```

Output:

```
{
  "Issuer": "https://integ.example.com/idp/shibboleth</Issuer",
  "AssumedRoleUser": {
    "Arn": "arn:aws:sts::123456789012:assumed-role/TestSaml",
    "AssumedRoleId": "AR0456EXAMPLE789:TestSaml"
  },
  "Credentials": {
    "AccessKeyId": "ASIAV3ZUEFP6EXAMPLE",
    "SecretAccessKey": "8P+SQvWIuLnKhh8d++jpw0nNmQRBZvNEXAMPLEKEY",
    "SessionToken": "IQoJb3JpZ2luX2VjE0z//////////
wEXAMPLEtMSJHMEUCIDoKK3JH9uGQE1z0sINr5M4jk
+Na8KHDcCYRVjJCZEv0AiEA30vJGtw1EcVi0leS2vhs8VdCKFJQPQrmGdeehM4IC1NtBmUpp2wUE8phUZampKsburED
+xo0rKwT38xVqr7ZD0u0iPPkUL64lIZbqBAz
+scqKmlzm8FDrypNC9Yjc8fP0Ln9FX9KSYvKTr4rvx3iSI1TJabIQwj2ICCR/oLxBA==",
    "Expiration": "2019-11-01T20:26:47Z"
  },
  "Audience": "https://signin.aws.amazon.com/saml",
  "SubjectType": "transient",
  "PackedPolicySize": "6",
  "NameQualifier": "SbdG0nUkh1i4+EXAMPLExL/jEvs=",
  "Subject": "SamlExample"
}
```

Untuk informasi lebih lanjut, lihat [Meminta Kredensial Keamanan Sementara](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [AssumeRoleWithSaml](#) di Referensi AWS CLI Perintah.

## **assume-role-with-web-identity**

Contoh kode berikut menunjukkan cara menggunakan `assume-role-with-web-identity`.

### AWS CLI

Untuk mendapatkan kredensi jangka pendek untuk peran yang diautentikasi dengan Identitas Web (OAuth 2.0)

`assume-role-with-web-identity` Perintah berikut mengambil satu set kredensi jangka pendek untuk peran IAM. `app1` Permintaan diautentikasi dengan menggunakan token identitas web yang disediakan oleh penyedia identitas web yang ditentukan. Dua kebijakan tambahan diterapkan pada sesi untuk lebih membatasi apa yang dapat dilakukan pengguna. Kredensi yang dikembalikan kedaluwarsa satu jam setelah dibuat.

```
aws sts assume-role-with-web-identity \
  --duration-seconds 3600 \
  --role-session-name "app1" \
  --provider-id "www.amazon.com" \
  --policy-arns "arn:aws:iam::123456789012:policy/
q=webidentitydemopolicy1","arn:aws:iam::123456789012:policy/webidentitydemopolicy2"
  \
  --role-arn arn:aws:iam::123456789012:role/FederatedWebIdentityRole \
  --web-identity-token "Atza
%7CIQEBLjAsAhRFiXuWpUXuRvQ9PZL3GMFcYevydwIUFAHZwXZXXXXXXXXXJnrulxKDHwy87oGKPznh0D6bEQZTSCzyoC
CrKqjG7nPBjNIL016GGvuS5gSvPRUxWES3VYfm1w17WTI7jn-Pcb6M-
buCgHhF0zTQxod27L9Cqn0Lio7N3gZAGpsp6n1-
AJB0CJckcyXe2c6uD0sr0JeZlKUm2eTDVMf8IehDVI0r1Q0nTV6KzzAI30Y87Vd_cVMQ"
```

### Output:

```
{
  "SubjectFromWebIdentityToken": "amzn1.account.AF6RH07KZU5XRvQJGxK6HB56KR2A"
  "Audience": "client.5498841531868486423.1548@apps.example.com",
  "AssumedRoleUser": {
    "Arn": "arn:aws:sts::123456789012:assumed-role/FederatedWebIdentityRole/
app1",
```

```

    "AssumedRoleId": "AROACLKWSQRAOEXAMPLE:app1"
  }
  "Credentials": {
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYzEXAMPLEKEY",
    "SessionToken": "AQoEXAMPLEH4aoAH0gNCAPyJxz4B1CFFxWNE1OPTgk5TthT
+FvwnKwRc0IfrrRh3c/LTo6UDdyJw00vEVPvLXCrrrUtdnniCEXAMPLE/
IvU1dYUg2RVAJBanLiHb4IgrmpRV3zrkuWJ0gQs8IZZaIv2BXIa2R40lgkBN9bkUDNCJiBeb/
AX1zBBko7b15fjrBs2+cTQtpZ3CYWFXG8C5zqx37wn0E49mR1/+0tkIKG07fAE",
    "Expiration": "2020-05-19T18:06:10+00:00"
  },
  "Provider": "www.amazon.com"
}

```

Untuk informasi lebih lanjut, lihat [Meminta Kredensial Keamanan Sementara](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [AssumeRoleWithWebIdentity](#) di Referensi AWS CLI Perintah.

## assume-role

Contoh kode berikut menunjukkan cara menggunakan `assume-role`.

### AWS CLI

Untuk mengambil peran

`assume-role` Perintah berikut mengambil satu set kredensi jangka pendek untuk peran IAM. `s3-access-example`

```

aws sts assume-role \
  --role-arn arn:aws:iam::123456789012:role/xaccounts3access \
  --role-session-name s3-access-example

```

Output:

```

{
  "AssumedRoleUser": {
    "AssumedRoleId": "ARO3XFRBF535PLBIFPI4:s3-access-example",
    "Arn": "arn:aws:sts::123456789012:assumed-role/xaccounts3access/s3-access-example"
  },
  "Credentials": {

```

```

    "SecretAccessKey": "9drTJvcXLB89EXAMPLEL8923FB892xMFI",
    "SessionToken": "AQoXdzELDDY//////////
wEaoAK1wvxJY12r2IrDFT2IvAzTCn3zHoZ7YNtpiQLF0MqZye/qwjzP2iEXAMPLEbw/
m3hsj8VBTkPORGvr9jM5sgP+w9IZWZnU+LWhmg
+a5fDi2oTGUYcdg9uexQ4mtCHIHfi4citgqZTgco40Yqr4lIlo4V2b2Dyauk0eYFNebHtY1FVgAUj
+7Indz3LU0aTWk1WKIjHmMCIoTkyYp/k7kUG7moeEYKSitwQIi6Gjn+nzM
+PtoA3685ixzv0R7i5rjQi0YE0lf1oeie3bDiNHncmzosRM6SFipzSvp6h/32xQuZsjcypmwsPSDtTPYcs0+YN/8BRi2
IcrxSpnWEXAMPLEXSDFTAQAM6Dl9zR0tXoybnlrZIwMLlMi1Kcgo50ytwU=",
    "Expiration": "2016-03-15T00:05:07Z",
    "AccessKeyId": "ASIAJEXAMPLEXEG2JICEA"
}
}

```

Output dari perintah berisi kunci akses, kunci rahasia, dan token sesi yang dapat Anda gunakan untuk AWS mengautentikasi.

Untuk penggunaan AWS CLI, Anda dapat mengatur profil bernama yang terkait dengan peran. Saat Anda menggunakan profil, AWS CLI akan memanggil peran `assume-dan` mengelola kredensial untuk Anda. Untuk informasi selengkapnya, lihat [Menggunakan peran IAM di AWS CLI](#) di Panduan Pengguna AWS CLI.

- Untuk detail API, lihat [AssumeRole](#) di Referensi AWS CLI Perintah.

## decode-authorization-message

Contoh kode berikut menunjukkan cara menggunakan `decode-authorization-message`.

### AWS CLI

Untuk memecahkan kode pesan otorisasi yang dikodekan dikembalikan sebagai tanggapan atas permintaan

`decode-authorization-message` Contoh berikut menerjemahkan informasi tambahan tentang status otorisasi permintaan dari pesan yang disandikan yang dikembalikan sebagai tanggapan atas permintaan Amazon Web Services.

```

aws sts decode-authorization-message \
  --encoded-message EXAMPLEWodyRNrtlQARDip-
eTA6i6Dr1UhHhPQrLWB_1Ab15pAKx19mPDLexYcGBreyIKQC1BGBIpBKr3dFDkwqe07e2NMk5j_hmzAiChJN-8oy3Ewi
0jau7BMj0TWw0tHPHv_Zaz87yENDipr745EjQwRd5LaoL3vN8_5ZfA9UiBMKDgVh1gjqZJFUiQoubv78V1RbHNYnK44E
p0u3FZjwYStfvTb3GHs3-6rLribG09jZ0ktkFE6vqx1FzLyeDr4P2ihC1wty9tArCvvGzIAUNmARQJ2VVWPxioqgoqCz
JWP5pwe_mAyqh0NLw-r1S56YC_90onj9A80sNrHlI-

```



```
tIiNd7tgNTYzDuPQYD2FMDBnp82V9eVmYGtPp5NIeSpuf3f0HanFuBZgENxZQZ2d1H3xJGMTtYayzZrRXjiq_SfX9zeB
FaoPIb8LmmKVBLpIB0iFhU9sEHPqKHVPi6jdxXqKaZaFGvYVmV0iuQdNQkuyk0p067P0FrZECLjj0tNPB0ZCcuEKEXAM
```

## Output:

```
{
  "DecodedMessage": "{\"allowed\":false,\"explicitDeny\":true,\"matchedStatements\
\":{\\"items\":[\\"statementId\":\\\"VisualEditor0\\\",\\\"effect\":\\\"DENY\\\",\\\"principals\
\":{\\"items\":[\\"value\":\\\"AROA123456789EXAMPLE\\\"]]},\\\"principalGroups\
\":{\\"items\":[]},\\\"actions\":{\\"items\":[\\"value\":\\\"ec2:RunInstances\
\\\"]]},\\\"resources\":{\\"items\":[\\"value\":\\\"*\\\"]]},\\\"conditions\":{\\"items\
\":[\\"\"]]},\\\"failures\":{\\"items\":[]},\\\"context\":{\\"principal\":{\\"id\":\
\\\"AROA123456789EXAMPLE:Ana\\\",\\\"arn\":\\\"arn:aws:sts::111122223333:assumed-role/\
Developer/Ana\\\"},\\\"action\":\\\"RunInstances\\\",\\\"resource\":\\\"arn:aws:ec2:us-
east-1:111122223333:instance/*\\\",\\\"conditions\":{\\"items\":[\\"key\":\
\\\"ec2:MetadataHttpPutResponseHopLimit\\\",\\\"values\":{\\"items\":[\\"value\
\":\\\"2\\\"]]},{\\"key\":\\\"ec2:InstanceMarketType\\\",\\\"values\":{\\"items\":[\\"value\
\":\\\"on-demand\\\"]]},{\\"key\":\\\"aws:Resource\\\",\\\"values\":{\\"items\":[\\"value\
\":\\\"instance/*\\\"]]},{\\"key\":\\\"aws:Account\\\",\\\"values\":{\\"items\":[\\"value\
\":\\\"111122223333\\\"]]},{\\"key\":\\\"ec2:AvailabilityZone\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"us-east-1f\\\"]]},{\\"key\":\\\"ec2:ecsOptimized\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"false\\\"]]},{\\"key\":\\\"ec2:IsLaunchTemplateResource\\\",\\\"values\
\":{\\"items\":[\\"value\":\\\"false\\\"]]},{\\"key\":\\\"ec2:InstanceType\\\",\\\"values\
\":{\\"items\":[\\"value\":\\\"t2.micro\\\"]]},{\\"key\":\\\"ec2:RootDeviceType\\\",\\\"values\
\":{\\"items\":[\\"value\":\\\"efs\\\"]]},{\\"key\":\\\"aws:Region\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"us-east-1\\\"]]},{\\"key\":\\\"ec2:MetadataHttpEndpoint\\\",\\\"values\
\":{\\"items\":[\\"value\":\\\"enabled\\\"]]},{\\"key\":\\\"aws:Service\\\",\\\"values\
\":{\\"items\":[\\"value\":\\\"ec2\\\"]]},{\\"key\":\\\"ec2:InstanceID\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"*\\\"]]},{\\"key\":\\\"ec2:MetadataHttpTokens\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"required\\\"]]},{\\"key\":\\\"aws:Type\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"instance\\\"]]},{\\"key\":\\\"ec2:Tenancy\\\",\\\"values\":{\\"items\
\":[\\"value\":\\\"default\\\"]]},{\\"key\":\\\"ec2:Region\\\",\\\"values\":{\\"items\
\":[\\"value\
\":\\\"us-east-1\\\"]]},{\\"key\":\\\"aws:ARN\\\",\\\"values\":{\\"items\":[\\"value\
\":\
\\\"arn:aws:ec2:us-east-1:111122223333:instance/*\\\"]}}}}}}}"
}
```

Untuk informasi selengkapnya, lihat [Logika evaluasi kebijakan](#) di Panduan Pengguna AWS IAM.

- Untuk detail API, lihat [DecodeAuthorizationMessage](#) di Referensi AWS CLI Perintah.

## get-caller-identity

Contoh kode berikut menunjukkan cara menggunakan `get-caller-identity`.

## AWS CLI

Untuk mendapatkan detail tentang identitas IAM saat ini

`get-caller-identity` Perintah berikut menampilkan informasi tentang identitas IAM yang digunakan untuk mengautentikasi permintaan. Penelepon adalah pengguna IAM.

```
aws sts get-caller-identity
```

Output:

```
{
  "UserId": "AIDASAMPLEUSERID",
  "Account": "123456789012",
  "Arn": "arn:aws:iam::123456789012:user/DevAdmin"
}
```

- Untuk detail API, lihat [GetCallerIdentity](#) di Referensi AWS CLI Perintah.

## **get-federation-token**

Contoh kode berikut menunjukkan cara menggunakan `get-federation-token`.

## AWS CLI

Untuk mengembalikan satu set kredensi keamanan sementara menggunakan kredensial kunci akses pengguna IAM

`get-federation-token` Contoh berikut mengembalikan satu set kredensial keamanan sementara (terdiri dari ID kunci akses, kunci akses rahasia, dan token keamanan) untuk pengguna. Anda harus memanggil `GetFederationToken` operasi menggunakan kredensial keamanan jangka panjang dari pengguna IAM.

```
aws sts get-federation-token \
  --name Bob \
  --policy file://myfile.json \
  --policy-arns arn=arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess \
  --duration-seconds 900
```

Isi dari `myfile.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:Describe*",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "elasticloadbalancing:Describe*",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:ListMetrics",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:Describe*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "autoscaling:Describe*",
      "Resource": "*"
    }
  ]
}
```

### Output:

```
{
  "Credentials": {
    "AccessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "SessionToken": "EXAMPLEpZ2luX2VjEGoaCXVzLXdlc3QtMiJIMEYCIQC/  
W9pL5ArQyDD5JwFL3/h5+WGopQ24GEXweNctwhi9sgIhAMkg  
+MZE35iWM8s4r5Lr25f9rSTVPFH98G42QQuWMTfKq0DCOP/////////  
wEQAxoMNDUy0TI1MTcwNTA3Igxuy3A0puuoLsk3MJwqgQPg8Q0d9HuoClUxq26wnc/nm  
+eZLjHDyGf2KUAHK2DuaS/nrGSEXAMPLE",
    "Expiration": "2023-12-20T02:06:07+00:00"
  },
}
```

```

    "FederatedUser": {
      "FederatedUserId": "111122223333:Bob",
      "Arn": "arn:aws:sts::111122223333:federated-user/Bob"
    },
    "PackedPolicySize": 36
  }

```

Untuk informasi lebih lanjut, lihat [Meminta Kredensial Keamanan Sementara](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetFederationToken](#) di Referensi AWS CLI Perintah.

## get-session-token

Contoh kode berikut menunjukkan cara menggunakan `get-session-token`.

### AWS CLI

Untuk mendapatkan satu set kredensi jangka pendek untuk identitas IAM

`get-session-token` Perintah berikut mengambil satu set kredensi jangka pendek untuk identitas IAM yang membuat panggilan. Kredensi yang dihasilkan dapat digunakan untuk permintaan di mana otentikasi multi-faktor (MFA) diperlukan oleh kebijakan. Kredensialnya kedaluwarsa 15 menit setelah dibuat.

```

aws sts get-session-token \
  --duration-seconds 900 \
  --serial-number "YourMFADeviceSerialNumber" \
  --token-code 123456

```

### Output:

```

{
  "Credentials": {
    "AccessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYzEXAMPLEKEY",
    "SessionToken": "AQoEXAMPLEH4aoAH0gNCAPyJxz4B1CFFxWNE1OPTgk5TthT
+FvwqnKwRc0IfrrRh3c/LTo6UDdyJw00vEVPvLXCrrrUtdnniCEXAMPLE/
IvU1dYUg2RVAJBanLiHb4IgRmpRV3zrkuWJ0gQs8IZZaIv2BXIa2R40lgkBN9bkUDNCJiBeb/
AX1zBBko7b15fjrBs2+cTQtpZ3CYWFXG8C5zqx37wn0E49mRl/+0tkIKG07fAE",
    "Expiration": "2020-05-19T18:06:10+00:00"
  }
}

```

```
}
```

Untuk informasi lebih lanjut, lihat [Meminta Kredensial Keamanan Sementara](#) dalam AWS Panduan Pengguna IAM.

- Untuk detail API, lihat [GetSessionToken](#) di Referensi AWS CLI Perintah.

## AWS Support contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS Support.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-attachments-to-set**

Contoh kode berikut menunjukkan cara menggunakan `add-attachments-to-set`.

AWS CLI

Untuk menambahkan lampiran ke set

`add-attachments-to-set` Contoh berikut menambahkan gambar ke set yang kemudian dapat Anda tentukan untuk kasus dukungan di AWS akun Anda.

```
aws support add-attachments-to-set \
  --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQ1Br70-
  G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE" \
```

```
--attachments fileName=troubleshoot-screenshot.png,data=base64-encoded-string
```

Output:

```
{
  "attachmentSetId": "as-2f5a6faa2a4a1e600-mu-nk5xQ1Br70-
G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE",
  "expiryTime": "2020-05-14T17:04:40.790+0000"
}
```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [AddAttachmentsToSet](#) di Referensi AWS CLI Perintah.

## add-communication-to-case

Contoh kode berikut menunjukkan cara menggunakan `add-communication-to-case`.

### AWS CLI

Untuk menambahkan komunikasi ke sebuah kasus

`add-communication-to-case` Contoh berikut menambahkan komunikasi ke kasus dukungan di AWS akun Anda.

```
aws support add-communication-to-case \
  --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
  --communication-body "I'm attaching a set of images to this case." \
  --cc-email-addresses "myemail@example.com" \
  --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQ1Br70-
G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE"
```

Output:

```
{
  "result": true
}
```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [AddCommunicationToCase](#) di Referensi AWS CLI Perintah.

## create-case

Contoh kode berikut menunjukkan cara menggunakan `create-case`.

### AWS CLI

Untuk membuat kasus

`create-case` Contoh berikut membuat kasus dukungan untuk AWS akun Anda.

```
aws support create-case \  
  --category-code "using-aws" \  
  --cc-email-addresses "myemail@example.com" \  
  --communication-body "I want to learn more about an AWS service." \  
  --issue-type "technical" \  
  --language "en" \  
  --service-code "general-info" \  
  --severity-code "low" \  
  --subject "Question about my account"
```

Output:

```
{  
  "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47"  
}
```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [CreateCase](#) di Referensi AWS CLI Perintah.

## describe-attachment

Contoh kode berikut menunjukkan cara menggunakan `describe-attachment`.

### AWS CLI

Untuk menggambarkan lampiran

`describe-attachment` Contoh berikut mengembalikan informasi tentang lampiran dengan ID yang ditentukan.

```
aws support describe-attachment \  
  --attachment-id "12345678910-2013-c4c1d2bf33c5cf47"
```

```
--attachment-id "attachment-KBnjrNrePd9D6Jx0-Mm00xZuDEaL2JAj_0-
gJv9qqDooTipsz3V1Nb19rCfkZneeQeDPgp8X1iVJyHH7UuhZDdNeqGoduZsPrAhyMakqlc60-
iJjL5HqyYGiT1FG8EXAMPLE"
```

Output:

```
{
  "attachment": {
    "fileName": "troubleshoot-screenshot.png",
    "data": "base64-blob"
  }
}
```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [DescribeAttachment](#) di Referensi AWS CLI Perintah.

## describe-cases

Contoh kode berikut menunjukkan cara menggunakan `describe-cases`.

### AWS CLI

Untuk menggambarkan sebuah kasus

`describe-cases` Contoh berikut mengembalikan informasi tentang kasus dukungan yang ditentukan di AWS akun Anda.

```
aws support describe-cases \
  --display-id "1234567890" \
  --after-time "2020-03-23T21:31:47.774Z" \
  --include-resolved-cases \
  --language "en" \
  --no-include-communications \
  --max-item 1
```

Output:

```
{
  "cases": [
    {
```



```

    "status": "resolved",
    "ccEmailAddresses": [],
    "timeCreated": "2020-03-23T21:31:47.774Z",
    "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
    "severityCode": "low",
    "language": "en",
    "categoryCode": "using-aws",
    "serviceCode": "general-info",
    "submittedBy": "myemail@example.com",
    "displayId": "1234567890",
    "subject": "Question about my account"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [DescribeCases](#) di Referensi AWS CLI Perintah.

## describe-communications

Contoh kode berikut menunjukkan cara menggunakan `describe-communications`.

### AWS CLI

Untuk menggambarkan komunikasi terbaru untuk suatu kasus

`describe-communications` Contoh berikut mengembalikan komunikasi terbaru untuk kasus dukungan yang ditentukan di AWS akun Anda.

```

aws support describe-communications \
  --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
  --after-time "2020-03-23T21:31:47.774Z" \
  --max-item 1

```

### Output:

```

{
  "communications": [
    {
      "body": "I want to learn more about an AWS service.",
      "attachmentSet": [],

```

```

        "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
        "timeCreated": "2020-05-12T23:12:35.000Z",
        "submittedBy": "Amazon Web Services"
    }
],
"NextToken": "eyJ1ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQEXAMPLE=="
}

```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [DescribeCommunications](#) di Referensi AWS CLI Perintah.

## describe-services

Contoh kode berikut menunjukkan cara menggunakan `describe-services`.

### AWS CLI

Untuk daftar AWS layanan dan kategori layanan

`describe-services` Contoh berikut mencantumkan kategori layanan yang tersedia untuk meminta informasi umum.

```
aws support describe-services \
  --service-code-list "general-info"
```

Output:

```

{
  "services": [
    {
      "code": "general-info",
      "name": "General Info and Getting Started",
      "categories": [
        {
          "code": "charges",
          "name": "How Will I Be Charged?"
        },
        {
          "code": "gdpr-queries",
          "name": "Data Privacy Query"
        }
      ]
    }
  ]
}

```

```

    {
      "code": "reserved-instances",
      "name": "Reserved Instances"
    },
    {
      "code": "resource",
      "name": "Where is my Resource?"
    },
    {
      "code": "using-aws",
      "name": "Using AWS & Services"
    },
    {
      "code": "free-tier",
      "name": "Free Tier"
    },
    {
      "code": "security-and-compliance",
      "name": "Security & Compliance"
    },
    {
      "code": "account-structure",
      "name": "Account Structure"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [DescribeServices](#) di Referensi AWS CLI Perintah.

## describe-severity-levels

Contoh kode berikut menunjukkan cara menggunakan `describe-severity-levels`.

### AWS CLI

Untuk daftar tingkat keparahan yang tersedia

`describe-severity-levels` Contoh berikut mencantumkan tingkat keparahan yang tersedia untuk kasus dukungan.

```
aws support describe-severity-levels
```

Output:

```
{
  "severityLevels": [
    {
      "code": "low",
      "name": "Low"
    },
    {
      "code": "normal",
      "name": "Normal"
    },
    {
      "code": "high",
      "name": "High"
    },
    {
      "code": "urgent",
      "name": "Urgent"
    },
    {
      "code": "critical",
      "name": "Critical"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memilih tingkat keparahan](#) di Panduan Pengguna AWS Support.

- Untuk detail API, lihat [DescribeSeverityLevels](#) di Referensi AWS CLI Perintah.

## **describe-trusted-advisor-check-refresh-statuses**

Contoh kode berikut menunjukkan cara menggunakan `describe-trusted-advisor-check-refresh-statuses`.

### AWS CLI

Untuk mencantumkan status penyegaran pemeriksaan AWS Trusted Advisor

`describe-trusted-advisor-check-refresh-statuses` Contoh berikut mencantumkan status penyegaran untuk dua pemeriksaan Trusted Advisor: Izin Bucket Amazon S3 dan Penggunaan IAM.

```
aws support describe-trusted-advisor-check-refresh-statuses \
  --check-id "Pfx0RwqBli" "zXCkfM1nI3"
```

Output:

```
{
  "statuses": [
    {
      "checkId": "Pfx0RwqBli",
      "status": "none",
      "millisUntilNextRefreshable": 0
    },
    {
      "checkId": "zXCkfM1nI3",
      "status": "none",
      "millisUntilNextRefreshable": 0
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [AWS Trusted Advisor](#) di Panduan Pengguna AWS Support.

- Untuk detail API, lihat [DescribeTrustedAdvisorCheckRefreshStatuses](#) di Referensi AWS CLI Perintah.

## **describe-trusted-advisor-check-result**

Contoh kode berikut menunjukkan cara menggunakan `describe-trusted-advisor-check-result`.

AWS CLI

Untuk mencantumkan hasil pemeriksaan AWS Trusted Advisor

`describe-trusted-advisor-check-result` Contoh berikut mencantumkan hasil pemeriksaan IAM Use.

```
aws support describe-trusted-advisor-check-result \
```

```
--check-id "zXCkfM1nI3"
```

Output:

```
{
  "result": {
    "checkId": "zXCkfM1nI3",
    "timestamp": "2020-05-13T21:38:05Z",
    "status": "ok",
    "resourcesSummary": {
      "resourcesProcessed": 1,
      "resourcesFlagged": 0,
      "resourcesIgnored": 0,
      "resourcesSuppressed": 0
    },
    "categorySpecificSummary": {
      "costOptimizing": {
        "estimatedMonthlySavings": 0.0,
        "estimatedPercentMonthlySavings": 0.0
      }
    },
    "flaggedResources": [
      {
        "status": "ok",
        "resourceId": "47DEQpj8HBSa-_TImW-5JCeuQeRkm5NMpJWZEXAMPLE",
        "isSuppressed": false
      }
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [AWS Trusted Advisor](#) di Panduan Pengguna AWS Support.

- Untuk detail API, lihat [DescribeTrustedAdvisorCheckResult](#) di Referensi AWS CLI Perintah.

## **describe-trusted-advisor-check-summaries**

Contoh kode berikut menunjukkan cara menggunakan `describe-trusted-advisor-check-summaries`.

### AWS CLI

Untuk membuat daftar ringkasan cek Trusted AWS Advisor

`describe-trusted-advisor-check-summaries` Contoh berikut mencantumkan hasil untuk dua pemeriksaan Trusted Advisor: Izin Bucket Amazon S3 dan Penggunaan IAM.

```
aws support describe-trusted-advisor-check-summaries \
  --check-ids "Pfx0RwqBli" "zXCkfM1nI3"
```

Output:

```
{
  "summaries": [
    {
      "checkId": "Pfx0RwqBli",
      "timestamp": "2020-05-13T21:38:12Z",
      "status": "ok",
      "hasFlaggedResources": true,
      "resourcesSummary": {
        "resourcesProcessed": 44,
        "resourcesFlagged": 0,
        "resourcesIgnored": 0,
        "resourcesSuppressed": 0
      },
      "categorySpecificSummary": {
        "costOptimizing": {
          "estimatedMonthlySavings": 0.0,
          "estimatedPercentMonthlySavings": 0.0
        }
      }
    },
    {
      "checkId": "zXCkfM1nI3",
      "timestamp": "2020-05-13T21:38:05Z",
      "status": "ok",
      "hasFlaggedResources": true,
      "resourcesSummary": {
        "resourcesProcessed": 1,
        "resourcesFlagged": 0,
        "resourcesIgnored": 0,
        "resourcesSuppressed": 0
      },
      "categorySpecificSummary": {
        "costOptimizing": {
          "estimatedMonthlySavings": 0.0,
          "estimatedPercentMonthlySavings": 0.0
        }
      }
    }
  ]
}
```

```

    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [AWS Trusted Advisor](#) di Panduan Pengguna AWS Support.

- Untuk detail API, lihat [DescribeTrustedAdvisorCheckSummaries](#) di Referensi AWS CLI Perintah.

## describe-trusted-advisor-checks

Contoh kode berikut menunjukkan cara menggunakan `describe-trusted-advisor-checks`.

### AWS CLI

Untuk daftar cek AWS Trusted Advisor yang tersedia

`describe-trusted-advisor-checks` Contoh berikut mencantumkan cek Trusted Advisor yang tersedia di akun Anda AWS . Informasi ini mencakup nama cek, ID, deskripsi, kategori, dan metadata. Perhatikan bahwa output dipersingkat agar mudah dibaca.

```
aws support describe-trusted-advisor-checks \
  --language "en"
```

### Output:

```
{
  "checks": [
    {
      "id": "zXCkfM1nI3",
      "name": "IAM Use",
      "description": "Checks for your use of AWS Identity and Access
Management (IAM). You can use IAM to create users, groups, and roles in AWS, and
you can use permissions to control access to AWS resources. \n<br>\n<br>\n<b>Alert
Criteria</b><br>\nYellow: No IAM users have been created for this account.\n<br>
\n<br>\n<b>Recommended Action</b><br>\nCreate one or more IAM users and groups in
your account. You can then create additional users whose permissions are limited
to perform specific tasks in your AWS environment. For more information, see <a
href=\"https://docs.aws.amazon.com/IAM/latest/UserGuide/IAMGettingStarted.html\"
target=\"_blank\">Getting Started</a>. \n<br><br>\n<b>Additional Resources</b><br>
\n<a href=\"https://docs.aws.amazon.com/IAM/latest/UserGuide/IAM_Introduction.html\"
target=\"_blank\">What Is IAM?</a>",
```



```
        "category": "security",
        "metadata": []
      }
    ]
  }
```

Untuk informasi selengkapnya, lihat [AWS Trusted Advisor](#) di Panduan Pengguna AWS Support.

- Untuk detail API, lihat [DescribeTrustedAdvisorChecks](#) di Referensi AWS CLI Perintah.

## refresh-trusted-advisor-check

Contoh kode berikut menunjukkan cara menggunakan `refresh-trusted-advisor-check`.

### AWS CLI

Untuk me-refresh pemeriksaan AWS Trusted Advisor

`refresh-trusted-advisor-check` Contoh berikut menyegarkan cek Trusted Advisor Amazon S3 Bucket Permissions di akun Anda. AWS

```
aws support refresh-trusted-advisor-check \
  --check-id "Pfx0RwqBli"
```

Output:

```
{
  "status": {
    "checkId": "Pfx0RwqBli",
    "status": "enqueued",
    "millisUntilNextRefreshable": 3599992
  }
}
```

Untuk informasi selengkapnya, lihat [AWS Trusted Advisor](#) di Panduan Pengguna AWS Support.

- Untuk detail API, lihat [RefreshTrustedAdvisorCheck](#) di Referensi AWS CLI Perintah.

## resolve-case

Contoh kode berikut menunjukkan cara menggunakan `resolve-case`.

## AWS CLI

Untuk menyelesaikan kasus dukungan

`resolve-case` Contoh berikut menyelesaikan kasus dukungan di akun Anda AWS .

```
aws support resolve-case \  
  --case-id "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Output:

```
{  
  "finalCaseStatus": "resolved",  
  "initialCaseStatus": "work-in-progress"  
}
```

Untuk informasi selengkapnya, lihat [Manajemen kasus](#) di AWS Support User Guide.

- Untuk detail API, lihat [ResolveCase](#) di Referensi AWS CLI Perintah.

## Contoh Amazon SWF menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan AWS Command Line Interface With Amazon SWF.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

## Tindakan

### **count-closed-workflow-executions**

Contoh kode berikut menunjukkan cara menggunakan `count-closed-workflow-executions`.

#### AWS CLI

##### Menghitung Eksekusi Alur Kerja Tertutup

Anda dapat menggunakan `swf count-closed-workflow-executions` untuk mengambil jumlah eksekusi alur kerja tertutup untuk domain tertentu. Anda dapat menentukan filter untuk menghitung kelas eksekusi tertentu.

Itu `--domain` dan salah satu `--close-time-filter` atau `--start-time-filter` argumen diperlukan. Semua argumen lainnya adalah opsional.

```
aws swf count-closed-workflow-executions \
  --domain DataFrobtzz \
  --close-time-filter "{ \"latestDate\" : 1377129600, \"oldestDate\" :
  1370044800 }"
```

##### Output:

```
{
  "count": 2,
  "truncated": false
}
```

Jika “terpotong” adalah `true`, maka “hitung” mewakili jumlah maksimum yang dapat dikembalikan oleh Amazon SWF. Hasil lebih lanjut terpotong.

Untuk mengurangi jumlah hasil yang dikembalikan, Anda dapat:

memodifikasi `--close-time-filter` atau `--start-time-filter` nilai untuk mempersempit rentang waktu yang dicari. Masing-masing dari ini saling eksklusif: Anda hanya dapat menentukan salah satunya dalam request. Use the `--close-status-filter`, `--execution-filter`, `--tag-filter` atau `--type-filter` argumen untuk memfilter hasil lebih lanjut. Namun, argumen ini juga saling eksklusif.

Lihat Juga [CountClosedWorkflowExecutions](#) di Referensi API Layanan Alur Kerja Sederhana Amazon

- Untuk detail API, lihat [CountClosedWorkflowExecutions](#) di Referensi AWS CLI Perintah.

## count-open-workflow-executions

Contoh kode berikut menunjukkan cara menggunakan `count-open-workflow-executions`.

### AWS CLI

#### Menghitung Eksekusi Alur Kerja Terbuka

Anda dapat menggunakan `swf count-open-workflow-executions` untuk mengambil jumlah eksekusi alur kerja terbuka untuk domain tertentu. Anda dapat menentukan filter untuk menghitung kelas eksekusi tertentu.

`--start-time-filter` Argumen `--domain` dan diperlukan. Semua argumen lainnya adalah opsional.

```
aws swf count-open-workflow-executions \  
  --domain DataFrobtzz \  
  --start-time-filter "{ \"latestDate\" : 1377129600, \"oldestDate\" :  
1370044800 }"
```

#### Output:

```
{  
  "count": 4,  
  "truncated": false  
}
```

Jika “terpotong” adalah `true`, maka “hitung” mewakili jumlah maksimum yang dapat dikembalikan oleh Amazon SWF. Hasil lebih lanjut terpotong.

Untuk mengurangi jumlah hasil yang dikembalikan, Anda dapat:

memodifikasi `--start-time-filter` nilai untuk mempersempit rentang waktu yang dicari. Gunakan `--close-status-filter`, `--execution-filter`, `--tag-filter` atau `--type-filter` argumen untuk memfilter hasil lebih lanjut. Masing-masing saling eksklusif: Anda hanya dapat menentukan salah satunya dalam permintaan.

Untuk informasi selengkapnya, lihat `CountOpenWorkflowExecutions` di Referensi API Amazon Simple Workflow Service

- Untuk detail API, lihat [CountOpenWorkflowExecutions](#) di Referensi AWS CLI Perintah.

## deprecate-domain

Contoh kode berikut menunjukkan cara menggunakan `deprecate-domain`.

### AWS CLI

#### Mengakhiri Domain

Untuk menghentikan domain (Anda masih dapat melihatnya, tetapi tidak dapat membuat eksekusi alur kerja baru atau mendaftarkan jenis di dalamnya), gunakan `swf deprecate-domain`. Ini memiliki parameter tunggal yang diperlukan `--name`, yang mengambil nama domain untuk tidak digunakan lagi.

```
aws swf deprecate-domain \  
  --name MyNeatNewDomain ""
```

Seperti halnya `register-domain`, tidak ada output yang dikembalikan. Namun, jika Anda menggunakan `list-domains` untuk melihat domain terdaftar, Anda akan melihat bahwa domain tersebut telah usang dan tidak lagi muncul di data yang dikembalikan.

```
aws swf list-domains \  
  --registration-status REGISTERED  
  {  
    "domainInfos": [  
      {  
        "status": "REGISTERED",  
        "name": "DataFrobotz"  
      },  
      {  
        "status": "REGISTERED",  
        "name": "erontest"  
      }  
    ]  
  }
```

Jika Anda menggunakan `--registration-status DEPRECATED` dengan `list-domains`, Anda akan melihat domain Anda yang tidak digunakan lagi.

```
aws swf list-domains \  
  --registration-status DEPRECATED
```

```
--registration-status DEPRECATED
{
  "domainInfos": [
    {
      "status": "DEPRECATED",
      "name": "MyNeatNewDomain"
    }
  ]
}
```

Anda masih dapat menggunakan `describe-domain` untuk mendapatkan informasi tentang domain usang.

```
aws swf describe-domain \
  --name MyNeatNewDomain
{
  "domainInfo": {
    "status": "DEPRECATED",
    "name": "MyNeatNewDomain"
  },
  "configuration": {
    "workflowExecutionRetentionPeriodInDays": "0"
  }
}
```

Lihat Juga [DeprecateDomain](#) di Referensi API Layanan Alur Kerja Sederhana Amazon

- Untuk detail API, lihat [DeprecateDomain](#) di Referensi AWS CLI Perintah.

## **describe-domain**

Contoh kode berikut menunjukkan cara menggunakan `describe-domain`.

### AWS CLI

#### Mendapatkan Informasi Tentang Domain

Untuk mendapatkan informasi terperinci tentang domain tertentu, gunakan `swf describe-domain` perintah. Ada satu parameter yang diperlukan: `--name`, yang mengambil nama domain yang Anda inginkan informasi tentang.

```
aws swf describe-domain \
```

```
--name DataFrobotz
{
  "domainInfo": {
    "status": "REGISTERED",
    "name": "DataFrobotz"
  },
  "configuration": {
    "workflowExecutionRetentionPeriodInDays": "1"
  }
}
```

Anda juga dapat menggunakan `describe-domain` untuk mendapatkan informasi tentang domain usang.

```
aws swf describe-domain \
  --name MyNeatNewDomain
{
  "domainInfo": {
    "status": "DEPRECATED",
    "name": "MyNeatNewDomain"
  },
  "configuration": {
    "workflowExecutionRetentionPeriodInDays": "0"
  }
}
```

Lihat Juga [DescribeDomain](#) di Referensi API Layanan Alur Kerja Sederhana Amazon

- Untuk detail API, lihat [DescribeDomain](#) di Referensi AWS CLI Perintah.

## list-activity-types

Contoh kode berikut menunjukkan cara menggunakan `list-activity-types`.

### AWS CLI

#### Jenis Kegiatan Daftar

Untuk mendapatkan daftar jenis aktivitas untuk domain, gunakan `swf list-activity-types`.  
`--registration-status` Argumen `--domain` dan diperlukan.

```
aws swf list-activity-types \
```

```
--domain DataFrobtzz \  
--registration-status REGISTERED
```

**Output:**

```
{  
  "typeInfos": [  
    {  
      "status": "REGISTERED",  
      "creationDate": 1371454150.451,  
      "activityType": {  
        "version": "1",  
        "name": "confirm-user-email"  
      },  
      "description": "subscribe confirm-user-email activity"  
    },  
    {  
      "status": "REGISTERED",  
      "creationDate": 1371454150.709,  
      "activityType": {  
        "version": "1",  
        "name": "confirm-user-phone"  
      },  
      "description": "subscribe confirm-user-phone activity"  
    },  
    {  
      "status": "REGISTERED",  
      "creationDate": 1371454149.871,  
      "activityType": {  
        "version": "1",  
        "name": "get-subscription-info"  
      },  
      "description": "subscribe get-subscription-info activity"  
    },  
    {  
      "status": "REGISTERED",  
      "creationDate": 1371454150.909,  
      "activityType": {  
        "version": "1",  
        "name": "send-subscription-success"  
      },  
      "description": "subscribe send-subscription-success activity"  
    }  
  ],  
}
```



```

    {
      "status": "REGISTERED",
      "creationDate": 1371454150.085,
      "activityType": {
        "version": "1",
        "name": "subscribe-user-sns"
      },
      "description": "subscribe subscribe-user-sns activity"
    }
  ]
}

```

Anda dapat menggunakan `--name` argumen untuk memilih hanya jenis aktivitas dengan nama tertentu:

```

aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
  --name "send-subscription-success"

```

Output:

```

{
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.909,
      "activityType": {
        "version": "1",
        "name": "send-subscription-success"
      },
      "description": "subscribe send-subscription-success activity"
    }
  ]
}

```

Untuk mengambil hasil di halaman, Anda dapat mengatur `--maximum-page-size` argumen. Jika lebih banyak hasil yang dikembalikan daripada yang akan muat di halaman hasil, `nextPageToken` "" akan dikembalikan dalam kumpulan hasil:

```

aws swf list-activity-types \
  --domain DataFrobtzz \

```

```
--registration-status REGISTERED \
--maximum-page-size 2
```

### Output:

```
{
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAAAA1Gp1BelJq
+PmHvAnDxJYbup8+0R4LVtbXLD17QNY7C30pHo9Ssz06D/GuFz10yC73umBQ1t0PJ/gC/
aYpzDMqUIWIA1T9W0s2DryyZX40C/6Lhk9/
o5kdsuWMSBkHhgaZjgwp3WJINIFJFdaSMxY2vYAX7AtRtpcqJuBDDRE9RaRqDGYqIYUM1tarki qpSY1ZVveBasBv1vyU
WGAaqehiDz7/JzLT/wWNNUM0d+Nhe",
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.451,
      "activityType": {
        "version": "1",
        "name": "confirm-user-email"
      },
      "description": "subscribe confirm-user-email activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.709,
      "activityType": {
        "version": "1",
        "name": "confirm-user-phone"
      },
      "description": "subscribe confirm-user-phone activity"
    }
  ]
}
```

Anda dapat meneruskan nextPageToken nilai ke panggilan berikutnya list-activity-types dalam --next-page-token argumen, mengambil halaman hasil berikutnya:

```
aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
  --maximum-page-size 2 \
  --next-page-token "AAAAKgAAAAEAAAAAAAAAAAAA1Gp1BelJq
+PmHvAnDxJYbup8+0R4LVtbXLD17QNY7C30pHo9Ssz06D/GuFz10yC73umBQ1t0PJ/gC/
aYpzDMqUIWIA1T9W0s2DryyZX40C/6Lhk9/
```

```
o5kdsuWMSBkHhgaZjgwp3WJINIFJFdaSMxY2vYAX7AtRtpcqJuBDDRE9RaRqDGYqIYUMltarki qpSY1ZVveBasBv1vyU
WGAaqehiDz7/JzLT/wWNUM0d+Nhe"
```

### Output:

```
{
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAAw+7LZ4GRZPzTqBHsp2wBxWB8m1sgLCc1gCuq3J+h/
m3+vOfFqtckjLwV5cc40jNAzTCuq/
XcylPumGwkjbajtpqZpbq0cVNfjFxGoi0LB201bvv0krbUISBvlpFPmSWpDSZJsxg5UxCcweteS1Fn1PNSZ/
MoinBZo80TkjMuzcsTuK0zH9wCaR8ITcALJ3SaqHU3pyIRS5hPmFA30LIc8zaAepj1aujo6hntNSCruB4"
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454149.871,
      "activityType": {
        "version": "1",
        "name": "get-subscription-info"
      },
      "description": "subscribe get-subscription-info activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.909,
      "activityType": {
        "version": "1",
        "name": "send-subscription-success"
      },
      "description": "subscribe send-subscription-success activity"
    }
  ]
}
```

Jika masih ada lebih banyak hasil untuk dikembalikan, `nextPageToken` "" akan dikembalikan dengan hasilnya. Ketika tidak ada lagi halaman hasil untuk dikembalikan, `nextPageToken` tidak akan dikembalikan dalam set hasil.

Anda dapat menggunakan `--reverse-order` argumen untuk membalikkan urutan hasil yang dikembalikan. Ini juga mempengaruhi hasil halaman.

```
aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
```

```
--maximum-page-size 2 \
--reverse-order
```

**Output:**

```
{
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAAwXcpu5ePSyQkrC
+8WMBmSrenuZC2ZkIXQYBPB/b9xIOVkj+bMEFhGj0KmmJ4rF7iddhjf7UMYCsfGkEn7mk
+yMCgVc1JxDWmB0EH46bhcmclmYNQihMDmUwocpr7To6/R7CLu0St1gkFayx0idJXErQW0zdNfQaIWAnF/
cwioBbX1kz1fQzmDeU3M5oYGMPQIrUqkPq7pMEW0q01K5eDN97NzFYdZZ/r1cLDWPZhUjY",
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.085,
      "activityType": {
        "version": "1",
        "name": "subscribe-user-sns"
      },
      "description": "subscribe subscribe-user-sns activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.909,
      "activityType": {
        "version": "1",
        "name": "send-subscription-success"
      },
      "description": "subscribe send-subscription-success activity"
    }
  ]
}
```

Lihat Juga [ListActivityTypes](#) di Referensi API Layanan Alur Kerja Sederhana Amazon

- Untuk detail API, lihat [ListActivityTypes](#) di Referensi AWS CLI Perintah.

**list-domains**

Contoh kode berikut menunjukkan cara menggunakan `list-domains`.

**AWS CLI**

Contoh 1: Untuk membuat daftar domain terdaftar

Contoh `list-domains` perintah berikut mencantumkan domain REGISTERED SWF yang telah Anda daftarkan untuk akun Anda.

```
aws swf list-domains \  
  --registration-status REGISTERED
```

Output:

```
{  
  "domainInfos": [  
    {  
      "status": "REGISTERED",  
      "name": "DataFrobotz"  
    },  
    {  
      "status": "REGISTERED",  
      "name": "erontest"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [ListDomains](#) di Referensi API Amazon Simple Workflow Service

Contoh 2: Untuk mencantumkan domain Anda yang tidak digunakan lagi

Contoh `list-domains` perintah berikut mencantumkan domain DEPRECATED SWF yang telah Anda daftarkan untuk akun Anda. Domain usang adalah domain yang tidak dapat mendaftarkan alur kerja atau aktivitas baru, tetapi masih dapat ditanyakan.

```
aws swf list-domains \  
  --registration-status DEPRECATED
```

Output:

```
{  
  "domainInfos": [  
    {  
      "status": "DEPRECATED",  
      "name": "MyNeatNewDomain"  
    }  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [ListDomains](#) di Referensi API Amazon Simple Workflow Service

Contoh 3: Untuk daftar halaman pertama domain terdaftar

Contoh `list-domains` perintah berikut mencantumkan domain REGISTERED SWF halaman pertama yang telah Anda daftarkan untuk akun Anda menggunakan opsi. `--maximum-page-size`

```
aws swf list-domains \
  --registration-status REGISTERED \
  --maximum-page-size 1
```

Output:

```
{
  "domainInfos": [
    {
      "status": "REGISTERED",
      "name": "DataFrobotz"
    }
  ],
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAA2QJKNtidVgd49TTeNwYcpD
+QKT2ynuEbibcQWe2QKrs1MGe63gpS0MgZGpcpoKttL40CXRFn98Xif557it
+wSZUsvUDtImjDLvguyuyyFdIZtvIxIKE0Pm3k2r40jAGaFsG0uVbrK1jv1a7wdU7FYH301kNCP8b7PBj9SBkUyGoiAg
}
```

Untuk informasi selengkapnya, lihat [ListDomains](#) di Referensi API Amazon Simple Workflow Service

Contoh 4: Untuk daftar satu halaman tertentu dari domain terdaftar

Contoh `list-domains` perintah berikut mencantumkan domain REGISTERED SWF halaman pertama yang telah Anda daftarkan untuk akun Anda menggunakan opsi. `--maximum-page-size`

Ketika Anda membuat panggilan lagi, kali ini memberikan nilai `nextPageToken` dalam `--next-page-token` argumen, Anda akan mendapatkan halaman lain dari hasil.

```
aws swf list-domains \
  --registration-status REGISTERED \
  --maximum-page-size 1 \
  --next-page-token "AAAAKgAAAAEAAAAAAAAA2QJKNtidVgd49TTeNwYcpD
+QKT2ynuEbibcQWe2QKrs1MGe63gpS0MgZGpcpoKttL40CXRFn98Xif557it
+wSZUsvUDtImjDLvguyuyyFdIZtvIxIKE0Pm3k2r40jAGaFsG0uVbrK1jv1a7wdU7FYH301kNCP8b7PBj9SBkUyGoiAg
```

Output:

```
{
  "domainInfos": [
    {
      "status": "REGISTERED",
      "name": "erontest"
    }
  ]
}
```

Ketika tidak ada halaman hasil lebih lanjut untuk diambil, tidak `nextPageToken` akan dikembalikan dalam hasil.

Untuk informasi selengkapnya, lihat [ListDomains](#) di Referensi API Amazon Simple Workflow Service

- Untuk detail API, lihat [ListDomains](#) di Referensi AWS CLI Perintah.

## list-workflow-types

Contoh kode berikut menunjukkan cara menggunakan `list-workflow-types`.

### AWS CLI

#### Daftar Jenis Alur Kerja

Untuk mendapatkan daftar jenis alur kerja untuk domain, gunakan `swf list-workflow-types`. `--registration-status` Argumen `--domain` dan diperlukan. Berikut adalah contoh sederhana.

```
aws swf list-workflow-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED
```

**Output:**

```
{
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454149.598,
      "description": "DataFrobtzz subscribe workflow",
      "workflowType": {
        "version": "v3",
        "name": "subscribe"
      }
    }
  ]
}
```

Seperti halnya `list-activity-types`, Anda dapat menggunakan `--name` argumen untuk memilih hanya jenis alur kerja dengan nama tertentu, dan menggunakan `--maximum-page-size` argumen dalam koordinasi dengan `--next-page-token` hasil halaman. Untuk membalikkan urutan pengembalian hasil, gunakan `--reverse-order`.

Lihat Juga [ListWorkflowTypes](#) di Referensi API Layanan Alur Kerja Sederhana Amazon

- Untuk detail API, lihat [ListWorkflowTypes](#) di Referensi AWS CLI Perintah.

**register-domain**

Contoh kode berikut menunjukkan cara menggunakan `register-domain`.

**AWS CLI****Mendaftarkan Domain**

Anda dapat menggunakan AWS CLI untuk mendaftarkan domain baru. Gunakan perintah `swf register-domain`. < [https://aws.amazon.com/swf/faqs/#retain\\_limit](https://aws.amazon.com/swf/faqs/#retain_limit) > Ada dua parameter yang diperlukan `--name`, yang mengambil nama domain, dan `--workflow-execution-retention-period-in-days`, yang membutuhkan bilangan bulat untuk menentukan jumlah hari untuk menyimpan data eksekusi alur kerja pada domain ini, hingga periode maksimum 90 hari (untuk informasi lebih lanjut, lihat FAQ SWF). Data eksekusi alur kerja tidak akan disimpan setelah jumlah hari yang ditentukan telah berlalu.

```
aws swf register-domain \
```



```
--name MyNeatNewDomain \  
--workflow-execution-retention-period-in-days 0  
""
```

Ketika Anda mendaftarkan domain, tidak ada yang dikembalikan (""), tetapi Anda dapat swf describe-domain menggunakan swf list-domains atau melihat domain baru.

```
aws swf list-domains \  
--registration-status REGISTERED  
{  
  "domainInfos": [  
    {  
      "status": "REGISTERED",  
      "name": "DataFrobotz"  
    },  
    {  
      "status": "REGISTERED",  
      "name": "MyNeatNewDomain"  
    },  
    {  
      "status": "REGISTERED",  
      "name": "erontest"  
    }  
  ]  
}
```

Menggunakan swf describe-domain :

```
aws swf describe-domain --name MyNeatNewDomain  
{  
  "domainInfo": {  
    "status": "REGISTERED",  
    "name": "MyNeatNewDomain"  
  },  
  "configuration": {  
    "workflowExecutionRetentionPeriodInDays": "0"  
  }  
}
```

Lihat Juga [RegisterDomain](#) di Referensi API Layanan Alur Kerja Sederhana Amazon

- Untuk detail API, lihat [RegisterDomain](#) di Referensi AWS CLI Perintah.

## register-workflow-type

Contoh kode berikut menunjukkan cara menggunakan `register-workflow-type`.

### AWS CLI

#### Mendaftarkan Jenis Alur Kerja

Untuk mendaftarkan tipe Workflow dengan AWS CLI, gunakan `swf register-workflow-type` perintah.

```
aws swf register-workflow-type \  
  --domain DataFrobtzz \  
  --name "MySimpleWorkflow" \  
  --workflow-version "v1"
```

Jika berhasil, perintah tidak menghasilkan output.

Pada kesalahan (misalnya, jika Anda mencoba mendaftarkan tipe alur kerja yang sama, atau menentukan domain yang tidak ada), Anda akan mendapatkan respons di JSON.

```
{  
  "message": "WorkflowType=[name=MySimpleWorkflow, version=v1]",  
  "__type": "com.amazonaws.swf.base.model#TypeAlreadyExistsFault"  
}
```

Itu `--domain`, `--name` dan `--workflow-version` diperlukan. Anda juga dapat mengatur deskripsi alur kerja, batas waktu, dan kebijakan alur kerja anak.

Untuk informasi selengkapnya, lihat [RegisterWorkflowType](#) di Referensi API Amazon Simple Workflow Service

- Untuk detail API, lihat [RegisterWorkflowType](#) di Referensi AWS CLI Perintah.

## Contoh Systems Manager menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Systems Manager.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **add-tags-to-resource**

Contoh kode berikut menunjukkan cara menggunakan `add-tags-to-resource`.

AWS CLI

Contoh 1: Untuk menambahkan tag ke jendela pemeliharaan

`add-tags-to-resource` Contoh berikut menambahkan tag ke jendela pemeliharaan yang ditentukan.

```
aws ssm add-tags-to-resource \  
  --resource-type "MaintenanceWindow" \  
  --resource-id "mw-03eb9db428EXAMPLE" \  
  --tags "Key=Stack,Value=Production"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk menambahkan tag ke parameter

`add-tags-to-resource` Contoh berikut menambahkan dua tag ke parameter yang ditentukan.

```
aws ssm add-tags-to-resource \  
  --resource-type "Parameter" \  
  --resource-id "My-Parameter" \  
  --tags '[{"Key":"Region","Value":"East"}, {"Key":"Environment",  
  "Value":"Production"}]'
```

Perintah ini tidak menghasilkan output.

Contoh 3: Untuk menambahkan tag ke dokumen SSM

`add-tags-to-resource` Contoh berikut menambahkan tag ke dokumen yang ditentukan.

```
aws ssm add-tags-to-resource \  
  --resource-type "Document" \  
  --resource-id "My-Document" \  
  --tags "Key=Quarter,Value=Q322"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai sumber daya Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [AddTagsToResource](#) di Referensi AWS CLI Perintah.

## **associate-ops-item-related-item**

Contoh kode berikut menunjukkan cara menggunakan `associate-ops-item-related-item`.

### AWS CLI

Untuk mengaitkan item terkait

`associate-ops-item-related-item` Contoh berikut mengaitkan item terkait dengan OpsItem

```
aws ssm associate-ops-item-related-item \  
  --ops-item-id "oi-649fExample" \  
  --association-type "RelatesTo" \  
  --resource-type "AWS::SSMIncidents::IncidentRecord" \  
  --resource-uri "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/c2bde883-f7d5-343a-b13a-bf5fe9ea689f"
```

Output:

```
{  
  "AssociationId": "61d7178d-a30d-4bc5-9b4e-a9e74EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan insiden Manajer Insiden OpsCenter di](#) Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [AssociateOpsItemRelatedItem](#) di Referensi AWS CLI Perintah.

## cancel-command

Contoh kode berikut menunjukkan cara menggunakan `cancel-command`.

### AWS CLI

Contoh 1: Untuk membatalkan perintah untuk semua instance

`cancel-command` Contoh berikut mencoba untuk membatalkan perintah tertentu yang sudah berjalan untuk semua instance.

```
aws ssm cancel-command \  
  --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
```

Perintah ini tidak menghasilkan output.

Contoh 2: Untuk membatalkan perintah untuk instance tertentu

`cancel-command` Contoh berikut mencoba untuk membatalkan perintah untuk instance tertentu saja.

```
aws ssm cancel-command \  
  --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE" \  
  --instance-ids "i-02573cafcfEXAMPLE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Parameter Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CancelCommand](#) di Referensi AWS CLI Perintah.

## cancel-maintenance-window-execution

Contoh kode berikut menunjukkan cara menggunakan `cancel-maintenance-window-execution`.

### AWS CLI

Untuk membatalkan eksekusi jendela pemeliharaan

`cancel-maintenance-window-execution` Contoh ini menghentikan eksekusi jendela pemeliharaan yang ditentukan yang sudah berlangsung.

```
aws ssm cancel-maintenance-window-execution \  
  --window-execution-id j218d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE
```

Output:

```
{  
  "WindowExecutionId": "j218d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Systems Manager Maintenance Windows Tutorial \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CancelMaintenanceWindowExecution](#) di Referensi AWS CLI Perintah.

## create-activation

Contoh kode berikut menunjukkan cara menggunakan `create-activation`.

AWS CLI

Untuk membuat aktivasi instance terkelola

`create-activation` Contoh berikut menciptakan aktivasi instance terkelola.

```
aws ssm create-activation \  
  --default-instance-name "HybridWebServers" \  
  --iam-role "HybridWebServersRole" \  
  --registration-limit 5
```

Output:

```
{  
  "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",  
  "ActivationCode": "dRmgnYaFv567vEXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat Aktivasi Instans Terkelola untuk Lingkungan Hybrid](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CreateActivation](#) di Referensi AWS CLI Perintah.

## create-association-batch

Contoh kode berikut menunjukkan cara menggunakan `create-association-batch`.

### AWS CLI

Untuk membuat beberapa asosiasi

Contoh ini mengaitkan dokumen konfigurasi dengan beberapa instance. Output mengembalikan daftar operasi yang berhasil dan gagal, jika berlaku.

Perintah:

```
aws ssm create-association-batch --entries "Name=AWS-UpdateSSMAgent,InstanceId=i-1234567890abcdef0" "Name=AWS-UpdateSSMAgent,InstanceId=i-9876543210abcdef0"
```

Output:

```
{
  "Successful": [
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-1234567890abcdef0",
      "AssociationVersion": "1",
      "Date": 1550504725.007,
      "LastUpdateAssociationDate": 1550504725.007,
      "Status": {
        "Date": 1550504725.007,
        "Name": "Associated",
        "Message": "Associated with AWS-UpdateSSMAgent"
      },
      "Overview": {
        "Status": "Pending",
        "DetailedStatus": "Creating"
      },
      "DocumentVersion": "$DEFAULT",
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "Targets": [
        {
          "Key": "InstanceIds",
          "Values": [
            "i-1234567890abcdef0"
          ]
        }
      ]
    }
  ]
}
```

```

    ]
  }
]
},
{
  "Name": "AWS-UpdateSSMAgent",
  "InstanceId": "i-9876543210abcdef0",
  "AssociationVersion": "1",
  "Date": 1550504725.057,
  "LastUpdateAssociationDate": 1550504725.057,
  "Status": {
    "Date": 1550504725.057,
    "Name": "Associated",
    "Message": "Associated with AWS-UpdateSSMAgent"
  },
  "Overview": {
    "Status": "Pending",
    "DetailedStatus": "Creating"
  },
  "DocumentVersion": "$DEFAULT",
  "AssociationId": "9c9f7f20-5154-4fed-a83e-0123456789ab",
  "Targets": [
    {
      "Key": "InstanceIds",
      "Values": [
        "i-9876543210abcdef0"
      ]
    }
  ]
}
],
"Failed": []
}

```

- Untuk detail API, lihat [CreateAssociationBatch](#) di Referensi AWS CLI Perintah.

## create-association

Contoh kode berikut menunjukkan cara menggunakan `create-association`.

### AWS CLI

Contoh 1: Untuk mengaitkan dokumen menggunakan ID instance



Contoh ini mengaitkan dokumen konfigurasi dengan instance, menggunakan ID instance.

```
aws ssm create-association \  
  --instance-id "i-0cb2b964d3e14fd9f" \  
  --name "AWS-UpdateSSMAgent"
```

Output:

```
{  
  "AssociationDescription": {  
    "Status": {  
      "Date": 1487875500.33,  
      "Message": "Associated with AWS-UpdateSSMAgent",  
      "Name": "Associated"  
    },  
    "Name": "AWS-UpdateSSMAgent",  
    "InstanceId": "i-0cb2b964d3e14fd9f",  
    "Overview": {  
      "Status": "Pending",  
      "DetailedStatus": "Creating"  
    },  
    "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",  
    "DocumentVersion": "$DEFAULT",  
    "LastUpdateAssociationDate": 1487875500.33,  
    "Date": 1487875500.33,  
    "Targets": [  
      {  
        "Values": [  
          "i-0cb2b964d3e14fd9f"  
        ],  
        "Key": "InstanceIds"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [CreateAssociation](#) di Referensi API AWS Systems Manager.

Contoh 2: Untuk mengaitkan dokumen menggunakan target

Contoh ini mengaitkan dokumen konfigurasi dengan instance, menggunakan target.

```
aws ssm create-association \  
  --name "AWS-UpdateSSMAgent"
```

```
--name "AWS-UpdateSSMAgent" \  
--targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f"
```

### Output:

```
{  
  "AssociationDescription": {  
    "Status": {  
      "Date": 1487875500.33,  
      "Message": "Associated with AWS-UpdateSSMAgent",  
      "Name": "Associated"  
    },  
    "Name": "AWS-UpdateSSMAgent",  
    "InstanceId": "i-0cb2b964d3e14fd9f",  
    "Overview": {  
      "Status": "Pending",  
      "DetailedStatus": "Creating"  
    },  
    "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",  
    "DocumentVersion": "$DEFAULT",  
    "LastUpdateAssociationDate": 1487875500.33,  
    "Date": 1487875500.33,  
    "Targets": [  
      {  
        "Values": [  
          "i-0cb2b964d3e14fd9f"  
        ],  
        "Key": "InstanceIds"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [CreateAssociation](#) di Referensi API AWS Systems Manager.

### Contoh 3: Untuk membuat asosiasi yang berjalan hanya sekali

Contoh ini menciptakan asosiasi baru yang hanya berjalan sekali pada tanggal dan waktu yang ditentukan. Asosiasi yang dibuat dengan tanggal di masa lalu atau sekarang (pada saat diproses tanggal di masa lalu) segera berjalan.

```
aws ssm create-association \  

```

```
--name "AWS-UpdateSSMAgent" \  
--targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \  
--schedule-expression "at(2020-05-14T15:55:00)" \  
--apply-only-at-cron-interval
```

### Output:

```
{  
  "AssociationDescription": {  
    "Status": {  
      "Date": 1487875500.33,  
      "Message": "Associated with AWS-UpdateSSMAgent",  
      "Name": "Associated"  
    },  
    "Name": "AWS-UpdateSSMAgent",  
    "InstanceId": "i-0cb2b964d3e14fd9f",  
    "Overview": {  
      "Status": "Pending",  
      "DetailedStatus": "Creating"  
    },  
    "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",  
    "DocumentVersion": "$DEFAULT",  
    "LastUpdateAssociationDate": 1487875500.33,  
    "Date": 1487875500.33,  
    "Targets": [  
      {  
        "Values": [  
          "i-0cb2b964d3e14fd9f"  
        ],  
        "Key": "InstanceIds"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [CreateAssociation](#) di Referensi atau Referensi API AWS Systems Manager: [Cron dan ekspresi rate untuk Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CreateAssociation](#) di Referensi AWS CLI Perintah.

## create-document

Contoh kode berikut menunjukkan cara menggunakan `create-document`.

### AWS CLI

Untuk membuat dokumen

`create-document` Contoh berikut membuat dokumen Systems Manager.

```
aws ssm create-document \  
  --content file://exampleDocument.yml \  
  --name "Example" \  
  --document-type "Automation" \  
  --document-format YAML
```

Output:

```
{  
  "DocumentDescription": {  
    "Hash": "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",  
    "HashType": "Sha256",  
    "Name": "Example",  
    "Owner": "29884EXAMPLE",  
    "CreateDate": 1583256349.452,  
    "Status": "Creating",  
    "DocumentVersion": "1",  
    "Description": "Document Example",  
    "Parameters": [  
      {  
        "Name": "AutomationAssumeRole",  
        "Type": "String",  
        "Description": "(Required) The ARN of the role that allows  
Automation to perform the actions on your behalf. If no role is specified, Systems  
Manager Automation uses your IAM permissions to execute this document.",  
        "DefaultValue": ""  
      },  
      {  
        "Name": "InstanceId",  
        "Type": "String",  
        "Description": "(Required) The ID of the Amazon EC2 instance.",  
        "DefaultValue": ""  
      }  
    ],  
  },  
}
```

```
    "PlatformTypes": [
      "Windows",
      "Linux"
    ],
    "DocumentType": "Automation",
    "SchemaVersion": "0.3",
    "LatestVersion": "1",
    "DefaultVersion": "1",
    "DocumentFormat": "YAML",
    "Tags": []
  }
}
```

Untuk informasi selengkapnya, lihat [Membuat Dokumen Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CreateDocument](#) di Referensi AWS CLI Perintah.

## **create-maintenance-window**

Contoh kode berikut menunjukkan cara menggunakan `create-maintenance-window`.

### AWS CLI

Contoh 1: Untuk membuat jendela pemeliharaan

`create-maintenance-window` Contoh berikut membuat jendela pemeliharaan baru yang setiap lima menit hingga dua jam (sesuai kebutuhan), mencegah tugas baru dimulai dalam waktu satu jam setelah akhir eksekusi jendela pemeliharaan, memungkinkan target yang tidak terkait (contoh yang belum Anda daftarkan dengan jendela pemeliharaan), dan menunjukkan melalui penggunaan tag khusus bahwa pembuatnya bermaksud menggunakannya dalam tutorial.

```
aws ssm create-maintenance-window \
  --name "My-Tutorial-Maintenance-Window" \
  --schedule "rate(5 minutes)" \
  --duration 2 --cutoff 1 \
  --allow-unassociated-targets \
  --tags "Key=Purpose,Value=Tutorial"
```

Output:

```
{
```

```
"WindowId": "mw-0c50858d01EXAMPLE"
}
```

Contoh 2: Untuk membuat jendela pemeliharaan yang berjalan hanya sekali

`create-maintenance-window` Contoh berikut membuat jendela pemeliharaan baru yang hanya berjalan satu kali pada tanggal dan waktu yang ditentukan.

```
aws ssm create-maintenance-window \
  --name My-One-Time-Maintenance-Window \
  --schedule "at(2020-05-14T15:55:00)" \
  --duration 5 \
  --cutoff 2 \
  --allow-unassociated-targets \
  --tags "Key=Environment,Value=Production"
```

Output:

```
{
  "WindowId": "mw-01234567890abcdef"
}
```

Untuk informasi selengkapnya, lihat [Pemeliharaan Windows](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CreateMaintenanceWindow](#) di Referensi AWS CLI Perintah.

## **create-ops-item**

Contoh kode berikut menunjukkan cara menggunakan `create-ops-item`.

### AWS CLI

Untuk membuat OpsItems

`create-ops-item` Contoh berikut menggunakan kunci `/aws/resources/OperationalData` untuk membuat OpsItem dengan sumber daya terkait Amazon DynamoDB.

```
aws ssm create-ops-item \
  --title "EC2 instance disk full" \
  --description "Log clean up may have failed which caused the disk to be full" \
  --priority 2 \
```

```

--source ec2 \
--operational-data '{"aws/resources":{"Value":["arn
\":"arn:aws:dynamodb:us-west-2:12345678:table/OpsItems
\}]}","Type":"SearchableString"}' \
--notifications Arn="arn:aws:sns:us-west-2:12345678:TestUser"

```

Output:

```

{
  "OpsItemId": "oi-1a2b3c4d5e6f"
}

```

Untuk informasi selengkapnya, lihat [Membuat OpsItems](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CreateOpsItem](#) di Referensi AWS CLI Perintah.

## create-patch-baseline

Contoh kode berikut menunjukkan cara menggunakan `create-patch-baseline`.

### AWS CLI

Contoh 1: Untuk membuat baseline patch dengan persetujuan otomatis

`create-patch-baseline` Contoh berikut membuat patch baseline untuk Windows Server yang menyetujui patch untuk lingkungan produksi tujuh hari setelah dirilis oleh Microsoft.

```

aws ssm create-patch-baseline \
  --name "Windows-Production-Baseline-AutoApproval" \
  --operating-system "WINDOWS" \
  --approval-rules
  "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Important
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},Approv
\
  --description "Baseline containing all updates approved for Windows Server
  production systems"

```

Output:

```

{
  "BaselineId": "pb-045f10b4f3EXAMPLE"
}

```

```
}

```

Contoh 2: Untuk membuat baseline patch dengan tanggal cutoff persetujuan

`create-patch-baseline` Contoh berikut membuat baseline patch untuk Windows Server yang menyetujui semua tambalan untuk lingkungan produksi yang dirilis pada atau sebelum 7 Juli 2020.

```
aws ssm create-patch-baseline \
  --name "Windows-Production-Baseline-AutoApproval" \
  --operating-system "WINDOWS" \
  --approval-rules
  "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Important
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},Approval
  \
  --description "Baseline containing all updates approved for Windows Server
  production systems"
```

Output:

```
{
  "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Contoh 3: Untuk membuat baseline patch dengan aturan persetujuan yang disimpan dalam file JSON

`create-patch-baseline` Contoh berikut membuat baseline patch untuk Amazon Linux 2017.09 yang menyetujui tambalan untuk lingkungan produksi tujuh hari setelah dirilis, menentukan aturan persetujuan untuk baseline patch, dan menentukan repositori kustom untuk tambalan.

```
aws ssm create-patch-baseline \
  --cli-input-json file://my-amazon-linux-approval-rules-and-repo.json
```

Isi dari `my-amazon-linux-approval-rules-and-repo.json`:

```
{
  "Name": "Amazon-Linux-2017.09-Production-Baseline",
  "Description": "My approval rules patch baseline for Amazon Linux 2017.09
instances",
  "OperatingSystem": "AMAZON_LINUX",
```



```
"Tags": [
  {
    "Key": "Environment",
    "Value": "Production"
  }
],
"ApprovalRules": {
  "PatchRules": [
    {
      "ApproveAfterDays": 7,
      "EnableNonSecurity": true,
      "PatchFilterGroup": {
        "PatchFilters": [
          {
            "Key": "SEVERITY",
            "Values": [
              "Important",
              "Critical"
            ]
          },
          {
            "Key": "CLASSIFICATION",
            "Values": [
              "Security",
              "Bugfix"
            ]
          },
          {
            "Key": "PRODUCT",
            "Values": [
              "AmazonLinux2017.09"
            ]
          }
        ]
      }
    }
  ]
},
"Sources": [
  {
    "Name": "My-AL2017.09",
    "Products": [
      "AmazonLinux2017.09"
    ]
  }
],
```

```

    "Configuration": "[amzn-main] \nname=amzn-main-Base
\nmirrorlist=http://repo./$awsregion./$awsdomain//$releasever/main/mirror.list //
\nmirrorlist_expire=300//\nmetadata_expire=300 \npriority=10 \nfailovermethod=priority
\nfastestmirror_enabled=0 \ngpgcheck=1 \ngpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-
KEY-amazon-ga \nenabled=1 \nretries=3 \ntimeout=5\nreport_instanceid=yes"
    }
  ]
}

```

Contoh 4: Untuk membuat baseline patch yang menentukan patch yang disetujui dan ditolak

`create-patch-baseline` Contoh berikut secara eksplisit menentukan patch untuk menyetujui dan menolak sebagai pengecualian untuk aturan persetujuan default.

```

aws ssm create-patch-baseline \
  --name "Amazon-Linux-2017.09-Alpha-Baseline" \
  --description "My custom approve/reject patch baseline for Amazon Linux 2017.09
instances" \
  --operating-system "AMAZON_LINUX" \
  --approved-patches "CVE-2018-1234567,example-pkg-EE-2018*.amzn1.noarch" \
  --approved-patches-compliance-level "HIGH" \
  --approved-patches-enable-non-security \
  --tags "Key=Environment,Value=Alpha"

```

Untuk informasi [selengkapnya, lihat Membuat Garis Dasar Patch Kustom](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [CreatePatchBaseline](#) di Referensi AWS CLI Perintah.

## **create-resource-data-sync**

Contoh kode berikut menunjukkan cara menggunakan `create-resource-data-sync`.

### AWS CLI

Untuk membuat sinkronisasi data sumber daya

Contoh ini membuat sinkronisasi data sumber daya. Tidak ada output jika perintah berhasil.

Perintah:

```
aws ssm create-resource-data-sync --sync-name "ssm-resource-data-sync" --s3-destination "BucketName=ssm-bucket,Prefix=inventory,SyncFormat=JsonSerDe,Region=us-east-1"
```

- Untuk detail API, lihat [CreateResourceDataSync](#) di Referensi AWS CLI Perintah.

## delete-activation

Contoh kode berikut menunjukkan cara menggunakan `delete-activation`.

### AWS CLI

Untuk menghapus aktivasi instance terkelola

`delete-activation` Contoh berikut menghapus aktivasi instance terkelola.

```
aws ssm delete-activation \  
  --activation-id "aa673477-d926-42c1-8757-1358cEXAMPLE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan AWS Systems Manager untuk Lingkungan Hybrid](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeleteActivation](#) di Referensi AWS CLI Perintah.

## delete-association

Contoh kode berikut menunjukkan cara menggunakan `delete-association`.

### AWS CLI

Contoh 1: Untuk menghapus asosiasi menggunakan ID asosiasi

`delete-association` Contoh berikut menghapus asosiasi untuk ID asosiasi tertentu. Tidak ada output jika perintah berhasil.

```
aws ssm delete-association \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengedit dan membuat versi baru asosiasi](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk menghapus asosiasi

`delete-association` Contoh berikut menghapus hubungan antara instance dan dokumen. Tidak ada output jika perintah berhasil.

```
aws ssm delete-association \  
  --instance-id "i-1234567890abcdef0" \  
  --name "AWS-UpdateSSMAgent"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan asosiasi di Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeleteAssociation](#) di Referensi AWS CLI Perintah.

## **delete-document**

Contoh kode berikut menunjukkan cara menggunakan `delete-document`.

### AWS CLI

Untuk menghapus dokumen

`delete-document` Contoh berikut menghapus dokumen Systems Manager.

```
aws ssm delete-document \  
  --name "Example"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membuat Dokumen Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeleteDocument](#) di Referensi AWS CLI Perintah.

## delete-inventory

Contoh kode berikut menunjukkan cara menggunakan `delete-inventory`.

### AWS CLI

Untuk menghapus jenis inventaris kustom

Contoh ini menghapus skema inventaris kustom.

Perintah:

```
aws ssm delete-inventory --type-name "Custom:RackInfo" --schema-delete-option "DeleteSchema"
```

Output:

```
{
  "DeletionId": "d72ac9e8-1f60-4d40-b1c6-bf8c78c68c4d",
  "TypeName": "Custom:RackInfo",
  "DeletionSummary": {
    "TotalCount": 1,
    "RemainingCount": 1,
    "SummaryItems": [
      {
        "Version": "1.0",
        "Count": 1,
        "RemainingCount": 1
      }
    ]
  }
}
```

Untuk menonaktifkan jenis inventaris kustom

Contoh ini menonaktifkan skema inventaris kustom.

Perintah:

```
aws ssm delete-inventory --type-name "Custom:RackInfo" --schema-delete-option "DisableSchema"
```

**Output:**

```
{
  "DeletionId": "6961492a-8163-44ec-aa1e-923364dd0850",
  "TypeName": "Custom:RackInformation",
  "DeletionSummary": {
    "TotalCount": 0,
    "RemainingCount": 0,
    "SummaryItems": []
  }
}
```

- Untuk detail API, lihat [DeleteInventory](#) di Referensi AWS CLI Perintah.

**delete-maintenance-window**

Contoh kode berikut menunjukkan cara menggunakan `delete-maintenance-window`.

**AWS CLI**

Untuk menghapus jendela pemeliharaan

`delete-maintenance-window` Contoh ini menghapus jendela pemeliharaan yang ditentukan.

```
aws ssm delete-maintenance-window \
  --window-id "mw-1a2b3c4d5e6f7g8h9"
```

**Output:**

```
{
  "WindowId": "mw-1a2b3c4d5e6f7g8h9"
}
```

Untuk informasi selengkapnya, lihat [Menghapus Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeleteMaintenanceWindow](#) di Referensi AWS CLI Perintah.

**delete-parameter**

Contoh kode berikut menunjukkan cara menggunakan `delete-parameter`.

## AWS CLI

Untuk menghapus parameter

`delete-parameter` Contoh berikut menghapus parameter tunggal yang ditentukan.

```
aws ssm delete-parameter \  
  --name "MyParameter"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeleteParameter](#) di Referensi AWS CLI Perintah.

## `delete-parameters`

Contoh kode berikut menunjukkan cara menggunakan `delete-parameters`.

## AWS CLI

Untuk menghapus daftar parameter

`delete-parameters` Contoh berikut menghapus parameter yang ditentukan.

```
aws ssm delete-parameters \  
  --names "MyFirstParameter" "MySecondParameter" "MyInvalidParameterName"
```

Output:

```
{  
  "DeletedParameters": [  
    "MyFirstParameter",  
    "MySecondParameter"  
  ],  
  "InvalidParameters": [  
    "MyInvalidParameterName"  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeleteParameters](#) di Referensi AWS CLI Perintah.

## **delete-patch-baseline**

Contoh kode berikut menunjukkan cara menggunakan `delete-patch-baseline`.

### AWS CLI

Untuk menghapus baseline patch

`delete-patch-baseline` Contoh berikut menghapus baseline patch yang ditentukan.

```
aws ssm delete-patch-baseline \  
  --baseline-id "pb-045f10b4f382baeda"
```

Output:

```
{  
  "BaselineId": "pb-045f10b4f382baeda"  
}
```

Untuk informasi [selengkapnya](#), lihat [Memperbarui atau Menghapus Garis Dasar Patch \(Konsol\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeletePatchBaseline](#) di Referensi AWS CLI Perintah.

## **delete-resource-data-sync**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource-data-sync`.

### AWS CLI

Untuk menghapus sinkronisasi data sumber daya

Contoh ini menghapus sinkronisasi data sumber daya. Tidak ada output jika perintah berhasil.

Perintah:

```
aws ssm delete-resource-data-sync --sync-name "ssm-resource-data-sync"
```

- Untuk detail API, lihat [DeleteResourceDataSync](#) di Referensi AWS CLI Perintah.



## deregister-managed-instance

Contoh kode berikut menunjukkan cara menggunakan `deregister-managed-instance`.

### AWS CLI

Untuk membatalkan pendaftaran instance terkelola

`deregister-managed-instance` Contoh berikut deregisters instance terkelola tertentu.

```
aws ssm deregister-managed-instance
  --instance-id "mi-08ab247cdfEXAMPLE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menderegistrasi Instans Terkelola di Lingkungan Hybrid di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [DeregisterManagedInstance](#) di Referensi AWS CLI Perintah.

## deregister-patch-baseline-for-patch-group

Contoh kode berikut menunjukkan cara menggunakan `deregister-patch-baseline-for-patch-group`.

### AWS CLI

Untuk membatalkan pendaftaran grup patch dari baseline patch

`deregister-patch-baseline-for-patch-group` Contoh berikut membatalkan pendaftaran grup patch yang ditentukan dari baseline patch yang ditentukan.

```
aws ssm deregister-patch-baseline-for-patch-group \
  --patch-group "Production" \
  --baseline-id "pb-0ca44a362fEXAMPLE"
```

Output:

```
{
  "PatchGroup": "Production",
  "BaselineId": "pb-0ca44a362fEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Menambahkan Grup Patch ke Garis Dasar Patch](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeregisterPatchBaselineForPatchGroup](#) di Referensi AWS CLI Perintah.

## **deregister-target-from-maintenance-window**

Contoh kode berikut menunjukkan cara menggunakan `deregister-target-from-maintenance-window`.

### AWS CLI

Untuk menghapus target dari jendela pemeliharaan

`deregister-target-from-maintenance-window` Contoh berikut menghapus target yang ditentukan dari jendela pemeliharaan yang ditentukan.

```
aws ssm deregister-target-from-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --window-target-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
```

Output:

```
{  
  "WindowId": "mw-ab12cd34ef56gh78",  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeregisterTargetFromMaintenanceWindow](#) di Referensi AWS CLI Perintah.

## **deregister-task-from-maintenance-window**

Contoh kode berikut menunjukkan cara menggunakan `deregister-task-from-maintenance-window`.

### AWS CLI

Untuk menghapus tugas dari jendela pemeliharaan

`deregister-task-from-maintenance-window` Contoh berikut menghapus tugas yang ditentukan dari jendela pemeliharaan yang ditentukan.

```
aws ssm deregister-task-from-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --window-task-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c"
```

Output:

```
{  
  "WindowTaskId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c",  
  "WindowId": "mw-ab12cd34ef56gh78"  
}
```

Untuk informasi selengkapnya, lihat [Systems Manager Maintenance Windows Tutorial \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DeregisterTaskFromMaintenanceWindow](#) di Referensi AWS CLI Perintah.

## **describe-activations**

Contoh kode berikut menunjukkan cara menggunakan `describe-activations`.

### AWS CLI

Untuk menggambarkan aktivasi

`describe-activations` Contoh berikut mencantumkan detail tentang aktivasi di AWS akun Anda.

```
aws ssm describe-activations
```

Output:

```
{  
  "ActivationList": [  
    {  
      "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",  
      "Description": "Example1",  
      "IamRole": "HybridWebServersRole",  
      "RegistrationLimit": 5,  
      "RegistrationsCount": 5,  
    }  
  ]  
}
```

```

    "ExpirationDate": 1584316800.0,
    "Expired": false,
    "CreateDate": 1581954699.792
  },
  {
    "ActivationId": "3ee0322b-f62d-40eb-b672-13ebfEXAMPLE",
    "Description": "Example2",
    "IamRole": "HybridDatabaseServersRole",
    "RegistrationLimit": 5,
    "RegistrationsCount": 5,
    "ExpirationDate": 1580515200.0,
    "Expired": true,
    "CreateDate": 1578064132.002
  },
]
}

```

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat Aktivasi Instans Terkelola untuk Lingkungan Hybrid](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeActivations](#) di Referensi AWS CLI Perintah.

## describe-association-execution-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-association-execution-targets`.

### AWS CLI

Untuk mendapatkan detail eksekusi asosiasi

`describe-association-execution-targets` Contoh berikut menjelaskan eksekusi asosiasi yang ditentukan.

```

aws ssm describe-association-execution-targets \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --execution-id "7abb6378-a4a5-4f10-8312-0123456789ab"

```

Output:

```

{
  "AssociationExecutionTargets": [
    {

```

```

    "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
    "AssociationVersion": "1",
    "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
    "ResourceId": "i-1234567890abcdef0",
    "ResourceType": "ManagedInstance",
    "Status": "Success",
    "DetailedStatus": "Success",
    "LastExecutionDate": 1550505538.497,
    "OutputSource": {
      "OutputSourceId": "97fff367-fc5a-4299-aed8-0123456789ab",
      "OutputSourceType": "RunCommand"
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat riwayat asosiasi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeAssociationExecutionTargets](#) di Referensi AWS CLI Perintah.

## describe-association-executions

Contoh kode berikut menunjukkan cara menggunakan `describe-association-executions`.

### AWS CLI

Contoh 1: Untuk mendapatkan rincian semua eksekusi untuk asosiasi

`describe-association-executions` Contoh berikut menjelaskan semua eksekusi asosiasi yang ditentukan.

```

aws ssm describe-association-executions \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"

```

Output:

```

{
  "AssociationExecutions": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",

```

```

        "Status": "Success",
        "DetailedStatus": "Success",
        "CreatedTime": 1550505827.119,
        "ResourceCountByStatus": "{Success=1}"
    },
    {
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "AssociationVersion": "1",
        "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
        "Status": "Success",
        "DetailedStatus": "Success",
        "CreatedTime": 1550505536.843,
        "ResourceCountByStatus": "{Success=1}"
    },
    ...
]
}

```

Untuk informasi selengkapnya, lihat [Melihat riwayat asosiasi](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk mendapatkan rincian semua eksekusi untuk asosiasi setelah tanggal dan waktu tertentu

`describe-association-executions` Contoh berikut menjelaskan semua eksekusi asosiasi setelah tanggal dan waktu yang ditentukan.

```

aws ssm describe-association-executions \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --filters "Key=CreatedTime,Value=2019-02-18T16:00:00Z,Type=GREATER_THAN"

```

Output:

```

{
  "AssociationExecutions": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",
      "Status": "Success",
      "DetailedStatus": "Success",
      "CreatedTime": 1550505827.119,
      "ResourceCountByStatus": "{Success=1}"
    }
  ]
}

```

```

    },
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
      "Status": "Success",
      "DetailedStatus": "Success",
      "CreatedTime": 1550505536.843,
      "ResourceCountByStatus": "{Success=1}"
    },
    ...
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat riwayat asosiasi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeAssociationExecutions](#) di Referensi AWS CLI Perintah.

## describe-association

Contoh kode berikut menunjukkan cara menggunakan `describe-association`.

### AWS CLI

Contoh 1: Untuk mendapatkan detail asosiasi

`describe-association` Contoh berikut menjelaskan asosiasi untuk ID asosiasi tertentu.

```

aws ssm describe-association \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"

```

Output:

```

{
  "AssociationDescription": {
    "Name": "AWS-GatherSoftwareInventory",
    "AssociationVersion": "1",
    "Date": 1534864780.995,
    "LastUpdateAssociationDate": 1543235759.81,
    "Overview": {
      "Status": "Success",
      "AssociationStatusAggregatedCount": {

```

```
        "Success": 2
      }
    },
    "DocumentVersion": "$DEFAULT",
    "Parameters": {
      "applications": [
        "Enabled"
      ],
      "awsComponents": [
        "Enabled"
      ],
      "customInventory": [
        "Enabled"
      ],
      "files": [
        ""
      ],
      "instanceDetailedInformation": [
        "Enabled"
      ],
      "networkConfig": [
        "Enabled"
      ],
      "services": [
        "Enabled"
      ],
      "windowsRegistry": [
        ""
      ],
      "windowsRoles": [
        "Enabled"
      ],
      "windowsUpdates": [
        "Enabled"
      ]
    },
    "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
    "Targets": [
      {
        "Key": "InstanceIds",
        "Values": [
          "*"
        ]
      }
    ]
  }
}
```



```

    ],
    "ScheduleExpression": "rate(24 hours)",
    "LastExecutionDate": 1550501886.0,
    "LastSuccessfulExecutionDate": 1550501886.0,
    "AssociationName": "Inventory-Association"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengedit dan membuat versi baru asosiasi](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk mendapatkan rincian asosiasi untuk contoh dan dokumen tertentu

`describe-association` Contoh berikut menjelaskan hubungan antara instance dan dokumen.

```

aws ssm describe-association \
  --instance-id "i-1234567890abcdef0" \
  --name "AWS-UpdateSSMAgent"

```

Output:

```

{
  "AssociationDescription": {
    "Status": {
      "Date": 1487876122.564,
      "Message": "Associated with AWS-UpdateSSMAgent",
      "Name": "Associated"
    },
    "Name": "AWS-UpdateSSMAgent",
    "InstanceId": "i-1234567890abcdef0",
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Associated",
      "AssociationStatusAggregatedCount": {
        "Pending": 1
      }
    },
    "AssociationId": "d8617c07-2079-4c18-9847-1234567890ab",
    "DocumentVersion": "$DEFAULT",
    "LastUpdateAssociationDate": 1487876122.564,
    "Date": 1487876122.564,
    "Targets": [
      {

```

```

        "Values": [
            "i-1234567890abcdef0"
        ],
        "Key": "InstanceIds"
    }
]
}
}

```

Untuk informasi selengkapnya, lihat [Mengedit dan membuat versi baru asosiasi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeAssociation](#) di Referensi AWS CLI Perintah.

## describe-automation-executions

Contoh kode berikut menunjukkan cara menggunakan `describe-automation-executions`.

### AWS CLI

Untuk menggambarkan eksekusi otomatisasi

`describe-automation-executions` Contoh berikut menampilkan rincian tentang eksekusi Otomasi.

```

aws ssm describe-automation-executions \
  --filters Key=ExecutionId,Values=73c8eef8-f4ee-4a05-820c-e354fEXAMPLE

```

Output:

```

{
  "AutomationExecutionMetadataList": [
    {
      "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",
      "DocumentName": "AWS-StartEC2Instance",
      "DocumentVersion": "1",
      "AutomationExecutionStatus": "Success",
      "ExecutionStartTime": 1583737233.748,
      "ExecutionEndTime": 1583737234.719,
      "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/OrchestrationService",
      "LogFile": "",
      "Outputs": {}
    }
  ]
}

```

```
    "Mode": "Auto",
    "Targets": [],
    "ResolvedTargets": {
      "ParameterValues": [],
      "Truncated": false
    },
    "AutomationType": "Local"
  }
]
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Alur Kerja Otomasi Sederhana](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeAutomationExecutions](#) di Referensi AWS CLI Perintah.

## describe-automation-step-executions

Contoh kode berikut menunjukkan cara menggunakan `describe-automation-step-executions`.

### AWS CLI

Contoh 1: Untuk menjelaskan semua langkah untuk eksekusi otomatisasi

`describe-automation-step-executions` Contoh berikut menampilkan rincian tentang langkah-langkah eksekusi Otomasi.

```
aws ssm describe-automation-step-executions \
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Output:

```
{
  "StepExecutions": [
    {
      "StepName": "startInstances",
      "Action": "aws:changeInstanceState",
      "ExecutionStartTime": 1583737234.134,
      "ExecutionEndTime": 1583737234.672,
      "StepStatus": "Success",
      "Inputs": {
```

```

        "DesiredState": "\"running\"",
        "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
    },
    "Outputs": {
        "InstanceStates": [
            "running"
        ]
    },
    "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
    "OverriddenParameters": {}
}
]
}

```

Contoh 2: Untuk menjelaskan langkah spesifik untuk eksekusi otomatisasi

`describe-automation-step-executions` Contoh berikut menampilkan rincian tentang langkah tertentu dari eksekusi Otomasi.

```

aws ssm describe-automation-step-executions \
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
  --filters Key=StepExecutionId,Values=95e70479-cf20-4d80-8018-7e4e2EXAMPLE

```

Untuk informasi selengkapnya, lihat [Menjalankan Alur Kerja Otomasi Langkah demi Langkah \(Baris Perintah\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeAutomationStepExecutions](#) di Referensi AWS CLI Perintah.

## **describe-available-patches**

Contoh kode berikut menunjukkan cara menggunakan `describe-available-patches`.

### AWS CLI

Untuk mendapatkan tambalan yang tersedia

`describe-available-patches` Contoh berikut mengambil rincian tentang semua patch yang tersedia untuk Windows Server 2019 yang memiliki tingkat keparahan MSRC Critical.

```

aws ssm describe-available-patches \
  --filters "Key=PRODUCT,Values=WindowsServer2019"
  "Key=MSRC_SEVERITY,Values=Critical"

```

## Output:

```
{
  "Patches": [
    {
      "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
      "ReleaseDate": 1544047205.0,
      "Title": "2018-11 Update for Windows Server 2019 for x64-based Systems (KB4470788)",
      "Description": "Install this update to resolve issues in Windows. For a complete listing of the issues that are included in this update, see the associated Microsoft Knowledge Base article for more information. After you install this item, you may have to restart your computer.",
      "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
      "Vendor": "Microsoft",
      "ProductFamily": "Windows",
      "Product": "WindowsServer2019",
      "Classification": "SecurityUpdates",
      "MsrcSeverity": "Critical",
      "KbNumber": "KB4470788",
      "MsrcNumber": "",
      "Language": "All"
    },
    {
      "Id": "c96115e1-5587-4115-b851-22baa46a3f11",
      "ReleaseDate": 1549994410.0,
      "Title": "2019-02 Security Update for Adobe Flash Player for Windows Server 2019 for x64-based Systems (KB4487038)",
      "Description": "A security issue has been identified in a Microsoft software product that could affect your system. You can help protect your system by installing this update from Microsoft. For a complete listing of the issues that are included in this update, see the associated Microsoft Knowledge Base article. After you install this update, you may have to restart your system.",
      "ContentUrl": "https://support.microsoft.com/en-us/kb/4487038",
      "Vendor": "Microsoft",
      "ProductFamily": "Windows",
      "Product": "WindowsServer2019",
      "Classification": "SecurityUpdates",
      "MsrcSeverity": "Critical",
      "KbNumber": "KB4487038",
      "MsrcNumber": "",
      "Language": "All"
    },
    ...
  ]
}
```

```
]
}
```

Untuk mendapatkan detail tambalan tertentu

`describe-available-patches` Contoh berikut mengambil rincian tentang patch yang ditentukan.

```
aws ssm describe-available-patches \
  --filters "Key=PATCH_ID,Values=KB4480979"
```

Output:

```
{
  "Patches": [
    {
      "Id": "680861e3-fb75-432e-818e-d72e5f2be719",
      "ReleaseDate": 1546970408.0,
      "Title": "2019-01 Security Update for Adobe Flash Player for Windows Server 2016 for x64-based Systems (KB4480979)",
      "Description": "A security issue has been identified in a Microsoft software product that could affect your system. You can help protect your system by installing this update from Microsoft. For a complete listing of the issues that are included in this update, see the associated Microsoft Knowledge Base article. After you install this update, you may have to restart your system.",
      "ContentUrl": "https://support.microsoft.com/en-us/kb/4480979",
      "Vendor": "Microsoft",
      "ProductFamily": "Windows",
      "Product": "WindowsServer2016",
      "Classification": "SecurityUpdates",
      "MsrcSeverity": "Critical",
      "KbNumber": "KB4480979",
      "MsrcNumber": "",
      "Language": "All"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Cara Kerja Operasi Manajer Patch](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeAvailablePatches](#) di Referensi AWS CLI Perintah.

## describe-document-permission

Contoh kode berikut menunjukkan cara menggunakan `describe-document-permission`.

### AWS CLI

Untuk menjelaskan izin dokumen

`describe-document-permission` Contoh berikut menampilkan rincian izin tentang dokumen Systems Manager yang dibagikan secara publik.

```
aws ssm describe-document-permission \  
  --name "Example" \  
  --permission-type "Share"
```

Output:

```
{  
  "AccountIds": [  
    "all"  
  ],  
  "AccountSharingInfoList": [  
    {  
      "AccountId": "all",  
      "SharedDocumentVersion": "$DEFAULT"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Membagikan Dokumen Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeDocumentPermission](#) di Referensi AWS CLI Perintah.

## describe-document

Contoh kode berikut menunjukkan cara menggunakan `describe-document`.

### AWS CLI

Untuk menampilkan detail dokumen

`describe-document` Contoh berikut menampilkan detail tentang dokumen Systems Manager di AWS akun Anda.

```
aws ssm describe-document \  
  --name "Example"
```

Output:

```
{  
  "Document": {  
    "Hash": "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",  
    "HashType": "Sha256",  
    "Name": "Example",  
    "Owner": "29884EXAMPLE",  
    "CreateDate": 1583257938.266,  
    "Status": "Active",  
    "DocumentVersion": "1",  
    "Description": "Document Example",  
    "Parameters": [  
      {  
        "Name": "AutomationAssumeRole",  
        "Type": "String",  
        "Description": "(Required) The ARN of the role that allows  
Automation to perform the actions on your behalf. If no role is specified, Systems  
Manager Automation uses your IAM permissions to execute this document.",  
        "DefaultValue": ""  
      },  
      {  
        "Name": "InstanceId",  
        "Type": "String",  
        "Description": "(Required) The ID of the Amazon EC2 instance.",  
        "DefaultValue": ""  
      }  
    ],  
    "PlatformTypes": [  
      "Windows",  
      "Linux"  
    ],  
    "DocumentType": "Automation",  
    "SchemaVersion": "0.3",  
    "LatestVersion": "1",  
    "DefaultVersion": "1",  
    "DocumentFormat": "YAML",  
  }  
}
```



```

    "Tags": []
  }
}

```

Untuk informasi selengkapnya, lihat [Membuat Dokumen Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeDocument](#) di Referensi AWS CLI Perintah.

## describe-effective-instance-associations

Contoh kode berikut menunjukkan cara menggunakan `describe-effective-instance-associations`.

### AWS CLI

Untuk mendapatkan rincian asosiasi yang efektif untuk sebuah contoh

`describe-effective-instance-associations` Contoh berikut mengambil rincian tentang asosiasi efektif untuk sebuah contoh.

Perintah:

```
aws ssm describe-effective-instance-associations --instance-id "i-1234567890abcdef0"
```

Output:

```

{
  "Associations": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "InstanceId": "i-1234567890abcdef0",
      "Content": "{\n  \"schemaVersion\": \"1.2\",\n  \"description\":\n  \"Update the Amazon SSM Agent to the latest version or specified version.\",\n  \"parameters\": {\n    \"version\": {\n      \"default\": \"\",\n      \"description\": \"(Optional) A specific version of the Amazon SSM Agent\n  to install. If not specified, the agent will be updated to the latest version.\",\n      \"type\": \"String\"\n    },\n    \"allowDowngrade\": {\n      \"default\": \"false\",\n      \"description\": \"(Optional)\n  Allow the Amazon SSM Agent service to be downgraded to an earlier version. If\n  set to false, the service can be upgraded to newer versions only (default). If\n  set to true, specify the earlier version.\",\n      \"type\": \"String\",\n      \"allowedValues\": [\n        \"true\"\n      ]
    }
  }"
    }
  ]
}

```

```

{"false": true,
 "aws:updateSsmAgent": {
   "properties": [
     {
       "agentName": "amazon-ssm-agent",
       "source": "https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/ssm-agent-manifest.json",
       "allowDowngrade": "{ allowDowngrade }",
       "targetVersion": "{ version }"
     }
   ],
   "AssociationVersion": "1"
 }
}

```

- Untuk detail API, lihat [DescribeEffectiveInstanceAssociations](#) di Referensi AWS CLI Perintah.

## describe-effective-patches-for-patch-baseline

Contoh kode berikut menunjukkan cara menggunakan `describe-effective-patches-for-patch-baseline`.

### AWS CLI

Contoh 1: Untuk mendapatkan semua tambalan yang ditentukan oleh baseline patch kustom

`describe-effective-patches-for-patch-baseline` Contoh berikut mengembalikan patch yang ditentukan oleh baseline patch kustom di akun saat ini. AWS Perhatikan bahwa untuk baseline kustom, hanya ID yang diperlukan untuk. `--baseline-id`

```
aws ssm describe-effective-patches-for-patch-baseline \
  --baseline-id "pb-08b654cf9b9681f04"
```

Output:

```
{
  "EffectivePatches": [
    {
      "Patch": {
        "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
        "ReleaseDate": 1544047205.0,
        "Title": "2018-11 Update for Windows Server 2019 for x64-based Systems (KB4470788)",
        "Description": "Install this update to resolve issues in Windows. For a complete listing of the issues that are included in this update, see the"
      }
    }
  ]
}
```

```

associated Microsoft Knowledge Base article for more information. After you install
this item, you may have to restart your computer.",
    "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
    "Vendor": "Microsoft",
    "ProductFamily": "Windows",
    "Product": "WindowsServer2019",
    "Classification": "SecurityUpdates",
    "MsrcSeverity": "Critical",
    "KbNumber": "KB4470788",
    "MsrcNumber": "",
    "Language": "All"
  },
  "PatchStatus": {
    "DeploymentStatus": "APPROVED",
    "ComplianceLevel": "CRITICAL",
    "ApprovalDate": 1544047205.0
  }
},
{
  "Patch": {
    "Id": "915a6b1a-f556-4d83-8f50-b2e75a9a7e58",
    "ReleaseDate": 1549994400.0,
    "Title": "2019-02 Cumulative Update for .NET Framework 3.5 and 4.7.2
for Windows Server 2019 for x64 (KB4483452)",
    "Description": "A security issue has been identified in a Microsoft
software product that could affect your system. You can help protect your system by
installing this update from Microsoft. For a complete listing of the issues that
are included in this update, see the associated Microsoft Knowledge Base article.
After you install this update, you may have to restart your system.",
    "ContentUrl": "https://support.microsoft.com/en-us/kb/4483452",
    "Vendor": "Microsoft",
    "ProductFamily": "Windows",
    "Product": "WindowsServer2019",
    "Classification": "SecurityUpdates",
    "MsrcSeverity": "Important",
    "KbNumber": "KB4483452",
    "MsrcNumber": "",
    "Language": "All"
  },
  "PatchStatus": {
    "DeploymentStatus": "APPROVED",
    "ComplianceLevel": "CRITICAL",
    "ApprovalDate": 1549994400.0
  }
}

```

```
    },  
    ...  
  ],  
  "NextToken": "--token string truncated--"  
}
```

Contoh 2: Untuk mendapatkan semua tambalan yang ditentukan oleh baseline patch AWS terkelola

`describe-effective-patches-for-patch-baseline` Contoh berikut mengembalikan patch yang ditentukan oleh baseline patch AWS terkelola. Perhatikan bahwa untuk baseline AWS terkelola, ARN baseline lengkap diperlukan untuk `--baseline-id`

```
aws ssm describe-effective-patches-for-patch-baseline \  
  --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/  
pb-020d361a05defe4ed"
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Bagaimana Patch Keamanan Dipilih](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeEffectivePatchesForPatchBaseline](#) di Referensi AWS CLI Perintah.

## **describe-instance-associations-status**

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-associations-status`.

### AWS CLI

Untuk menggambarkan status asosiasi instans

Contoh ini menunjukkan rincian asosiasi untuk sebuah instance.

Perintah:

```
aws ssm describe-instance-associations-status --instance-id "i-1234567890abcdef0"
```

Output:

```
{
  "InstanceAssociationStatusInfos": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "Name": "AWS-GatherSoftwareInventory",
      "DocumentVersion": "1",
      "AssociationVersion": "1",
      "InstanceId": "i-1234567890abcdef0",
      "ExecutionDate": 1550501886.0,
      "Status": "Success",
      "ExecutionSummary": "1 out of 1 plugin processed, 1 success, 0 failed, 0
      timedout, 0 skipped. ",
      "AssociationName": "Inventory-Association"
    },
    {
      "AssociationId": "5c5a31f6-6dae-46f9-944c-0123456789ab",
      "Name": "AWS-UpdateSSMAgent",
      "DocumentVersion": "1",
      "AssociationVersion": "1",
      "InstanceId": "i-1234567890abcdef0",
      "ExecutionDate": 1550505828.548,
      "Status": "Success",
      "DetailedStatus": "Success",
      "AssociationName": "UpdateSSMAgent"
    }
  ]
}
```

- Untuk detail API, lihat [DescribeInstanceAssociationsStatus](#) di Referensi AWS CLI Perintah.

## describe-instance-information

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-information`.

### AWS CLI

Contoh 1: Untuk mendeskripsikan informasi instans terkelola

`describe-instance-information` Contoh berikut mengambil detail dari setiap instans terkelola Anda.

```
aws ssm describe-instance-information
```

## Contoh 2: Untuk mendeskripsikan informasi tentang instance terkelola tertentu

`describe-instance-information` Contoh berikut menunjukkan rincian instance yang dikelola `i-028ea792daEXAMPLE`.

```
aws ssm describe-instance-information \  
  --filters "Key=InstanceIds,Values=i-028ea792daEXAMPLE"
```

## Contoh 3: Untuk menjelaskan informasi tentang instance terkelola dengan kunci tag tertentu

`describe-instance-information` Contoh berikut menunjukkan rincian untuk instance terkelola yang memiliki kunci DEV tag.

```
aws ssm describe-instance-information \  
  --filters "Key=tag-key,Values=DEV"
```

## Output:

```
{  
  "InstanceInformationList": [  
    {  
      "InstanceId": "i-028ea792daEXAMPLE",  
      "PingStatus": "Online",  
      "LastPingDateTime": 1582221233.421,  
      "AgentVersion": "2.3.842.0",  
      "IsLatestVersion": true,  
      "PlatformType": "Linux",  
      "PlatformName": "SLES",  
      "PlatformVersion": "15.1",  
      "ResourceType": "EC2Instance",  
      "IPAddress": "192.0.2.0",  
      "ComputerName": "ip-198.51.100.0.us-east-2.compute.internal",  
      "AssociationStatus": "Success",  
      "LastAssociationExecutionDate": 1582220806.0,  
      "LastSuccessfulAssociationExecutionDate": 1582220806.0,  
      "AssociationOverview": {  
        "DetailedStatus": "Success",  
        "InstanceAssociationStatusAggregatedCount": {  
          "Success": 2  
        }  
      }  
    }  
  ]  
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Instans Terkelola](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeInstanceInformation](#) di Referensi AWS CLI Perintah.

## describe-instance-patch-states-for-patch-group

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-patch-states-for-patch-group`.

### AWS CLI

Contoh 1: Untuk mendapatkan status instance untuk grup patch

`describe-instance-patch-states-for-patch-group` Contoh berikut mengambil rincian tentang status ringkasan patch per instance untuk grup patch yang ditentukan.

```
aws ssm describe-instance-patch-states-for-patch-group \
  --patch-group "Production"
```

Output:

```
{
  "InstancePatchStates": [
    {
      "InstanceId": "i-02573cafcfEXAMPLE",
      "PatchGroup": "Production",
      "BaselineId": "pb-0c10e65780EXAMPLE",
      "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
      "OwnerInformation": "",
      "InstalledCount": 32,
      "InstalledOtherCount": 1,
      "InstalledPendingRebootCount": 0,
      "InstalledRejectedCount": 0,
      "MissingCount": 2,
      "FailedCount": 0,
      "UnreportedNotApplicableCount": 2671,
      "NotApplicableCount": 400,
      "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
      "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
    }
  ]
}
```

```

    "Operation": "Scan",
    "RebootOption": "NoReboot",
    "CriticalNonCompliantCount": 0,
    "SecurityNonCompliantCount": 1,
    "OtherNonCompliantCount": 0
  },
  {
    "InstanceId": "i-0471e04240EXAMPLE",
    "PatchGroup": "Production",
    "BaselineId": "pb-09ca3fb51fEXAMPLE",
    "SnapshotId": "05d8ffb0-1bbe-4812-ba2d-d9b7bEXAMPLE",
    "OwnerInformation": "",
    "InstalledCount": 32,
    "InstalledOtherCount": 1,
    "InstalledPendingRebootCount": 0,
    "InstalledRejectedCount": 0,
    "MissingCount": 2,
    "FailedCount": 0,
    "UnreportedNotApplicableCount": 2671,
    "NotApplicableCount": 400,
    "OperationStartTime": "2021-08-04T22:06:20.340000-07:00",
    "OperationEndTime": "2021-08-04T22:07:11.220000-07:00",
    "Operation": "Scan",
    "RebootOption": "NoReboot",
    "CriticalNonCompliantCount": 0,
    "SecurityNonCompliantCount": 1,
    "OtherNonCompliantCount": 0
  }
]
}

```

Contoh 2: Untuk mendapatkan status instance untuk grup tambalan dengan lebih dari lima tambalan yang hilang

`describe-instance-patch-states-for-patch-group` Contoh berikut mengambil detail tentang status ringkasan tambalan untuk grup patch yang ditentukan untuk instance dengan lebih dari lima tambalan yang hilang.

```

aws ssm describe-instance-patch-states-for-patch-group \
  --filters Key=MissingCount,Type=GreaterThan,Values=5 \
  --patch-group "Production"

```

Output:



```
{
  "InstancePatchStates": [
    {
      "InstanceId": "i-02573cafcfEXAMPLE",
      "PatchGroup": "Production",
      "BaselineId": "pb-0c10e65780EXAMPLE",
      "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
      "OwnerInformation": "",
      "InstalledCount": 46,
      "InstalledOtherCount": 4,
      "InstalledPendingRebootCount": 1,
      "InstalledRejectedCount": 1,
      "MissingCount": 7,
      "FailedCount": 0,
      "UnreportedNotApplicableCount": 232,
      "NotApplicableCount": 654,
      "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
      "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
      "Operation": "Scan",
      "RebootOption": "NoReboot",
      "CriticalNonCompliantCount": 0,
      "SecurityNonCompliantCount": 1,
      "OtherNonCompliantCount": 1
    }
  ]
}
```

Contoh 3: Untuk mendapatkan status instance untuk grup patch dengan kurang dari sepuluh instance yang memerlukan reboot

`describe-instance-patch-states-for-patch-group` Contoh berikut mengambil detail tentang status ringkasan tambalan untuk grup tambalan yang ditentukan untuk instance dengan kurang dari sepuluh instance yang memerlukan reboot.

```
aws ssm describe-instance-patch-states-for-patch-group \
  --filters Key=InstalledPendingRebootCount,Type=LessThan,Values=10 \
  --patch-group "Production"
```

Output:

```
{
  "InstancePatchStates": [
```

```
{
  "InstanceId": "i-02573cafcfEXAMPLE",
  "BaselineId": "pb-0c10e65780EXAMPLE",
  "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
  "PatchGroup": "Production",
  "OwnerInformation": "",
  "InstalledCount": 32,
  "InstalledOtherCount": 1,
  "InstalledPendingRebootCount": 4,
  "InstalledRejectedCount": 0,
  "MissingCount": 2,
  "FailedCount": 0,
  "UnreportedNotApplicableCount": 846,
  "NotApplicableCount": 212,
  "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
  "OperationEndTime": "2021-08-06T11:04:21.555000-07:00",
  "Operation": "Scan",
  "RebootOption": "NoReboot",
  "CriticalNonCompliantCount": 0,
  "SecurityNonCompliantCount": 1,
  "OtherNonCompliantCount": 0
}
]
```

Untuk informasi selengkapnya, lihat [Memahami nilai status kepatuhan tambalan](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeInstancePatchStatesForPatchGroup](#) di Referensi AWS CLI Perintah.

## describe-instance-patch-states

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-patch-states`.

### AWS CLI

Untuk mendapatkan status ringkasan tambalan untuk instance

`describe-instance-patch-states` Contoh ini mendapatkan status ringkasan patch untuk sebuah instance.

```
aws ssm describe-instance-patch-states \
```

```
--instance-ids "i-1234567890abcdef0"
```

Output:

```
{
  "InstancePatchStates": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "PatchGroup": "my-patch-group",
      "BaselineId": "pb-0713accee01234567",
      "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",
      "CriticalNonCompliantCount": 2,
      "SecurityNonCompliantCount": 2,
      "OtherNonCompliantCount": 1,
      "InstalledCount": 123,
      "InstalledOtherCount": 334,
      "InstalledPendingRebootCount": 0,
      "InstalledRejectedCount": 0,
      "MissingCount": 1,
      "FailedCount": 2,
      "UnreportedNotApplicableCount": 11,
      "NotApplicableCount": 2063,
      "OperationStartTime": "2021-05-03T11:00:56-07:00",
      "OperationEndTime": "2021-05-03T11:01:09-07:00",
      "Operation": "Scan",
      "LastNoRebootInstallOperationTime": "2020-06-14T12:17:41-07:00",
      "RebootOption": "RebootIfNeeded"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Tentang Patch Compliance](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeInstancePatchStates](#) di Referensi AWS CLI Perintah.

## describe-instance-patches

Contoh kode berikut menunjukkan cara menggunakan `describe-instance-patches`.

### AWS CLI

Contoh 1: Untuk mendapatkan detail status patch untuk sebuah instance

`describe-instance-patches` Contoh berikut mengambil rincian tentang patch untuk contoh tertentu.

```
aws ssm describe-instance-patches \  
  --instance-id "i-1234567890abcdef0"
```

Output:

```
{  
  "Patches": [  
    {  
      "Title": "2019-01 Security Update for Adobe Flash Player for Windows  
Server 2016 for x64-based Systems (KB4480979)",  
      "KBId": "KB4480979",  
      "Classification": "SecurityUpdates",  
      "Severity": "Critical",  
      "State": "Installed",  
      "InstalledTime": "2019-01-09T00:00:00+00:00"  
    },  
    {  
      "Title": "",  
      "KBId": "KB4481031",  
      "Classification": "",  
      "Severity": "",  
      "State": "InstalledOther",  
      "InstalledTime": "2019-02-08T00:00:00+00:00"  
    },  
    ...  
  ],  
  "NextToken": "--token string truncated--"  
}
```

Contoh 2: Untuk mendapatkan daftar tambalan dalam keadaan Hilang untuk sebuah instance

`describe-instance-patches` Contoh berikut mengambil informasi tentang patch yang berada dalam keadaan Hilang untuk instance tertentu.

```
aws ssm describe-instance-patches \  
  --instance-id "i-1234567890abcdef0" \  
  --filters Key=State,Values=Missing
```

Output:

```
{
  "Patches": [
    {
      "Title": "Windows Malicious Software Removal Tool x64 - February 2019
(KB890830)",
      "KBId": "KB890830",
      "Classification": "UpdateRollups",
      "Severity": "Unspecified",
      "State": "Missing",
      "InstalledTime": "1970-01-01T00:00:00+00:00"
    },
    ...
  ],
  "NextToken": "--token string truncated--"
}
```

Untuk informasi selengkapnya, lihat [Tentang Status Kepatuhan Patch](#) di Panduan Pengguna AWS Systems Manager.

Contoh 3: Untuk mendapatkan daftar tambalan yang diinstal sejak ditentukan InstalledTime untuk sebuah instance

`describe-instance-patches` Contoh berikut mengambil informasi tentang patch diinstal sejak waktu yang ditentukan untuk instance tertentu dengan menggabungkan penggunaan `--filters` dan `--query`

```
aws ssm describe-instance-patches \
  --instance-id "i-1234567890abcdef0" \
  --filters Key=State,Values=Installed \
  --query "Patches[?InstalledTime >= `2023-01-01T16:00:00`]"
```

Output:

```
{
  "Patches": [
    {
      "Title": "2023-03 Cumulative Update for Windows Server 2019 (1809) for
x64-based Systems (KB5023702)",
      "KBId": "KB5023702",
      "Classification": "SecurityUpdates",
      "Severity": "Critical",
      "State": "Installed",
    }
  ]
}
```

```
        "InstalledTime": "2023-03-16T11:00:00+00:00"
    },
    ...
],
"NextToken": "--token string truncated--"
}
```

- Untuk detail API, lihat [DescribeInstancePatches](#) di Referensi AWS CLI Perintah.

## describe-inventory-deletions

Contoh kode berikut menunjukkan cara menggunakan `describe-inventory-deletions`.

### AWS CLI

Untuk mendapatkan penghapusan inventaris

Contoh ini mengambil rincian untuk operasi penghapusan inventaris.

Perintah:

```
aws ssm describe-inventory-deletions
```

Output:

```
{
  "InventoryDeletions": [
    {
      "DeletionId": "6961492a-8163-44ec-aa1e-01234567850",
      "TypeName": "Custom:RackInformation",
      "DeletionStartTime": 1550254911.0,
      "LastStatus": "InProgress",
      "LastStatusMessage": "The Delete is in progress",
      "DeletionSummary": {
        "TotalCount": 0,
        "RemainingCount": 0,
        "SummaryItems": []
      },
      "LastStatusUpdateTime": 1550254911.0
    },
    {
      "DeletionId": "d72ac9e8-1f60-4d40-b1c6-987654321c4d",
```

```
    "TypeName": "Custom:RackInfo",
    "DeletionStartTime": 1550254859.0,
    "LastStatus": "InProgress",
    "LastStatusMessage": "The Delete is in progress",
    "DeletionSummary": {
      "TotalCount": 1,
      "RemainingCount": 1,
      "SummaryItems": [
        {
          "Version": "1.0",
          "Count": 1,
          "RemainingCount": 1
        }
      ]
    },
    "LastStatusUpdateTime": 1550254859.0
  }
]
```

Untuk mendapatkan detail penghapusan inventaris tertentu

Contoh ini mengambil rincian untuk operasi penghapusan inventaris tertentu.

Perintah:

```
aws ssm describe-inventory-deletions --deletion-id "d72ac9e8-1f60-4d40-
b1c6-987654321c4d"
```

Output:

```
{
  "InventoryDeletions": [
    {
      "DeletionId": "d72ac9e8-1f60-4d40-b1c6-987654321c4d",
      "TypeName": "Custom:RackInfo",
      "DeletionStartTime": 1550254859.0,
      "LastStatus": "InProgress",
      "LastStatusMessage": "The Delete is in progress",
      "DeletionSummary": {
        "TotalCount": 1,
        "RemainingCount": 1,
        "SummaryItems": [
```

```

        {
            "Version": "1.0",
            "Count": 1,
            "RemainingCount": 1
        }
    ],
    "LastStatusUpdateTime": 1550254859.0
}
]
}

```

- Untuk detail API, lihat [DescribeInventoryDeletions](#) di Referensi AWS CLI Perintah.

## describe-maintenance-window-execution-task-invocations

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-window-execution-task-invocations`.

### AWS CLI

Untuk mendapatkan pemanggilan tugas tertentu dilakukan untuk eksekusi tugas jendela pemeliharaan

`describe-maintenance-window-execution-task-invocations` Contoh berikut mencantumkan pemanggilan untuk tugas tertentu yang dijalankan sebagai bagian dari eksekusi jendela pemeliharaan yang ditentukan.

```

aws ssm describe-maintenance-window-execution-task-invocations \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2a638355" \
  --task-id "ac0c6ae1-daa3-4a89-832e-d384503b6586"

```

### Output:

```

{
  "WindowExecutionTaskInvocationIdentities": [
    {
      "Status": "SUCCESS",
      "Parameters": "{\"documentName\":\"AWS-RunShellScript\",\"instanceIds\":[\"i-0000293ffd8c57862\"],\"parameters\":{\"commands\":[\"df\"]},\"maxConcurrency\":1,\"maxErrors\":\"1\"}"
    }
  ]
}

```



```

    "InvocationId": "e274b6e1-fe56-4e32-bd2a-8073c6381d8b",
    "StartTime": 1487692834.723,
    "EndTime": 1487692834.871,
    "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2a638355",
    "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d384503b6586"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Tugas dan Eksekusi Tugas \(AWS CLI\)](#) di Panduan Pengguna Systems AWS Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowExecutionTaskInvocations](#) di Referensi AWS CLI Perintah.

## describe-maintenance-window-execution-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-window-execution-tasks`.

### AWS CLI

Untuk membuat daftar semua tugas yang terkait dengan eksekusi jendela pemeliharaan

`ssm describe-maintenance-window-execution-tasks` Contoh berikut mencantumkan tugas-tugas yang terkait dengan eksekusi jendela pemeliharaan yang ditentukan.

```

aws ssm describe-maintenance-window-execution-tasks \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"

```

### Output:

```

{
  "WindowExecutionTaskIdentities": [
    {
      "Status": "SUCCESS",
      "TaskArn": "AWS-RunShellScript",
      "StartTime": 1487692834.684,
      "TaskType": "RUN_COMMAND",
      "EndTime": 1487692835.005,
      "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
      "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
    }
  ]
}

```

```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Tugas dan Eksekusi Tugas \(AWS CLI\)](#) di Panduan Pengguna Systems AWS Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowExecutionTasks](#) di Referensi AWS CLI Perintah.

## describe-maintenance-window-executions

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-window-executions`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua eksekusi untuk jendela pemeliharaan

`describe-maintenance-window-executions` Contoh berikut mencantumkan semua eksekusi untuk jendela pemeliharaan yang ditentukan.

```
aws ssm describe-maintenance-window-executions \  
  --window-id "mw-ab12cd34eEXAMPLE"
```

Output:

```
{  
  "WindowExecutions": [  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",  
      "Status": "IN_PROGRESS",  
      "StartTime": "2021-08-04T11:00:00.000000-07:00"  
    },  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "WindowExecutionId": "ff75b750-4834-4377-8f61-b3cadEXAMPLE",  
      "Status": "SUCCESS",  
      "StartTime": "2021-08-03T11:00:00.000000-07:00",  
      "EndTime": "2021-08-03T11:37:21.450000-07:00"  
    }  
  ]  
}
```

```

    },
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
      "Status": "FAILED",
      "StatusDetails": "One or more tasks in the orchestration failed.",
      "StartTime": "2021-08-02T11:00:00.000000-07:00",
      "EndTime": "2021-08-02T11:22:36.190000-07:00"
    }
  ]
}

```

Contoh 2: Untuk mencantumkan semua eksekusi untuk jendela pemeliharaan sebelum tanggal yang ditentukan

`describe-maintenance-window-executions` Contoh berikut mencantumkan semua eksekusi untuk jendela pemeliharaan yang ditentukan sebelum tanggal yang ditentukan.

```

aws ssm describe-maintenance-window-executions \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=ExecutedBefore,Values=2021-08-03T00:00:00Z"

```

Output:

```

{
  "WindowExecutions": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
      "Status": "FAILED",
      "StatusDetails": "One or more tasks in the orchestration failed.",
      "StartTime": "2021-08-02T11:00:00.000000-07:00",
      "EndTime": "2021-08-02T11:22:36.190000-07:00"
    }
  ]
}

```

Contoh 3: Untuk mencantumkan semua eksekusi untuk jendela pemeliharaan setelah tanggal yang ditentukan

`describe-maintenance-window-executions` Contoh berikut mencantumkan semua eksekusi untuk jendela pemeliharaan yang ditentukan setelah tanggal yang ditentukan.

```
aws ssm describe-maintenance-window-executions \  
  --window-id "mw-ab12cd34eEXAMPLE" \  
  --filters "Key=ExecutedAfter,Values=2021-08-04T00:00:00Z"
```

Output:

```
{  
  "WindowExecutions": [  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",  
      "Status": "IN_PROGRESS",  
      "StartTime": "2021-08-04T11:00:00.000000-07:00"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Melihat informasi tentang tugas dan eksekusi tugas \(AWS CLI\)](#) di Panduan Pengguna Systems AWS Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowExecutions](#) di Referensi AWS CLI Perintah.

## describe-maintenance-window-schedule

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-window-schedule`.

AWS CLI

Contoh 1: Untuk mencantumkan eksekusi yang akan datang untuk jendela pemeliharaan

`describe-maintenance-window-schedule` Contoh berikut mencantumkan semua eksekusi yang akan datang untuk jendela pemeliharaan yang ditentukan.

```
aws ssm describe-maintenance-window-schedule \  
  --window-id mw-ab12cd34eEXAMPLE
```

Output:

```
{
```

```

    "ScheduledWindowExecutions": [
      {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "Name": "My-First-Maintenance-Window",
        "ExecutionTime": "2020-02-19T16:00Z"
      },
      {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "Name": "My-First-Maintenance-Window",
        "ExecutionTime": "2020-02-26T16:00Z"
      },
      ...
    ]
  }

```

Contoh 2: Untuk mencantumkan semua eksekusi yang akan datang untuk jendela pemeliharaan sebelum tanggal yang ditentukan

`describe-maintenance-window-schedule` Contoh berikut mencantumkan semua eksekusi yang akan datang untuk jendela pemeliharaan tertentu yang terjadi sebelum tanggal yang ditentukan.

```

aws ssm describe-maintenance-window-schedule \
  --window-id mw-0ecb1226dd7b2e9a6 \
  --filters "Key=ScheduledBefore,Values=2020-02-15T06:00:00Z"

```

Contoh 3: Untuk mencantumkan semua eksekusi yang akan datang untuk jendela pemeliharaan setelah tanggal yang ditentukan

`describe-maintenance-window-schedule` Contoh berikut mencantumkan semua eksekusi yang akan datang untuk jendela pemeliharaan tertentu yang terjadi setelah tanggal yang ditentukan.

```

aws ssm describe-maintenance-window-schedule \
  --window-id mw-0ecb1226dd7b2e9a6 \
  --filters "Key=ScheduledAfter,Values=2020-02-15T06:00:00Z"

```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Pemeliharaan Windows \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowScheduled](#) di Referensi AWS CLI Perintah.

## describe-maintenance-window-targets

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-window-targets`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua target untuk Jendela Pemeliharaan

`describe-maintenance-window-targets` Contoh berikut mencantumkan semua target untuk jendela pemeliharaan.

```
aws ssm describe-maintenance-window-targets \
  --window-id "mw-06cf17cbefEXAMPLE"
```

Output:

```
{
  "Targets": [
    {
      "ResourceType": "INSTANCE",
      "OwnerInformation": "Single instance",
      "WindowId": "mw-06cf17cbefEXAMPLE",
      "Targets": [
        {
          "Values": [
            "i-0000293ffdEXAMPLE"
          ],
          "Key": "InstanceIds"
        }
      ],
      "WindowTargetId": "350d44e6-28cc-44e2-951f-4b2c9EXAMPLE"
    },
    {
      "ResourceType": "INSTANCE",
      "OwnerInformation": "Two instances in a list",
      "WindowId": "mw-06cf17cbefEXAMPLE",
      "Targets": [
        {
          "Values": [
            "i-0000293ffdEXAMPLE",
            "i-0cb2b964d3EXAMPLE"
          ],
          "Key": "InstanceIds"
        }
      ],
      "WindowTargetId": "350d44e6-28cc-44e2-951f-4b2c9EXAMPLE"
    }
  ]
}
```

```

        "Key": "InstanceIds"
      }
    ],
    "WindowTargetId": "e078a987-2866-47be-bedd-d9cf4EXAMPLE"
  }
]
}

```

Contoh 2: Untuk mencantumkan semua target untuk jendela pemeliharaan yang cocok dengan nilai informasi pemilik tertentu

`describe-maintenance-window-targets` Contoh ini mencantumkan semua target untuk jendela pemeliharaan dengan nilai tertentu.

```

aws ssm describe-maintenance-window-targets \
  --window-id "mw-0ecb1226ddEXAMPLE" \
  --filters "Key=OwnerInformation,Values=CostCenter1"

```

Output:

```

{
  "Targets": [
    {
      "WindowId": "mw-0ecb1226ddEXAMPLE",
      "WindowTargetId": "da89dcc3-7f9c-481d-ba2b-edcb7d0057f9",
      "ResourceType": "INSTANCE",
      "Targets": [
        {
          "Key": "tag:Environment",
          "Values": [
            "Prod"
          ]
        }
      ],
      "OwnerInformation": "CostCenter1",
      "Name": "ProdTarget1"
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Pemeliharaan Windows \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowTargets](#) di Referensi AWS CLI Perintah.

## describe-maintenance-window-tasks

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-window-tasks`.

### AWS CLI

Contoh 1: Untuk membuat daftar semua tugas untuk jendela pemeliharaan

`describe-maintenance-window-tasks` Contoh berikut mencantumkan semua tugas untuk jendela pemeliharaan yang ditentukan.

```
aws ssm describe-maintenance-window-tasks \
  --window-id "mw-06cf17cbefEXAMPLE"
```

Output:

```
{
  "Tasks": [
    {
      "WindowId": "mw-06cf17cbefEXAMPLE",
      "WindowTaskId": "018b31c3-2d77-4b9e-bd48-c91edEXAMPLE",
      "TaskArn": "AWS-RestartEC2Instance",
      "TaskParameters": {},
      "Type": "AUTOMATION",
      "Description": "Restarting EC2 Instance for maintenance",
      "MaxConcurrency": "1",
      "MaxErrors": "1",
      "Name": "My-Automation-Example-Task",
      "Priority": 0,
      "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
      "Targets": [
        {
          "Key": "WindowTargetIds",
          "Values": [
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
          ]
        }
      ]
    }
  ],
}
```



```

    {
      "WindowId": "mw-06cf17cbefEXAMPLE",
      "WindowTaskId": "1943dee0-0a17-4978-9bf4-3cc2fEXAMPLE",
      "TaskArn": "AWS-DisableS3BucketPublicReadWrite",
      "TaskParameters": {},
      "Type": "AUTOMATION",
      "Description": "Automation task to disable read/write access on public
S3 buckets",
      "MaxConcurrency": "10",
      "MaxErrors": "5",
      "Name": "My-Disable-S3-Public-Read-Write-Access-Automation-Task",
      "Priority": 0,
      "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
      "Targets": [
        {
          "Key": "WindowTargetIds",
          "Values": [
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
          ]
        }
      ]
    }
  ]
}

```

Contoh 2: Untuk membuat daftar semua tugas untuk jendela pemeliharaan yang memanggil dokumen `RunPowerShellScript` perintah AWS-

`describe-maintenance-window-tasks` Contoh berikut mencantumkan semua tugas untuk jendela pemeliharaan tertentu yang memanggil dokumen `AWS-RunPowerShellScript` perintah.

```

aws ssm describe-maintenance-window-tasks \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"

```

Output:

```

{
  "Tasks": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",

```

```

    "TaskArn": "AWS-RunPowerShellScript",
    "Type": "RUN_COMMAND",
    "Targets": [
      {
        "Key": "WindowTargetIds",
        "Values": [
          "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
        ]
      }
    ],
    "TaskParameters": {},
    "Priority": 1,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
    ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Name": "MyTask"
  }
]
}

```

Contoh 3: Untuk membuat daftar semua tugas untuk jendela pemeliharaan yang memiliki Prioritas 3

`describe-maintenance-window-tasks` Contoh berikut mencantumkan semua tugas untuk jendela pemeliharaan tertentu yang memiliki Priority a3.

```

aws ssm describe-maintenance-window-tasks \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=Priority,Values=3"

```

Output:

```

{
  "Tasks": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
      "TaskArn": "AWS-RunPowerShellScript",
      "Type": "RUN_COMMAND",
      "Targets": [
        {
          "Key": "WindowTargetIds",

```

```

        "Values": [
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
        ]
    },
    ],
    "TaskParameters": {},
    "Priority": 3,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Name": "MyRunCommandTask"
},
{
    "WindowId": "mw-ab12cd34eEXAMPLE",
    "WindowTaskId": "ee45feff-ad65-4a6c-b478-5cab8EXAMPLE",
    "TaskArn": "AWS-RestartEC2Instance",
    "Type": "AUTOMATION",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
            ]
        }
    ],
    "TaskParameters": {},
    "Priority": 3,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxConcurrency": "10",
    "MaxErrors": "5",
    "Name": "My-Automation-Task",
    "Description": "A description for my Automation task"
}
]
}

```

Contoh 4: Untuk membuat daftar semua tugas untuk jendela pemeliharaan yang memiliki Prioritas 1 dan menggunakan Run Command

`describe-maintenance-window-tasks` Contoh ini mencantumkan semua tugas untuk jendela pemeliharaan tertentu yang memiliki Priority of 1 dan penggunaan `Run Command`.

```
aws ssm describe-maintenance-window-tasks \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"
```

Output:

```
{
  "Tasks": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
      "TaskArn": "AWS-RunPowerShellScript",
      "Type": "RUN_COMMAND",
      "Targets": [
        {
          "Key": "WindowTargetIds",
          "Values": [
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
          ]
        }
      ],
      "TaskParameters": {},
      "Priority": 1,
      "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
      "MaxConcurrency": "1",
      "MaxErrors": "1",
      "Name": "MyRunCommandTask"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat informasi tentang jendela pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowTasks](#) di Referensi AWS CLI Perintah.

## describe-maintenance-windows-for-target

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-windows-for-target`.

## AWS CLI

Untuk mencantumkan semua jendela pemeliharaan yang terkait dengan instance tertentu `describe-maintenance-windows-for-target` Contoh berikut mencantumkan jendela pemeliharaan yang memiliki target atau tugas yang terkait dengan instance yang ditentukan.

```
aws ssm describe-maintenance-windows-for-target \
  --targets Key=InstanceIds,Values=i-1234567890EXAMPLE \
  --resource-type INSTANCE
```

Output:

```
{
  "WindowIdentities": [
    {
      "WindowId": "mw-0c5ed765acEXAMPLE",
      "Name": "My-First-Maintenance-Window"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Pemeliharaan Windows \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindowsForTarget](#) di Referensi AWS CLI Perintah.

## **describe-maintenance-windows**

Contoh kode berikut menunjukkan cara menggunakan `describe-maintenance-windows`.

### AWS CLI

Contoh 1: Untuk daftar semua jendela pemeliharaan

`describe-maintenance-windows` Contoh berikut mencantumkan semua jendela pemeliharaan di AWS akun Anda di Wilayah saat ini.

```
aws ssm describe-maintenance-windows
```

Output:

```
{
  "WindowIdentities": [
    {
      "WindowId": "mw-0ecb1226ddEXAMPLE",
      "Name": "MyMaintenanceWindow-1",
      "Enabled": true,
      "Duration": 2,
      "Cutoff": 1,
      "Schedule": "rate(180 minutes)",
      "NextExecutionTime": "2020-02-12T23:19:20.596Z"
    },
    {
      "WindowId": "mw-03eb9db428EXAMPLE",
      "Name": "MyMaintenanceWindow-2",
      "Enabled": true,
      "Duration": 3,
      "Cutoff": 1,
      "Schedule": "rate(7 days)",
      "NextExecutionTime": "2020-02-17T23:22:00.956Z"
    }
  ]
}
```

Contoh 2: Untuk mencantumkan semua jendela pemeliharaan yang diaktifkan

`describe-maintenance-windows` Contoh berikut mencantumkan semua jendela pemeliharaan yang diaktifkan.

```
aws ssm describe-maintenance-windows \
  --filters "Key=Enabled,Values=true"
```

Contoh 3: Untuk mencantumkan jendela pemeliharaan yang cocok dengan nama tertentu

`describe-maintenance-windows` Contoh ini mencantumkan semua jendela pemeliharaan dengan nama yang ditentukan.

```
aws ssm describe-maintenance-windows \
  --filters "Key=Name,Values=MyMaintenanceWindow"
```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Pemeliharaan Windows \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeMaintenanceWindows](#) di Referensi AWS CLI Perintah.

## describe-ops-items

Contoh kode berikut menunjukkan cara menggunakan `describe-ops-items`.

### AWS CLI

Untuk membuat daftar satu set OpsItems

`describe-ops-items` Contoh berikut menampilkan daftar semua yang terbuka OpsItems di AWS akun Anda.

```
aws ssm describe-ops-items \
  --ops-item-filters "Key=Status,Values=Open,Operator=Equal"
```

Output:

```
{
  "OpsItemSummaries": [
    {
      "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
      "CreatedTime": "2020-03-14T17:02:46.375000-07:00",
      "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
      "LastModifiedTime": "2020-03-14T17:02:46.375000-07:00",
      "Source": "SSM",
      "Status": "Open",
      "OpsItemId": "oi-7cfc5EXAMPLE",
      "Title": "SSM Maintenance Window execution failed",
      "OperationalData": {
        "/aws/dedup": {
          "Value": "{\"dedupString\": \"SSMOpsItems-SSM-maintenance-window-
execution-failed\"}",
          "Type": "SearchableString"
        },
        "/aws/resources": {
          "Value": "[{\"arn\": \"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-034093d322EXAMPLE\"}]",
          "Type": "SearchableString"
        }
      }
    }
  ]
}
```

```

    },
    "Category": "Availability",
    "Severity": "3"
  },
  {
    "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
    "CreatedTime": "2020-02-26T11:43:15.426000-08:00",
    "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
    "LastModifiedTime": "2020-02-26T11:43:15.426000-08:00",
    "Source": "EC2",
    "Status": "Open",
    "OpsItemId": "oi-6f966EXAMPLE",
    "Title": "EC2 instance stopped",
    "OperationalData": {
      "/aws/automations": {
        "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",
\"automationId\": \"AWS-RestartEC2Instance\" } ]",
        "Type": "SearchableString"
      },
      "/aws/dedup": {
        "Value": "{\"dedupString\": \"SSMOpsItems-EC2-instance-stopped
\"}",
        "Type": "SearchableString"
      },
      "/aws/resources": {
        "Value": "[{\"arn\": \"arn:aws:ec2:us-
east-2:111222333444:instance/i-0beccfbc02EXAMPLE\"}]",
        "Type": "SearchableString"
      }
    },
    "Category": "Availability",
    "Severity": "3"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan OpsItems](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeOpsItems](#) di Referensi AWS CLI Perintah.



## describe-parameters

Contoh kode berikut menunjukkan cara menggunakan `describe-parameters`.

### AWS CLI

Contoh 1: Untuk daftar semua parameter

`describe-parameters` Contoh berikut mencantumkan semua parameter di AWS akun saat ini dan Wilayah.

```
aws ssm describe-parameters
```

Output:

```
{
  "Parameters": [
    {
      "Name": "MySecureStringParameter",
      "Type": "SecureString",
      "KeyId": "alias/aws/ssm",
      "LastModifiedDate": 1582155479.205,
      "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/Admin/Richard-Roe-Managed",
      "Description": "This is a SecureString parameter",
      "Version": 2,
      "Tier": "Advanced",
      "Policies": [
        {
          "PolicyText": "{\"Type\":\"Expiration\",\"Version\":\"1.0\",
\\Attributes\":{\"Timestamp\":\"2020-07-07T22:30:00Z\"}}",
          "PolicyType": "Expiration",
          "PolicyStatus": "Pending"
        },
        {
          "PolicyText": "{\"Type\":\"ExpirationNotification\",\"Version\":
\\\"1.0\\\",\\Attributes\":{\"Before\":\"12\",\"Unit\":\"Hours\"}}",
          "PolicyType": "ExpirationNotification",
          "PolicyStatus": "Pending"
        }
      ]
    },
    {
```

```

    "Name": "MyStringListParameter",
    "Type": "StringList",
    "LastModifiedDate": 1582154764.222,
    "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
    "Description": "This is a StringList parameter",
    "Version": 1,
    "Tier": "Standard",
    "Policies": []
  },
  {
    "Name": "MyStringParameter",
    "Type": "String",
    "LastModifiedDate": 1582154711.976,
    "LastModifiedUser": "arn:aws:iam::111222333444:user/Alejandro-Rosalez",
    "Description": "This is a String parameter",
    "Version": 1,
    "Tier": "Standard",
    "Policies": []
  },
  {
    "Name": "latestAmi",
    "Type": "String",
    "LastModifiedDate": 1580862415.521,
    "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/lambda-ssm-
role/Automation-UpdateSSM-Param",
    "Version": 3,
    "Tier": "Standard",
    "Policies": []
  }
]
}

```

Contoh 2: Untuk mencantumkan semua parameter yang cocok dengan metadata tertentu

`describe-parameters` Contoh ini mencantumkan semua parameter yang cocok dengan filter.

`aws ssm describe-parameters --filter "Kunci = Jenis, Nilai =" StringList`

Output:

```

{
  "Parameters": [
    {

```

```

        "Name": "MyStringListParameter",
        "Type": "StringList",
        "LastModifiedDate": 1582154764.222,
        "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
        "Description": "This is a StringList parameter",
        "Version": 1,
        "Tier": "Standard",
        "Policies": []
    }
]
}

```

Untuk informasi selengkapnya, lihat [Mencari Parameter Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeParameters](#) di Referensi AWS CLI Perintah.

## describe-patch-baselines

Contoh kode berikut menunjukkan cara menggunakan `describe-patch-baselines`.

### AWS CLI

Contoh 1: Untuk membuat daftar semua baseline patch

`describe-patch-baselines` Contoh berikut memuat ulang detail untuk semua baseline patch di akun Anda di Wilayah saat ini.

```
aws ssm describe-patch-baselines
```

Output:

```

{
  "BaselineIdentities": [
    {
      "BaselineName": "AWS-SuseDefaultPatchBaseline",
      "DefaultBaseline": true,
      "BaselineDescription": "Default Patch Baseline for Suse Provided by
AWS.",
      "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0123fdb36e334a3b2",

```

```

        "OperatingSystem": "SUSE"
    },
    {
        "BaselineName": "AWS-DefaultPatchBaseline",
        "DefaultBaseline": false,
        "BaselineDescription": "Default Patch Baseline Provided by AWS.",
        "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed",
        "OperatingSystem": "WINDOWS"
    },
    ...
    {
        "BaselineName": "MyWindowsPatchBaseline",
        "DefaultBaseline": true,
        "BaselineDescription": "My patch baseline for EC2 instances for Windows
Server",
        "BaselineId": "pb-0ad00e0dd7EXAMPLE",
        "OperatingSystem": "WINDOWS"
    }
]
}

```

Contoh 2: Untuk membuat daftar semua baseline patch yang disediakan oleh AWS

`describe-patch-baselines` Contoh berikut mencantumkan semua baseline patch yang disediakan oleh AWS

```

aws ssm describe-patch-baselines \
  --filters "Key=OWNER,Values=[AWS]"

```

Contoh 3: Untuk membuat daftar semua baseline patch yang Anda miliki

`describe-patch-baselines` Contoh berikut mencantumkan semua baseline patch kustom yang dibuat di akun Anda di Wilayah saat ini.

```

aws ssm describe-patch-baselines \
  --filters "Key=OWNER,Values=[Self]"

```

Untuk informasi [selengkapnya, lihat Tentang Garis Dasar Patch Standar dan Kustom di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [DescribePatchBaselines](#) di Referensi AWS CLI Perintah.

## describe-patch-group-state

Contoh kode berikut menunjukkan cara menggunakan `describe-patch-group-state`.

### AWS CLI

Untuk mendapatkan status grup patch

`describe-patch-group-state` Contoh berikut mengambil ringkasan kepatuhan patch tingkat tinggi untuk grup patch.

```
aws ssm describe-patch-group-state \
  --patch-group "Production"
```

Output:

```
{
  "Instances": 21,
  "InstancesWithCriticalNonCompliantPatches": 1,
  "InstancesWithFailedPatches": 2,
  "InstancesWithInstalledOtherPatches": 3,
  "InstancesWithInstalledPatches": 21,
  "InstancesWithInstalledPendingRebootPatches": 2,
  "InstancesWithInstalledRejectedPatches": 1,
  "InstancesWithMissingPatches": 3,
  "InstancesWithNotApplicablePatches": 4,
  "InstancesWithOtherNonCompliantPatches": 1,
  "InstancesWithSecurityNonCompliantPatches": 1,
  "InstancesWithUnreportedNotApplicablePatches": 2
}
```

Untuk informasi selengkapnya, lihat Tentang grup tambalan < <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-patchgroups.html> > dan [Memahami nilai status kepatuhan patch di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [DescribePatchGroupState](#) di Referensi AWS CLI Perintah.

## describe-patch-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-patch-groups`.

## AWS CLI

Untuk menampilkan pendaftaran grup tambalan

`describe-patch-groups` Contoh berikut mencantumkan pendaftaran grup patch.

```
aws ssm describe-patch-groups
```

Output:

```
{
  "Mappings": [
    {
      "PatchGroup": "Production",
      "BaselineIdentity": {
        "BaselineId": "pb-0123456789abcdef0",
        "BaselineName": "ProdPatching",
        "OperatingSystem": "WINDOWS",
        "BaselineDescription": "Patches for Production",
        "DefaultBaseline": false
      }
    },
    {
      "PatchGroup": "Development",
      "BaselineIdentity": {
        "BaselineId": "pb-0713accee01234567",
        "BaselineName": "DevPatching",
        "OperatingSystem": "WINDOWS",
        "BaselineDescription": "Patches for Development",
        "DefaultBaseline": true
      }
    },
    ...
  ]
}
```

Untuk informasi selengkapnya, lihat [Membuat Grup Patch < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>](https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html) dan [Tambahkan Grup Patch ke Baseline Patch di](#) Panduan Pengguna Systems Manager AWS .

- Untuk detail API, lihat [DescribePatchGroups](#) di Referensi AWS CLI Perintah.

## describe-patch-properties

Contoh kode berikut menunjukkan cara menggunakan `describe-patch-properties`.

### AWS CLI

Untuk membuat daftar ketersediaan patch Amazon Linux

`describe-patch-properties` Contoh berikut menampilkan daftar produk Amazon Linux yang tambalannya tersedia di AWS akun Anda.

```
aws ssm describe-patch-properties \
  --operating-system AMAZON_LINUX \
  --property PRODUCT
```

### Output:

```
{
  "Properties": [
    {
      "Name": "AmazonLinux2012.03"
    },
    {
      "Name": "AmazonLinux2012.09"
    },
    {
      "Name": "AmazonLinux2013.03"
    },
    {
      "Name": "AmazonLinux2013.09"
    },
    {
      "Name": "AmazonLinux2014.03"
    },
    {
      "Name": "AmazonLinux2014.09"
    },
    {
      "Name": "AmazonLinux2015.03"
    },
    {
      "Name": "AmazonLinux2015.09"
    }
  ]
}
```

```
{
  "Name": "AmazonLinux2016.03"
},
{
  "Name": "AmazonLinux2016.09"
},
{
  "Name": "AmazonLinux2017.03"
},
{
  "Name": "AmazonLinux2017.09"
},
{
  "Name": "AmazonLinux2018.03"
}
]
```

Untuk informasi selengkapnya, lihat [Tentang Patch Baseline](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribePatchProperties](#) di Referensi AWS CLI Perintah.

## describe-sessions

Contoh kode berikut menunjukkan cara menggunakan `describe-sessions`.

### AWS CLI

Contoh 1: Untuk mencantumkan semua sesi Manajer Sesi yang aktif

`describe-sessions` Contoh ini mengambil daftar sesi aktif yang dibuat terbaru (baik sesi terhubung maupun terputus) selama 30 hari terakhir yang dimulai oleh pengguna yang ditentukan. Perintah ini hanya mengembalikan hasil untuk koneksi ke target yang dimulai menggunakan Session Manager. Itu tidak mencantumkan koneksi yang dibuat melalui cara lain, seperti Remote Desktop Connections atau SSH.

```
aws ssm describe-sessions \
  --state "Active" \
  --filters "key=Owner,value=arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-Rodriguez"
```



## Output:

```
{
  "Sessions": [
    {
      "SessionId": "John-07a16060613c408b5",
      "Target": "i-1234567890abcdef0",
      "Status": "Connected",
      "StartDate": 1550676938.352,
      "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-Rodriguez",
      "OutputUrl": {}
    },
    {
      "SessionId": "John-01edf534b8b56e8eb",
      "Target": "i-9876543210abcdef0",
      "Status": "Connected",
      "StartDate": 1550676842.194,
      "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-Rodriguez",
      "OutputUrl": {}
    }
  ]
}
```

Contoh 2: Untuk mencantumkan semua sesi Manajer Sesi yang dihentikan

`describe-sessions` Contoh ini mengambil daftar sesi yang terakhir dihentikan dari 30 hari terakhir untuk semua pengguna.

```
aws ssm describe-sessions \
  --state "History"
```

## Output:

```
{
  "Sessions": [
    {
      "SessionId": "Mary-Major-0022b1eb2b0d9e3bd",
      "Target": "i-1234567890abcdef0",
      "Status": "Terminated",
      "StartDate": 1550520701.256,
```

```

    "EndDate": 1550521931.563,
    "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Mary-
Major"
  },
  {
    "SessionId": "Jane-Roe-0db53f487931ed9d4",
    "Target": "i-9876543210abcdef0",
    "Status": "Terminated",
    "StartDate": 1550161369.149,
    "EndDate": 1550162580.329,
    "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Jane-Roe"
  },
  ...
],
"NextToken": "--token string truncated--"
}

```

Untuk informasi selengkapnya, lihat [Melihat Riwayat Sesi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [DescribeSessions](#) di Referensi AWS CLI Perintah.

## disassociate-ops-item-related-item

Contoh kode berikut menunjukkan cara menggunakan `disassociate-ops-item-related-item`.

### AWS CLI

Untuk menghapus asosiasi item terkait

`disassociate-ops-item-related-item` Contoh berikut menghapus hubungan antara `OpsItem` dan item terkait.

```

aws ssm disassociate-ops-item-related-item \
  --ops-item-id "oi-f99f2EXAMPLE" \
  --association-id "e2036148-cccb-490e-ac2a-390e5EXAMPLE"

```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan insiden Manajer Insiden OpsCenter di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [DisassociateOpsItemRelatedItem](#) di Referensi AWS CLI Perintah.

## get-automation-execution

Contoh kode berikut menunjukkan cara menggunakan `get-automation-execution`.

### AWS CLI

Untuk menampilkan detail tentang eksekusi otomatisasi

`get-automation-execution` Contoh berikut menampilkan informasi rinci tentang eksekusi Otomasi.

```
aws ssm get-automation-execution \  
--automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Output:

```
{  
  "AutomationExecution": {  
    "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",  
    "DocumentName": "AWS-StartEC2Instance",  
    "DocumentVersion": "1",  
    "ExecutionStartTime": 1583737233.748,  
    "ExecutionEndTime": 1583737234.719,  
    "AutomationExecutionStatus": "Success",  
    "StepExecutions": [  
      {  
        "StepName": "startInstances",  
        "Action": "aws:changeInstanceState",  
        "ExecutionStartTime": 1583737234.134,  
        "ExecutionEndTime": 1583737234.672,  
        "StepStatus": "Success",  
        "Inputs": {  
          "DesiredState": "\"running\"",  
          "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"  
        },  
        "Outputs": {  
          "InstanceStates": [  
            "running"  
          ]  
        },  
        "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",  
        "OverriddenParameters": {}  
      }  
    ]  
  }  
}
```

```
    ],
    "StepExecutionsTruncated": false,
    "Parameters": {
      "AutomationAssumeRole": [
        ""
      ],
      "InstanceId": [
        "i-0cb99161f6EXAMPLE"
      ]
    },
    "Outputs": {},
    "Mode": "Auto",
    "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/
OrchestrationService",
    "Targets": [],
    "ResolvedTargets": {
      "ParameterValues": [],
      "Truncated": false
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Panduan: Menambal AMI Linux \(AWS CLI\)](#) di [Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [GetAutomationExecution](#) di Referensi AWS CLI Perintah.

## get-calendar-state

Contoh kode berikut menunjukkan cara menggunakan `get-calendar-state`.

### AWS CLI

Contoh 1: Untuk mendapatkan status kalender perubahan saat ini

`get-calendar-state` Contoh ini mengembalikan keadaan kalender pada saat ini. Karena contoh tidak menentukan waktu, status kalender saat ini dilaporkan.

```
aws ssm get-calendar-state \
  --calendar-names "MyCalendar"
```

Output:

```
{
  "State": "OPEN",
  "AtTime": "2020-02-19T22:28:51Z",
  "NextTransitionTime": "2020-02-24T21:15:19Z"
}
```

Contoh 2: Untuk mendapatkan status kalender perubahan pada waktu yang ditentukan

`get-calendar-state` Contoh ini mengembalikan keadaan kalender pada waktu yang ditentukan.

```
aws ssm get-calendar-state \
  --calendar-names "MyCalendar" \
  --at-time "2020-07-19T21:15:19Z"
```

Output:

```
{
  "State": "CLOSED",
  "AtTime": "2020-07-19T21:15:19Z"
}
```

Untuk informasi selengkapnya, lihat [Mendapatkan Status Kalender Perubahan](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetCalendarState](#) di Referensi AWS CLI Perintah.

## get-command-invocation

Contoh kode berikut menunjukkan cara menggunakan `get-command-invocation`.

### AWS CLI

Untuk menampilkan rincian pemanggilan perintah

`get-command-invocation` Contoh berikut mencantumkan semua pemanggilan perintah yang ditentukan pada instance tertentu.

```
aws ssm get-command-invocation \
  --command-id "ef7fd8-9b57-4151-a15c-db9a12345678" \
```

```
--instance-id "i-1234567890abcdef0"
```

### Output:

```
{
  "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
  "InstanceId": "i-1234567890abcdef0",
  "Comment": "b48291dd-ba76-43e0-b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
  "DocumentName": "AWS-UpdateSSMAgent",
  "DocumentVersion": "",
  "PluginName": "aws:updateSsmAgent",
  "ResponseCode": 0,
  "ExecutionStartDateTime": "2020-02-19T18:18:03.419Z",
  "ExecutionElapsedTime": "PT0.091S",
  "ExecutionEndDateTime": "2020-02-19T18:18:03.419Z",
  "Status": "Success",
  "StatusDetails": "Success",
  "StandardOutputContent": "Updating amazon-ssm-agent from 2.3.842.0 to latest\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been installed,\nupdate skipped\n",
  "StandardOutputUrl": "",
  "StandardErrorContent": "",
  "StandardErrorUrl": "",
  "CloudWatchOutputConfig": {
    "CloudWatchLogGroupName": "",
    "CloudWatchOutputEnabled": false
  }
}
```

Untuk informasi selengkapnya, lihat [Memahami Status Perintah](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetCommandInvocation](#) di Referensi AWS CLI Perintah.

## get-connection-status

Contoh kode berikut menunjukkan cara menggunakan `get-connection-status`.

### AWS CLI

Untuk menampilkan status koneksi dari instance terkelola

`get-connection-status` Contoh ini mengembalikan status koneksi dari instance terkelola tertentu.

```
aws ssm get-connection-status \  
  --target i-1234567890abcdef0
```

Output:

```
{  
  "Target": "i-1234567890abcdef0",  
  "Status": "connected"  
}
```

- Untuk detail API, lihat [GetConnectionStatus](#) di Referensi AWS CLI Perintah.

## `get-default-patch-baseline`

Contoh kode berikut menunjukkan cara menggunakan `get-default-patch-baseline`.

### AWS CLI

Contoh 1: Untuk menampilkan baseline patch Windows default

`get-default-patch-baseline` Contoh berikut mengambil rincian untuk baseline patch default untuk Windows Server.

```
aws ssm get-default-patch-baseline
```

Output:

```
{  
  "BaselineId": "pb-0713accee01612345",  
  "OperatingSystem": "WINDOWS"  
}
```

Contoh 2: Untuk menampilkan baseline patch default untuk Amazon Linux

`get-default-patch-baseline` Contoh berikut mengambil detail untuk baseline patch default untuk Amazon Linux.

```
aws ssm get-default-patch-baseline \  
  --target i-1234567890abcdef0
```

```
--operating-system AMAZON_LINUX
```

Output:

```
{
  "BaselineId": "pb-047c6eb9c8fc12345",
  "OperatingSystem": "AMAZON_LINUX"
}
```

Untuk informasi selengkapnya, lihat [Tentang Garis Dasar Patch Standar dan Kustom < https:// docs.aws.amazon.com/systems-manager/latest/userguide/ sysman-patch-baselines .html>](https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-baselines.html) dan [Tetapkan Garis Dasar Patch yang Ada sebagai Default di Panduan Pengguna Systems Manager.AWS](#)

- Untuk detail API, lihat [GetDefaultPatchBaseline](#) di Referensi AWS CLI Perintah.

## get-deployable-patch-snapshot-for-instance

Contoh kode berikut menunjukkan cara menggunakan `get-deployable-patch-snapshot-for-instance`.

AWS CLI

Untuk mengambil snapshot saat ini untuk baseline patch, sebuah instance menggunakan

`get-deployable-patch-snapshot-for-instance` Contoh berikut mengambil detail untuk snapshot saat ini untuk baseline patch tertentu yang digunakan oleh sebuah instance. Perintah ini harus dijalankan dari instance menggunakan kredensi instance. Untuk memastikannya menggunakan kredensyal instance, jalankan `aws configure` dan tentukan hanya Wilayah instance Anda. Biarkan `Secret Key` bidang `Access Key` dan kosong.

Tip: Gunakan `uuidgen` untuk menghasilkan `snapshot-id`.

```
aws ssm get-deployable-patch-snapshot-for-instance \
  --instance-id "i-1234567890abcdef0" \
  --snapshot-id "521c3536-930c-4aa9-950e-01234567abcd"
```

Output:

```
{
```



```

    "InstanceId": "i-1234567890abcdef0",
    "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",
    "Product": "AmazonLinux2018.03",
    "SnapshotDownloadUrl": "https://patch-baseline-snapshot-us-east-1.s3.amazonaws.com/ed85194ef27214f5984f28b4d664d14f7313568fea7d4b6ac6c10ad1f729d7e7-773304212436/AMAZON_LINUX-521c3536-930c-4aa9-950e-01234567abcd?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20190215T164031Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-Credential=AKIAJ5C56P35AEBRX2Q0%2F20190215%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signature=efaaaf6e3878e77f48a6697e015efdbda9c426b09c5822055075c062f6ad2149"
  }

```

Untuk informasi selengkapnya, lihat [Nama parameter: ID Snapshot](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetDeployablePatchSnapshotForInstance](#) di Referensi AWS CLI Perintah.

## get-document

Contoh kode berikut menunjukkan cara menggunakan `get-document`.

### AWS CLI

Untuk mendapatkan konten dokumen

`get-document` Contoh berikut menampilkan konten dokumen Systems Manager.

```
aws ssm get-document \
  --name "AWS-RunShellScript"
```

Output:

```

{
  "Name": "AWS-RunShellScript",
  "DocumentVersion": "1",
  "Status": "Active",
  "Content": "{\n  \"schemaVersion\": \"1.2\", \n  \"description\": \"Run a\nshell script or specify the commands to run.\", \n  \"parameters\": {\n\n    \"commands\": {\n      \"type\": \"StringList\", \n      \"description\": \"(Required) Specify a shell script or a command to run.\", \n      \"minItems\": 1, \n      \"displayType\": \"textarea\" \n    }, \n    \"workingDirectory\": {\n      \"type\": \"String\", \n      \"default\": \"\"

```

```

\":"\",\\n
  "description\":"(Optional) The path to the working
  directory on your instance.\\n
  "maxChars\":"4096\\n
  },\\n
  "executionTimeout\":{\\n
    "type\":"String\\n
    "default
\\":"3600\\n
    "description\":"(Optional) The time in seconds for a
    command to complete before it is considered to have failed. Default is 3600 (1
    hour). Maximum is 172800 (48 hours).\\n
    "allowedPattern\":"([1-9]
[0-9]{0,4})|(1[0-6][0-9]{4})|(17[0-1][0-9]{3})|(172[0-7][0-9]{2})|(172800)\\n
    }\\n
  },\\n
  "runtimeConfig\":{\\n
    "aws:runShellScript\":{\\n
      "properties\":[\\n
        {\\n
          "id\":"
\\":"0.aws:runShellScript\\n
          "runCommand\":"{{ commands }}\\n
          "workingDirectory\":"{{ workingDirectory }}\\n
          "timeoutSeconds\":"{{ executionTimeout }}\\n
        }\\n
      ]\\n
    }\\n
  }\\n
},
"DocumentType": "Command",
"DocumentFormat": "JSON"
}

```

Untuk informasi selengkapnya, lihat [Dokumen AWS Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetDocument](#) di Referensi AWS CLI Perintah.

## get-inventory-schema

Contoh kode berikut menunjukkan cara menggunakan `get-inventory-schema`.

### AWS CLI

Untuk melihat skema inventaris Anda

Contoh ini mengembalikan daftar nama jenis inventaris untuk akun.

Perintah:

```
aws ssm get-inventory-schema
```

Output:

```

{
  "Schemas": [
    {
      "TypeName": "AWS:AWSComponent",
      "Version": "1.0",

```

```
    "Attributes": [  
      {  
        "Name": "Name",  
        "DataType": "STRING"  
      },  
      {  
        "Name": "ApplicationType",  
        "DataType": "STRING"  
      },  
      {  
        "Name": "Publisher",  
        "DataType": "STRING"  
      },  
      {  
        "Name": "Version",  
        "DataType": "STRING"  
      },  
      {  
        "Name": "InstalledTime",  
        "DataType": "STRING"  
      },  
      {  
        "Name": "Architecture",  
        "DataType": "STRING"  
      },  
      {  
        "Name": "URL",  
        "DataType": "STRING"  
      }  
    ],  
    ...  
  ],  
  "NextToken": "--token string truncated--"  
}
```

Untuk melihat skema inventaris untuk jenis inventaris tertentu

Contoh ini mengembalikan skema inventaris untuk tipe inventaris AWS:AWS Komponen.

Perintah:

```
aws ssm get-inventory-schema --type-name "AWS:AWSComponent"
```

- Untuk detail API, lihat [GetInventorySchema](#) di Referensi AWS CLI Perintah.

## get-inventory

Contoh kode berikut menunjukkan cara menggunakan `get-inventory`.

### AWS CLI

Untuk melihat inventaris Anda

Contoh ini mendapatkan metadata kustom untuk inventaris Anda.

Perintah:

```
aws ssm get-inventory
```

Output:

```
{
  "Entities": [
    {
      "Data": {
        "AWS:InstanceInformation": {
          "Content": [
            {
              "ComputerName": "ip-172-31-44-222.us-
west-2.compute.internal",
              "InstanceId": "i-0cb2b964d3e14fd9f",
              "IpAddress": "172.31.44.222",
              "AgentType": "amazon-ssm-agent",
              "ResourceType": "EC2Instance",
              "AgentVersion": "2.0.672.0",
              "PlatformVersion": "2016.09",
              "PlatformName": "Amazon Linux AMI",
              "PlatformType": "Linux"
            }
          ],
          "TypeName": "AWS:InstanceInformation",
          "SchemaVersion": "1.0",
          "CaptureTime": "2017-02-20T18:03:58Z"
        }
      }
    }
  ],
}
```

```

      "Id": "i-0cb2b964d3e14fd9f"
    }
  ]
}

```

- Untuk detail API, lihat [GetInventory](#) di Referensi AWS CLI Perintah.

## get-maintenance-window-execution-task-invocation

Contoh kode berikut menunjukkan cara menggunakan `get-maintenance-window-execution-task-invocation`.

### AWS CLI

Untuk mendapatkan informasi tentang pemanggilan tugas jendela pemeliharaan

`get-maintenance-window-execution-task-invocation` Contoh berikut mencantumkan informasi tentang pemanggilan tugas tertentu yang merupakan bagian dari eksekusi jendela pemeliharaan yang ditentukan.

```

aws ssm get-maintenance-window-execution-task-invocation \
  --window-execution-id "bc494bfa-e63b-49f6-8ad1-aa9f2EXAMPLE" \
  --task-id "96f2ad59-97e3-461d-a63d-40c8aEXAMPLE" \
  --invocation-id "a5273e2c-d2c6-4880-b3e1-5e550EXAMPLE"

```

### Output:

```

{
  "Status": "SUCCESS",
  "Parameters": "{\"comment\":\"\", \"documentName\":\"AWS-RunPowerShellScript\", \"instanceIds\": [\"i-1234567890EXAMPLE\"], \"maxConcurrency\": \"1\", \"maxErrors\": \"1\", \"parameters\": {\"executionTimeout\": [\"3600\"], \"workingDirectory\": [\"\"], \"commands\": [\"echo Hello\"]}, \"timeoutSeconds\": 600}\",
  "ExecutionId": "03b6baa0-5460-4e15-83f2-ea685EXAMPLE",
  "InvocationId": "a5273e2c-d2c6-4880-b3e1-5e550EXAMPLE",
  "StartTime": 1549998326.421,
  "TaskType": "RUN_COMMAND",
  "EndTime": 1550001931.784,
  "WindowExecutionId": "bc494bfa-e63b-49f6-8ad1-aa9f2EXAMPLE",
  "StatusDetails": "Failed",
  "TaskExecutionId": "96f2ad59-97e3-461d-a63d-40c8aEXAMPLE"
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Tugas dan Eksekusi Tugas \(AWS CLI\)](#) di Panduan Pengguna Systems AWS Manager.

- Untuk detail API, lihat [GetMaintenanceWindowExecutionTaskInvocation](#) di Referensi AWS CLI Perintah.

## get-maintenance-window-execution-task

Contoh kode berikut menunjukkan cara menggunakan `get-maintenance-window-execution-task`.

### AWS CLI

Untuk mendapatkan informasi tentang eksekusi tugas jendela pemeliharaan

`get-maintenance-window-execution-task` Contoh berikut mencantumkan informasi tentang tugas yang merupakan bagian dari eksekusi jendela pemeliharaan yang ditentukan.

```
aws ssm get-maintenance-window-execution-task \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE" \
  --task-id "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
```

### Output:

```
{
  "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
  "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE",
  "TaskArn": "AWS-RunPatchBaseline",
  "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
  "Type": "RUN_COMMAND",
  "TaskParameters": [
    {
      "BaselineOverride": {
        "Values": [
          ""
        ]
      },
      "InstallOverrideList": {
        "Values": [
```

```

        ""
    ],
    "Operation": {
        "Values": [
            "Scan"
        ]
    },
    "RebootOption": {
        "Values": [
            "RebootIfNeeded"
        ]
    },
    "SnapshotId": {
        "Values": [
            "{{ aws:ORCHESTRATION_ID }}"
        ]
    },
    "aws:InstanceId": {
        "Values": [
            "i-02573cafcfEXAMPLE",
            "i-0471e04240EXAMPLE",
            "i-07782c72faEXAMPLE"
        ]
    }
}
],
"Priority": 1,
"MaxConcurrency": "1",
"MaxErrors": "3",
"Status": "SUCCESS",
"StartTime": "2021-08-04T11:45:35.088000-07:00",
"EndTime": "2021-08-04T11:53:09.079000-07:00"
}

```

Untuk informasi selengkapnya, lihat [Melihat informasi tentang tugas dan eksekusi tugas \(AWS CLI\)](#) di Panduan Pengguna Systems AWS Manager.

- Untuk detail API, lihat [GetMaintenanceWindowExecutionTask](#) di Referensi AWS CLI Perintah.

## get-maintenance-window-execution

Contoh kode berikut menunjukkan cara menggunakan `get-maintenance-window-execution`.

## AWS CLI

Untuk mendapatkan informasi tentang eksekusi tugas jendela pemeliharaan

`get-maintenance-window-execution` Contoh berikut mencantumkan informasi tentang tugas yang dijalankan sebagai bagian dari eksekusi jendela pemeliharaan yang ditentukan.

```
aws ssm get-maintenance-window-execution \  
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Output:

```
{  
  "Status": "SUCCESS",  
  "TaskIds": [  
    "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"  
  ],  
  "StartTime": 1487692834.595,  
  "EndTime": 1487692835.051,  
  "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",  
}
```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Tugas dan Eksekusi Tugas \(AWS CLI\)](#) di Panduan Pengguna Systems AWS Manager.

- Untuk detail API, lihat [GetMaintenanceWindowExecution](#) di Referensi AWS CLI Perintah.

## `get-maintenance-window-task`

Contoh kode berikut menunjukkan cara menggunakan `get-maintenance-window-task`.

## AWS CLI

Untuk mendapatkan informasi tentang tugas jendela pemeliharaan

`get-maintenance-window-task` Contoh berikut merinci detail tentang tugas jendela pemeliharaan yang ditentukan.

```
aws ssm get-maintenance-window-task \  
  --window-id mw-0c5ed765acEXAMPLE \  
  --window-task-id 0e842a8d-2d44-4886-bb62-af8dcEXAMPLE
```



**Output:**

```
{
  "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
  "MaxErrors": "1",
  "TaskArn": "AWS-RunPowerShellScript",
  "MaxConcurrency": "1",
  "WindowTaskId": "0e842a8d-2d44-4886-bb62-af8dcEXAMPLE",
  "TaskParameters": {},
  "Priority": 1,
  "TaskInvocationParameters": {
    "RunCommand": {
      "Comment": "",
      "TimeoutSeconds": 600,
      "Parameters": {
        "commands": [
          "echo Hello"
        ],
        "executionTimeout": [
          "3600"
        ],
        "workingDirectory": [
          ""
        ]
      }
    }
  },
  "WindowId": "mw-0c5ed765acEXAMPLE",
  "TaskType": "RUN_COMMAND",
  "Targets": [
    {
      "Values": [
        "84c818da-b619-4d3d-9651-946f3EXAMPLE"
      ],
      "Key": "WindowTargetIds"
    }
  ],
  "Name": "ExampleTask"
}
```

Untuk informasi selengkapnya, lihat [Melihat Informasi Tentang Pemeliharaan Windows \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetMaintenanceWindowTask](#) di Referensi AWS CLI Perintah.

## get-maintenance-window

Contoh kode berikut menunjukkan cara menggunakan `get-maintenance-window`.

### AWS CLI

Untuk mendapatkan informasi tentang jendela pemeliharaan

`get-maintenance-window` Contoh berikut mengambil rincian tentang jendela pemeliharaan yang ditentukan.

```
aws ssm get-maintenance-window \  
  --window-id "mw-03eb9db428EXAMPLE"
```

Output:

```
{  
  "AllowUnassociatedTargets": true,  
  "CreateDate": 1515006912.957,  
  "Cutoff": 1,  
  "Duration": 6,  
  "Enabled": true,  
  "ModifiedDate": 2020-01-01T10:04:04.099Z,  
  "Name": "My-Maintenance-Window",  
  "Schedule": "rate(3 days)",  
  "WindowId": "mw-03eb9db428EXAMPLE",  
  "NextExecutionTime": "2020-02-25T00:08:15.099Z"  
}
```

Untuk informasi selengkapnya, lihat [Melihat informasi tentang jendela pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetMaintenanceWindow](#) di Referensi AWS CLI Perintah.

## get-ops-item

Contoh kode berikut menunjukkan cara menggunakan `get-ops-item`.

## AWS CLI

Untuk melihat informasi tentang OpsItem

`get-ops-item` Contoh berikut menampilkan rincian tentang yang ditentukan OpsItem.

```
aws ssm get-ops-item \  
  --ops-item-id oi-0b725EXAMPLE
```

Output:

```
{  
  "OpsItem": {  
    "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/  
fbf77cbe264a33509569f23e4EXAMPLE",  
    "CreatedTime": "2019-12-04T15:52:16.793000-08:00",  
    "Description": "CloudWatch Event Rule SSMOpsItems-EC2-instance-terminated  
was triggered. Your EC2 instance has terminated. See below for more details.",  
    "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/  
fbf77cbe264a33509569f23e4EXAMPLE",  
    "LastModifiedTime": "2019-12-04T15:52:16.793000-08:00",  
    "Notifications": [],  
    "RelatedOpsItems": [],  
    "Status": "Open",  
    "OpsItemId": "oi-0b725EXAMPLE",  
    "Title": "EC2 instance terminated",  
    "Source": "EC2",  
    "OperationalData": {  
      "/aws/automations": {  
        "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",  
\"automationId\": \"AWS>CreateManagedWindowsInstance\" }, { \"automationType\":  
\"AWS:SSM:Automation\", \"automationId\": \"AWS>CreateManagedLinuxInstance\" } ]",  
        "Type": "SearchableString"  
      },  
      "/aws/dedup": {  
        "Value": "{ \"dedupString\": \"SSMOpsItems-EC2-instance-terminated  
\" }",  
        "Type": "SearchableString"  
      },  
      "/aws/resources": {  
        "Value": "[ { \"arn\": \"arn:aws:ec2:us-east-2:111222333444:instance/  
i-05adec7e97EXAMPLE\" } ]",  
        "Type": "SearchableString"  
      }  
    }  
  }  
}
```

```

    },
    "event-time": {
      "Value": "2019-12-04T23:52:16Z",
      "Type": "String"
    },
    "instance-state": {
      "Value": "terminated",
      "Type": "String"
    }
  },
  "Category": "Availability",
  "Severity": "4"
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan OpsItems](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetOpsItem](#) di Referensi AWS CLI Perintah.

## get-ops-summary

Contoh kode berikut menunjukkan cara menggunakan `get-ops-summary`.

### AWS CLI

Untuk melihat ringkasan dari semua OpsItems

`get-ops-summary` Contoh berikut menampilkan ringkasan semua OpsItems di AWS akun Anda.

```
aws ssm get-ops-summary
```

Output:

```

{
  "Entities": [
    {
      "Id": "oi-4309fEXAMPLE",
      "Data": {
        "AWS:OpsItem": {
          "CaptureTime": "2020-02-26T18:58:32.918Z",
          "Content": [
            {

```

```

        "AccountId": "111222333444",
        "Category": "Availability",
        "CreatedBy": "arn:aws:sts::111222333444:assumed-role/
OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
        "CreatedTime": "2020-02-26T19:10:44.149Z",
        "Description": "CloudWatch Event Rule SSM0psItems-EC2-
instance-terminated was triggered. Your EC2 instance has terminated. See below for
more details.",
        "LastModifiedBy": "arn:aws:sts::111222333444:assumed-
role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
        "LastModifiedTime": "2020-02-26T19:10:44.149Z",
        "Notifications": "",
        "OperationalData": "{\"/aws/automations\":
{\"type\": \"SearchableString\", \"value\": \"[ { \\\"automationType\\\": \\
\\\"AWS:SSM:Automation\\\", \\\"automationId\\\": \\\"AWS-CreateManagedWindowsInstance
\\\" }, { \\\"automationType\\\": \\\"AWS:SSM:Automation\\\", \\\"automationId
\\\": \\\"AWS-CreateManagedLinuxInstance\\\" } ]\", \"/aws/resources\":
{\"type\": \"SearchableString\", \"value\": \"[{\\\"arn\\\": \\\"arn:aws:ec2:us-
east-2:111222333444:instance/i-0acbd0800fEXAMPLE\\\"]\", \"/aws/dedup\": {\"type\":
\"SearchableString\", \"value\": \"{\\\"dedupString\\\": \\\"SSM0psItems-EC2-instance-
terminated\\\"}\"}}",
        "OpsItemId": "oi-4309fEXAMPLE",
        "RelatedItems": "",
        "Severity": "3",
        "Source": "EC2",
        "Status": "Open",
        "Title": "EC2 instance terminated"
    }
  ]
}
},
{
  "Id": "oi-bb2a0e6a4541",
  "Data": {
    "AWS:OpsItem": {
      "CaptureTime": "2019-11-26T19:20:06.161Z",
      "Content": [
        {
          "AccountId": "111222333444",
          "Category": "Availability",
          "CreatedBy": "arn:aws:sts::111222333444:assumed-role/
OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
          "CreatedTime": "2019-11-26T20:00:07.237Z",

```

```

        "Description": "CloudWatch Event Rule SSMOpsItems-SSM-
maintenance-window-execution-failed was triggered. Your SSM Maintenance Window
execution has failed. See below for more details.",
        "LastModifiedBy": "arn:aws:sts::111222333444:assumed-
role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
        "LastModifiedTime": "2019-11-26T20:00:07.237Z",
        "Notifications": "",
        "OperationalData": "{\"/aws/resources\":{\"type
\\":\\\"SearchableString\\\",\\\"value\\\":\\\"[{\\\"arn\\\":\\\"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-0e83ba440dEXAMPLE\\\"]\\\"},\\\"/aws/dedup\\\":
{\\\"type\\\":\\\"SearchableString\\\",\\\"value\\\":\\\"{\\\"dedupString\\\":\\\"SSMOpsItems-SSM-
maintenance-window-execution-failed\\\"]\\\"}}\",
        "OpsItemId": "oi-bb2a0EXAMPLE",
        "RelatedItems": "",
        "Severity": "3",
        "Source": "SSM",
        "Status": "Open",
        "Title": "SSM Maintenance Window execution failed"
    }
  ]
}
}
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan OpsItems](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetOpsSummary](#) di Referensi AWS CLI Perintah.

## get-parameter-history

Contoh kode berikut menunjukkan cara menggunakan `get-parameter-history`.

### AWS CLI

Untuk mendapatkan riwayat nilai untuk parameter

`get-parameter-history` Contoh berikut mencantumkan riwayat perubahan untuk parameter yang ditentukan, termasuk nilainya.

```
aws ssm get-parameter-history \
```

```
--name "MyStringParameter"
```

**Output:**

```
{
  "Parameters": [
    {
      "Name": "MyStringParameter",
      "Type": "String",
      "LastModifiedDate": 1582154711.976,
      "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
      "Description": "This is the first version of my String parameter",
      "Value": "Veni",
      "Version": 1,
      "Labels": [],
      "Tier": "Standard",
      "Policies": []
    },
    {
      "Name": "MyStringParameter",
      "Type": "String",
      "LastModifiedDate": 1582156093.471,
      "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
      "Description": "This is the second version of my String parameter",
      "Value": "Vidi",
      "Version": 2,
      "Labels": [],
      "Tier": "Standard",
      "Policies": []
    },
    {
      "Name": "MyStringParameter",
      "Type": "String",
      "LastModifiedDate": 1582156117.545,
      "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
      "Description": "This is the third version of my String parameter",
      "Value": "Vici",
      "Version": 3,
      "Labels": [],
      "Tier": "Standard",
      "Policies": []
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan versi parameter](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetParameterHistory](#) di Referensi AWS CLI Perintah.

## get-parameter

Contoh kode berikut menunjukkan cara menggunakan `get-parameter`.

### AWS CLI

Contoh 1: Untuk menampilkan nilai parameter

`get-parameter` Contoh berikut mencantumkan nilai untuk parameter tunggal yang ditentukan.

```
aws ssm get-parameter \  
  --name "MyStringParameter"
```

Output:

```
{  
  "Parameter": {  
    "Name": "MyStringParameter",  
    "Type": "String",  
    "Value": "Veni",  
    "Version": 1,  
    "LastModifiedDate": 1530018761.888,  
    "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter"  
    "DataType": "text"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk mendekripsi nilai parameter `SecureString`

`get-parameter` Contoh berikut mendekripsi nilai parameter yang ditentukan `SecureString`.

```
aws ssm get-parameter \  
  --name "MySecureStringParameter" \  
  --with-decryption
```



```
--with-decryption
```

Output:

```
{
  "Parameter": {
    "Name": "MySecureStringParameter",
    "Type": "SecureString",
    "Value": "16679b88-310b-4895-a943-e0764EXAMPLE",
    "Version": 2,
    "LastModifiedDate": 1582155479.205,
    "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MySecureStringParameter"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

Contoh 3: Untuk menampilkan nilai parameter menggunakan label

`get-parameter` Contoh berikut mencantumkan nilai untuk parameter tunggal yang ditentukan dengan label tertentu.

```
aws ssm get-parameter \
  --name "MyParameter:label"
```

Output:

```
{
  "Parameter": {
    "Name": "MyParameter",
    "Type": "String",
    "Value": "parameter version 2",
    "Version": 2,
    "Selector": ":label",
    "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
    "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
    "DataType": "text"
  }
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan label parameter](#) di Panduan Pengguna AWS Systems Manager.

Contoh 4: Untuk menampilkan nilai parameter menggunakan versi

`get-parameter` Contoh berikut mencantumkan nilai untuk versi parameter tunggal yang ditentukan.

```
aws ssm get-parameter \  
  --name "MyParameter:2"
```

Output:

```
{  
  "Parameter": {  
    "Name": "MyParameter",  
    "Type": "String",  
    "Value": "parameter version 2",  
    "Version": 2,  
    "Selector": ":2",  
    "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",  
    "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",  
    "DataType": "text"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan label parameter](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetParameter](#) di Referensi AWS CLI Perintah.

## **get-parameters-by-path**

Contoh kode berikut menunjukkan cara menggunakan `get-parameters-by-path`.

### AWS CLI

Untuk membuat daftar parameter di jalur tertentu

`get-parameters-by-path` Contoh berikut mencantumkan parameter dalam hierarki yang ditentukan.

```
aws ssm get-parameters-by-path \  
  --path "/site/newyork/department/"
```

Output:

```
{  
  "Parameters": [  
    {  
      "Name": "/site/newyork/department/marketing",  
      "Type": "String",  
      "Value": "Floor 2",  
      "Version": 1,  
      "LastModifiedDate": 1530018761.888,  
      "ARN": "arn:aws:ssm:us-east-1:111222333444:parameter/site/newyork/  
department/marketing"  
    },  
    {  
      "Name": "/site/newyork/department/infotech",  
      "Type": "String",  
      "Value": "Floor 3",  
      "Version": 1,  
      "LastModifiedDate": 1530018823.429,  
      "ARN": "arn:aws:ssm:us-east-1:111222333444:parameter/site/newyork/  
department/infotech"  
    },  
    ...  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan hierarki parameter](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetParametersByPath](#) di Referensi AWS CLI Perintah.

## get-parameters

Contoh kode berikut menunjukkan cara menggunakan `get-parameters`.

### AWS CLI

Contoh 1: Untuk membuat daftar nilai untuk parameter

`get-parameters` Contoh berikut mencantumkan nilai-nilai untuk tiga parameter yang ditentukan.

```
aws ssm get-parameters \
  --names "MyStringParameter" "MyStringListParameter" "MyInvalidParameterName"
```

Output:

```
{
  "Parameters": [
    {
      "Name": "MyStringListParameter",
      "Type": "StringList",
      "Value": "alpha,beta,gamma",
      "Version": 1,
      "LastModifiedDate": 1582154764.222,
      "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringListParameter",
      "DataType": "text"
    },
    {
      "Name": "MyStringParameter",
      "Type": "String",
      "Value": "Vici",
      "Version": 3,
      "LastModifiedDate": 1582156117.545,
      "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter",
      "DataType": "text"
    }
  ],
  "InvalidParameters": [
    "MyInvalidParameterName"
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk mencantumkan nama dan nilai dari beberapa parameter menggunakan opsi `--query``

`get-parameters` Contoh berikut mencantumkan nama dan nilai untuk parameter yang ditentukan.

```
aws ssm get-parameters \  
  --names MyStringParameter MyStringListParameter \  
  --query "Parameters[*].{Name:Name,Value:Value}"
```

Output:

```
[  
  {  
    "Name": "MyStringListParameter",  
    "Value": "alpha,beta,gamma"  
  },  
  {  
    "Name": "MyStringParameter",  
    "Value": "Vidi"  
  }  
]
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

Contoh 3: Untuk menampilkan nilai parameter menggunakan label

`get-parameter` Contoh berikut mencantumkan nilai untuk parameter tunggal yang ditentukan dengan label tertentu.

```
aws ssm get-parameter \  
  --name "MyParameter:label"
```

Output:

```
{  
  "Parameters": [  
    {  
      "Name": "MyLabelParameter",  
      "Type": "String",  
      "Value": "parameter by label",  
      "Version": 1,  
      "Selector": ":label",  
      "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",  
      "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",  
      "DataType": "text"  
    }  
  ]  
}
```

```
    },
    {
      "Name": "MyVersionParameter",
      "Type": "String",
      "Value": "parameter by version",
      "Version": 2,
      "Selector": ":2",
      "LastModifiedDate": "2021-03-24T16:20:28.236000-07:00",
      "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/unlabel-param",
      "DataType": "text"
    }
  ],
  "InvalidParameters": []
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan label parameter](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetParameters](#) di Referensi AWS CLI Perintah.

## get-patch-baseline-for-patch-group

Contoh kode berikut menunjukkan cara menggunakan `get-patch-baseline-for-patch-group`.

### AWS CLI

Untuk menampilkan baseline patch untuk grup patch

`get-patch-baseline-for-patch-group` Contoh berikut mengambil rincian tentang baseline patch untuk kelompok patch yang ditentukan.

```
aws ssm get-patch-baseline-for-patch-group \
  --patch-group "DEV"
```

Output:

```
{
  "PatchGroup": "DEV",
  "BaselineId": "pb-0123456789abcdef0",
  "OperatingSystem": "WINDOWS"
}
```

Untuk informasi selengkapnya, lihat Membuat Grup Patch < <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>>\_\_ dan [Tambahkan Grup Patch ke Baseline Patch di](#) Panduan Pengguna Systems Manager AWS .

- Untuk detail API, lihat [GetPatchBaselineForPatchGroup](#) di Referensi AWS CLI Perintah.

## get-patch-baseline

Contoh kode berikut menunjukkan cara menggunakan `get-patch-baseline`.

### AWS CLI

Untuk menampilkan baseline patch

`get-patch-baseline` Contoh berikut mengambil rincian untuk baseline patch yang ditentukan.

```
aws ssm get-patch-baseline \  
  --baseline-id "pb-0123456789abcdef0"
```

Output:

```
{  
  "BaselineId": "pb-0123456789abcdef0",  
  "Name": "WindowsPatching",  
  "OperatingSystem": "WINDOWS",  
  "GlobalFilters": {  
    "PatchFilters": []  
  },  
  "ApprovalRules": {  
    "PatchRules": [  
      {  
        "PatchFilterGroup": {  
          "PatchFilters": [  
            {  
              "Key": "PRODUCT",  
              "Values": [  
                "WindowsServer2016"  
              ]  
            }  
          ]  
        }  
      ]  
    },  
    "ComplianceLevel": "CRITICAL",  
    "ApproveAfterDays": 0,  
  }  
}
```

```

        "EnableNonSecurity": false
      }
    ]
  },
  "ApprovedPatches": [],
  "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
  "ApprovedPatchesEnableNonSecurity": false,
  "RejectedPatches": [],
  "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
  "PatchGroups": [
    "QA",
    "DEV"
  ],
  "CreateDate": 1550244180.465,
  "ModifiedDate": 1550244180.465,
  "Description": "Patches for Windows Servers",
  "Sources": []
}

```

Untuk informasi selengkapnya, lihat [Tentang Patch Baseline](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetPatchBaseline](#) di Referensi AWS CLI Perintah.

## get-service-setting

Contoh kode berikut menunjukkan cara menggunakan `get-service-setting`.

### AWS CLI

Untuk mengambil pengaturan layanan untuk throughput Parameter Store

Berikut `get-service-setting` ini Contoh ini mengambil pengaturan layanan saat ini untuk throughput Parameter Store di wilayah tertentu.

```

aws ssm get-service-setting \
  --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
store/high-throughput-enabled

```

Output:

```
{
```



```
"ServiceSetting": {
  "SettingId": "/ssm/parameter-store/high-throughput-enabled",
  "SettingValue": "false",
  "LastModifiedDate": 1555532818.578,
  "LastModifiedUser": "System",
  "ARN": "arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled",
  "Status": "Default"
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Throughput Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [GetServiceSetting](#) di Referensi AWS CLI Perintah.

## label-parameter-version

Contoh kode berikut menunjukkan cara menggunakan `label-parameter-version`.

### AWS CLI

Contoh 1: Untuk menambahkan label ke versi terbaru dari parameter

`label-parameter-version` Contoh berikut menambahkan label ke versi terbaru dari parameter yang ditentukan.

```
aws ssm label-parameter-version \
  --name "MyStringParameter" \
  --labels "ProductionReady"
```

Output:

```
{
  "InvalidLabels": [],
  "ParameterVersion": 3
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan label parameter](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk menambahkan label ke versi parameter tertentu

`label-parameter-version` Contoh berikut menambahkan label ke versi tertentu dari parameter.

```
aws ssm label-parameter-version \  
  --name "MyStringParameter" \  
  --labels "ProductionReady" \  
  --parameter-version "2" --labels "DevelopmentReady"
```

Untuk informasi selengkapnya, lihat [Bekerja dengan label parameter](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [LabelParameterVersion](#) di Referensi AWS CLI Perintah.

## **list-association-versions**

Contoh kode berikut menunjukkan cara menggunakan `list-association-versions`.

### AWS CLI

Untuk mencantumkan semua versi asosiasi untuk ID asosiasi tertentu

`list-association-versions` Contoh berikut mencantumkan semua versi asosiasi yang ditentukan.

```
aws ssm list-association-versions \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Output:

```
{  
  "AssociationVersions": [  
    {  
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",  
      "AssociationVersion": "1",  
      "CreateDate": 1550505536.726,  
      "Name": "AWS-UpdateSSMAgent",  
      "Parameters": {  
        "allowDowngrade": [  
          "false"  
        ],  
        "version": [  
          ""  
        ]  
      }  
    }  
  ]  
}
```

```

    ]
  },
  "Targets": [
    {
      "Key": "InstanceIds",
      "Values": [
        "i-1234567890abcdef0"
      ]
    }
  ],
  "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
  "AssociationName": "UpdateSSMAgent"
}
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan asosiasi di Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ListAssociationVersions](#) di Referensi AWS CLI Perintah.

## list-associations

Contoh kode berikut menunjukkan cara menggunakan `list-associations`.

### AWS CLI

Contoh 1: Untuk membuat daftar asosiasi Anda untuk contoh tertentu

Contoh daftar-asosiasi berikut mencantumkan semua asosiasi dengan, `AssociationName UpdateSSMAgent`.

```
aws ssm list-associations /
  --association-filter-list "key=AssociationName,value=UpdateSSMAgent"
```

Output:

```
{
  "Associations": [
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-1234567890abcdef0",
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",

```

```

    "AssociationVersion": "1",
    "Targets": [
      {
        "Key": "InstanceIds",
        "Values": [
          "i-016648b75dd622dab"
        ]
      }
    ],
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Associated",
      "AssociationStatusAggregatedCount": {
        "Pending": 1
      }
    },
    "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
    "AssociationName": "UpdateSSMAgent"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan asosiasi di Systems Manager](#) di Panduan Pengguna Systems Manager.

Contoh 2: Untuk membuat daftar asosiasi Anda untuk dokumen tertentu

Contoh daftar-asosiasi berikut mencantumkan semua asosiasi untuk dokumen yang ditentukan.

```

aws ssm list-associations /
  --association-filter-list "key=Name,value=AWS-UpdateSSMAgent"

```

Output:

```

{
  "Associations": [
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-1234567890abcdef0",
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "Targets": [
        {

```

```

        "Key": "InstanceIds",
        "Values": [
            "i-1234567890abcdef0"
        ]
    },
    ],
    "LastExecutionDate": 1550505828.548,
    "Overview": {
        "Status": "Success",
        "DetailedStatus": "Success",
        "AssociationStatusAggregatedCount": {
            "Success": 1
        }
    },
    "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
    "AssociationName": "UpdateSSMAgent"
},
{
    "Name": "AWS-UpdateSSMAgent",
    "InstanceId": "i-9876543210abcdef0",
    "AssociationId": "fbc07ef7-b985-4684-b82b-0123456789ab",
    "AssociationVersion": "1",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-9876543210abcdef0"
            ]
        }
    ],
    "LastExecutionDate": 1550507531.0,
    "Overview": {
        "Status": "Success",
        "AssociationStatusAggregatedCount": {
            "Success": 1
        }
    }
}
]
}
}

```

Untuk informasi selengkapnya, lihat [Bekerja dengan asosiasi di Systems Manager](#) di Panduan Pengguna Systems Manager.

- Untuk detail API, lihat [ListAssociations](#) di Referensi AWS CLI Perintah.

## list-command-invocations

Contoh kode berikut menunjukkan cara menggunakan `list-command-invocations`.

### AWS CLI

Untuk membuat daftar pemanggilan perintah tertentu

`list-command-invocations` Contoh berikut mencantumkan semua pemanggilan perintah.

```
aws ssm list-command-invocations \
  --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
  --details
```

### Output:

```
{
  "CommandInvocations": [
    {
      "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
      "InstanceId": "i-02573cafcfEXAMPLE",
      "InstanceName": "",
      "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
      "DocumentName": "AWS-UpdateSSMAgent",
      "DocumentVersion": "",
      "RequestedDateTime": 1582136283.089,
      "Status": "Success",
      "StatusDetails": "Success",
      "StandardOutputUrl": "",
      "StandardErrorUrl": "",
      "CommandPlugins": [
        {
          "Name": "aws:updateSsmAgent",
          "Status": "Success",
          "StatusDetails": "Success",
          "ResponseCode": 0,
          "ResponseStartDateTime": 1582136283.419,
          "ResponseFinishDateTime": 1582136283.51,
          "Output": "Updating amazon-ssm-agent from 2.3.842.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/"
        }
      ]
    }
  ]
}
```

```

ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been installed,
update skipped\n",
    "StandardOutputUrl": "",
    "StandardErrorUrl": "",
    "OutputS3Region": "us-east-2",
    "OutputS3BucketName": "",
    "OutputS3KeyPrefix": ""
  }
],
"ServiceRole": "",
"NotificationConfig": {
  "NotificationArn": "",
  "NotificationEvents": [],
  "NotificationType": ""
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
},
{
  "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
  "InstanceId": "i-0471e04240EXAMPLE",
  "InstanceName": "",
  "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
  "DocumentName": "AWS-UpdateSSMAgent",
  "DocumentVersion": "",
  "RequestedDateTime": 1582136283.02,
  "Status": "Success",
  "StatusDetails": "Success",
  "StandardOutputUrl": "",
  "StandardErrorUrl": "",
  "CommandPlugins": [
    {
      "Name": "aws:updateSsmAgent",
      "Status": "Success",
      "StatusDetails": "Success",
      "ResponseCode": 0,
      "ResponseStartDateTime": 1582136283.812,
      "ResponseFinishDateTime": 1582136295.031,
      "Output": "Updating amazon-ssm-agent from 2.3.672.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
ssm-agent-manifest.json\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/

```

```

amazon-ssm-us-east-2/amazon-ssm-agent-updater/2.3.842.0/amazon-ssm-agent-updater-
snap-amd64.tar.gz\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/
amazon-ssm-us-east-2/amazon-ssm-agent/2.3.672.0/amazon-ssm-agent-snap-amd64.tar.gz
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
amazon-ssm-agent/2.3.842.0/amazon-ssm-agent-snap-amd64.tar.gz\nInitiating amazon-
ssm-agent update to 2.3.842.0\namazon-ssm-agent updated successfully to 2.3.842.0",
    "StandardOutputUrl": "",
    "StandardErrorUrl": "",
    "OutputS3Region": "us-east-2",
    "OutputS3BucketName": "",
    "OutputS3KeyPrefix": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE/i-0471e04240EXAMPLE/awsupdateSsmAgent"
  }
],
"ServiceRole": "",
"NotificationConfig": {
  "NotificationArn": "",
  "NotificationEvents": [],
  "NotificationType": ""
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
}
]
}

```

Untuk informasi selengkapnya, lihat [Memahami Status Perintah](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ListCommandInvocations](#) di Referensi AWS CLI Perintah.

## list-commands

Contoh kode berikut menunjukkan cara menggunakan `list-commands`.

### AWS CLI

Contoh 1: Untuk mendapatkan status perintah tertentu

`list-commands` Contoh berikut mengambil dan menampilkan status perintah yang ditentukan.



```
aws ssm list-commands \  
  --command-id "0831e1a8-a1ac-4257-a1fd-c831bEXAMPLE"
```

Contoh 2: Untuk mendapatkan status perintah yang diminta setelah tanggal tertentu

`list-commands` Contoh berikut mengambil rincian perintah yang diminta setelah tanggal yang ditentukan.

```
aws ssm list-commands \  
  --filter "key=InvokedAfter,value=2020-02-01T00:00:00Z"
```

Contoh 3: Untuk mencantumkan semua perintah yang diminta dalam AWS akun

`list-commands` Contoh berikut mencantumkan semua perintah yang diminta oleh pengguna di AWS akun saat ini dan Wilayah.

```
aws ssm list-commands
```

Output:

```
{  
  "Commands": [  
    {  
      "CommandId": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE",  
      "DocumentName": "AWS-UpdateSSMAgent",  
      "DocumentVersion": "",  
      "Comment": "b48291dd-ba76-43e0-b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",  
      "ExpiresAfter": "2020-02-19T11:28:02.500000-08:00",  
      "Parameters": {},  
      "InstanceIds": [  
        "i-028ea792daEXAMPLE",  
        "i-02feef8c46EXAMPLE",  
        "i-038613f3f0EXAMPLE",  
        "i-03a530a2d4EXAMPLE",  
        "i-083b678d37EXAMPLE",  
        "i-0dee81debaEXAMPLE"  
      ],  
      "Targets": [],  
      "RequestedDateTime": "2020-02-19T10:18:02.500000-08:00",  
      "Status": "Success",  
      "StatusDetails": "Success",
```

```
"OutputS3BucketName": "",
"OutputS3KeyPrefix": "",
"MaxConcurrency": "50",
"MaxErrors": "100%",
"TargetCount": 6,
"CompletedCount": 6,
"ErrorCount": 0,
"DeliveryTimedOutCount": 0,
"ServiceRole": "",
"NotificationConfig": {
  "NotificationArn": "",
  "NotificationEvents": [],
  "NotificationType": ""
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
}
{
  "CommandId": "e9ade581-c03d-476b-9b07-26667EXAMPLE",
  "DocumentName": "AWS-FindWindowsUpdates",
  "DocumentVersion": "1",
  "Comment": "",
  "ExpiresAfter": "2020-01-24T12:37:31.874000-08:00",
  "Parameters": {
    "KbArticleIds": [
      ""
    ],
    "UpdateLevel": [
      "All"
    ]
  },
  "InstanceIds": [],
  "Targets": [
    {
      "Key": "InstanceIds",
      "Values": [
        "i-00ec29b21eEXAMPLE",
        "i-09911ddd90EXAMPLE"
      ]
    }
  ],
  "RequestedDateTime": "2020-01-24T11:27:31.874000-08:00",
```

```
"Status": "Success",
"StatusDetails": "Success",
"OutputS3BucketName": "my-us-east-2-bucket",
"OutputS3KeyPrefix": "my-rc-output",
"MaxConcurrency": "50",
"MaxErrors": "0",
"TargetCount": 2,
"CompletedCount": 2,
"ErrorCount": 0,
"DeliveryTimedOutCount": 0,
"ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
"NotificationConfig": {
  "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-east-2-
notification-arn",
  "NotificationEvents": [
    "All"
  ],
  "NotificationType": "Invocation"
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
}
{
  "CommandId": "d539b6c3-70e8-4853-80e5-0ce4fEXAMPLE",
  "DocumentName": "AWS-RunPatchBaseline",
  "DocumentVersion": "1",
  "Comment": "",
  "ExpiresAfter": "2020-01-24T12:21:04.350000-08:00",
  "Parameters": {
    "InstallOverrideList": [
      ""
    ],
    "Operation": [
      "Install"
    ],
    "RebootOption": [
      "RebootIfNeeded"
    ],
    "SnapshotId": [
      ""
    ]
  ]
}
```

```
    },
    "InstanceIds": [],
    "Targets": [
      {
        "Key": "InstanceIds",
        "Values": [
          "i-00ec29b21eEXAMPLE",
          "i-09911ddd90EXAMPLE"
        ]
      }
    ],
    "RequestedDateTime": "2020-01-24T11:11:04.350000-08:00",
    "Status": "Success",
    "StatusDetails": "Success",
    "OutputS3BucketName": "my-us-east-2-bucket",
    "OutputS3KeyPrefix": "my-rc-output",
    "MaxConcurrency": "50",
    "MaxErrors": "0",
    "TargetCount": 2,
    "CompletedCount": 2,
    "ErrorCount": 0,
    "DeliveryTimedOutCount": 0,
    "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "NotificationConfig": {
      "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-east-2-
notification-arn",
      "NotificationEvents": [
        "All"
      ],
      "NotificationType": "Invocation"
    },
    "CloudWatchOutputConfig": {
      "CloudWatchLogGroupName": "",
      "CloudWatchOutputEnabled": false
    }
  }
]
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ListCommands](#) di Referensi AWS CLI Perintah.

## list-compliance-items

Contoh kode berikut menunjukkan cara menggunakan `list-compliance-items`.

### AWS CLI

Untuk mencantumkan item kepatuhan untuk contoh tertentu

Contoh ini mencantumkan semua item kepatuhan untuk contoh yang ditentukan.

Perintah:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-types "ManagedInstance"
```

Output:

```
{
  "ComplianceItems": [
    {
      "ComplianceType": "Association",
      "ResourceType": "ManagedInstance",
      "ResourceId": "i-1234567890abcdef0",
      "Id": "8dfe3659-4309-493a-8755-0123456789ab",
      "Title": "",
      "Status": "COMPLIANT",
      "Severity": "UNSPECIFIED",
      "ExecutionSummary": {
        "ExecutionTime": 1550408470.0
      },
      "Details": {
        "DocumentName": "AWS-GatherSoftwareInventory",
        "DocumentVersion": "1"
      }
    },
    {
      "ComplianceType": "Association",
      "ResourceType": "ManagedInstance",
      "ResourceId": "i-1234567890abcdef0",
      "Id": "e4c2ed6d-516f-41aa-aa2a-0123456789ab",
      "Title": "",
      "Status": "COMPLIANT",
      "Severity": "UNSPECIFIED",
      "ExecutionSummary": {
```

```

        "ExecutionTime": 1550508475.0
      },
      "Details": {
        "DocumentName": "AWS-UpdateSSMAgent",
        "DocumentVersion": "1"
      }
    },
    ...
  ],
  "NextToken": "--token string truncated--"
}

```

Untuk mencantumkan item kepatuhan untuk instance dan ID asosiasi tertentu

Contoh ini mencantumkan semua item kepatuhan untuk instance dan ID asosiasi yang ditentukan.

Perintah:

```

aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-types
"ManagedInstance" --filters "Key=ComplianceType,Values=Association,Type=EQUAL "
"Key=Id,Values=e4c2ed6d-516f-41aa-aa2a-0123456789ab,Type=EQUAL "

```

Untuk mencantumkan item kepatuhan untuk sebuah instans setelah tanggal dan waktu tertentu

Contoh ini mencantumkan semua item kepatuhan untuk sebuah instance setelah tanggal dan waktu yang ditentukan.

Perintah:

```

aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-types
"ManagedInstance" --filters
"Key=ExecutionTime,Values=2019-02-18T16:00:00Z,Type=GREATER_THAN"

```

- Untuk detail API, lihat [ListComplianceItems](#) di Referensi AWS CLI Perintah.

## list-compliance-summaries

Contoh kode berikut menunjukkan cara menggunakan `list-compliance-summaries`.

### AWS CLI

Untuk membuat daftar ringkasan kepatuhan untuk semua jenis kepatuhan

Contoh ini mencantumkan ringkasan kepatuhan untuk semua jenis kepatuhan di akun Anda.

Perintah:

```
aws ssm list-compliance-summaries
```

Output:

```
{
  "ComplianceSummaryItems": [
    {
      "ComplianceType": "Association",
      "CompliantSummary": {
        "CompliantCount": 2,
        "SeveritySummary": {
          "CriticalCount": 0,
          "HighCount": 0,
          "MediumCount": 0,
          "LowCount": 0,
          "InformationalCount": 0,
          "UnspecifiedCount": 2
        }
      },
      "NonCompliantSummary": {
        "NonCompliantCount": 0,
        "SeveritySummary": {
          "CriticalCount": 0,
          "HighCount": 0,
          "MediumCount": 0,
          "LowCount": 0,
          "InformationalCount": 0,
          "UnspecifiedCount": 0
        }
      }
    },
    {
      "ComplianceType": "Patch",
      "CompliantSummary": {
        "CompliantCount": 1,
        "SeveritySummary": {
          "CriticalCount": 0,
          "HighCount": 0,
          "MediumCount": 0,
```

```

        "LowCount": 0,
        "InformationalCount": 0,
        "UnspecifiedCount": 1
      }
    },
    "NonCompliantSummary": {
      "NonCompliantCount": 1,
      "SeveritySummary": {
        "CriticalCount": 1,
        "HighCount": 0,
        "MediumCount": 0,
        "LowCount": 0,
        "InformationalCount": 0,
        "UnspecifiedCount": 0
      }
    }
  },
  ...
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiB1bnRlcXV9b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

Untuk membuat daftar ringkasan kepatuhan untuk jenis kepatuhan tertentu

Contoh ini mencantumkan ringkasan kepatuhan untuk jenis kepatuhan Patch.

Perintah:

```
aws ssm list-compliance-summaries --filters
  "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Untuk detail API, lihat [ListComplianceSummaries](#) di Referensi AWS CLI Perintah.

## list-document-metadata-history

Contoh kode berikut menunjukkan cara menggunakan `list-document-metadata-history`.

AWS CLI

Contoh: Untuk melihat riwayat persetujuan dan status untuk templat perubahan

`list-document-metadata-history` Contoh berikut mengembalikan riwayat persetujuan untuk perubahan template Change Manager yang ditentukan.



```
aws ssm list-document-metadata-history \  
  --name MyChangeManageTemplate \  
  --metadata DocumentReviews
```

Output:

```
{  
  "Name": "MyChangeManagerTemplate",  
  "DocumentVersion": "1",  
  "Author": "arn:aws:iam::111222333444;user/JohnDoe",  
  "Metadata": {  
    "ReviewerResponse": [  
      {  
        "CreateTime": "2021-07-30T11:58:28.025000-07:00",  
        "UpdateTime": "2021-07-30T12:01:19.274000-07:00",  
        "ReviewStatus": "APPROVED",  
        "Comment": [  
          {  
            "Type": "COMMENT",  
            "Content": "I approve this template version"  
          }  
        ],  
        "Reviewer": "arn:aws:iam::111222333444;user/ShirleyRodriguez"  
      },  
      {  
        "CreateTime": "2021-07-30T11:58:28.025000-07:00",  
        "UpdateTime": "2021-07-30T11:58:28.025000-07:00",  
        "ReviewStatus": "PENDING"  
      }  
    ]  
  }  
}
```

Untuk informasi selengkapnya, lihat [Meninjau dan menyetujui atau menolak templat perubahan di Panduan](#) Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ListDocumentMetadataHistory](#) di Referensi AWS CLI Perintah.

## list-document-versions

Contoh kode berikut menunjukkan cara menggunakan `list-document-versions`.

## AWS CLI

Untuk daftar versi dokumen

`list-document-versions` Contoh berikut mencantumkan semua versi untuk dokumen Systems Manager.

```
aws ssm list-document-versions \  
  --name "Example"
```

Output:

```
{  
  "DocumentVersions": [  
    {  
      "Name": "Example",  
      "DocumentVersion": "1",  
      "CreateDate": 1583257938.266,  
      "IsDefaultVersion": true,  
      "DocumentFormat": "YAML",  
      "Status": "Active"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengirim Perintah yang Menggunakan Parameter Versi Dokumen](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ListDocumentVersions](#) di Referensi AWS CLI Perintah.

## **list-documents**

Contoh kode berikut menunjukkan cara menggunakan `list-documents`.

## AWS CLI

Contoh 1: Untuk daftar dokumen

`list-documents` Contoh berikut mencantumkan dokumen yang dimiliki oleh akun permintaan yang ditandai dengan tag kustom.

```
aws ssm list-documents \  
  --tag-key "tag-key" \  
  --tag-value "tag-value"
```

```
--filters Key=Owner,Values=Self Key=tag:DocUse,Values=Testing
```

Output:

```
{
  "DocumentIdentifiers": [
    {
      "Name": "Example",
      "Owner": "29884EXAMPLE",
      "PlatformTypes": [
        "Windows",
        "Linux"
      ],
      "DocumentVersion": "1",
      "DocumentType": "Automation",
      "SchemaVersion": "0.3",
      "DocumentFormat": "YAML",
      "Tags": [
        {
          "Key": "DocUse",
          "Value": "Testing"
        }
      ]
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Dokumen AWS Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk daftar dokumen bersama

`list-documents` Contoh berikut mencantumkan dokumen bersama, termasuk dokumen bersama pribadi yang tidak dimiliki oleh AWS.

```
aws ssm list-documents \
  --filters Key=Name,Values=sharedDocNamePrefix Key=Owner,Values=Private
```

Output:

```
{
  "DocumentIdentifiers": [
```

```
{
  "Name": "Example",
  "Owner": "12345EXAMPLE",
  "PlatformTypes": [
    "Windows",
    "Linux"
  ],
  "DocumentVersion": "1",
  "DocumentType": "Command",
  "SchemaVersion": "0.3",
  "DocumentFormat": "YAML",
  "Tags": []
}
]
```

Untuk informasi selengkapnya, lihat [Dokumen AWS Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ListDocuments](#) di Referensi AWS CLI Perintah.

## list-inventory-entries

Contoh kode berikut menunjukkan cara menggunakan `list-inventory-entries`.

### AWS CLI

Contoh 1: Untuk melihat entri jenis inventaris tertentu untuk sebuah instance

`list-inventory-entries` Contoh berikut ini mencantumkan entri inventaris untuk jenis AWS inventaris: Aplikasi pada instance tertentu.

```
aws ssm list-inventory-entries \
  --instance-id "i-1234567890abcdef0" \
  --type-name "AWS:Application"
```

Output:

```
{
  "TypeName": "AWS:Application",
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.1",
  "CaptureTime": "2019-02-15T12:17:55Z",
```

```

"Entries": [
  {
    "Architecture": "i386",
    "Name": "Amazon SSM Agent",
    "PackageId": "{88a60be2-89a1-4df8-812a-80863c2a2b68}",
    "Publisher": "Amazon Web Services",
    "Version": "2.3.274.0"
  },
  {
    "Architecture": "x86_64",
    "InstalledTime": "2018-05-03T13:42:34Z",
    "Name": "AmazonCloudWatchAgent",
    "Publisher": "",
    "Version": "1.200442.0"
  }
]
}

```

Contoh 2: Untuk melihat entri inventaris kustom yang ditetapkan ke sebuah instance

`list-inventory-entries` Contoh berikut mencantumkan entri inventaris kustom yang ditetapkan ke sebuah instance.

```

aws ssm list-inventory-entries \
  --instance-id "i-1234567890abcdef0" \
  --type-name "Custom:RackInfo"

```

Output:

```

{
  "TypeName": "Custom:RackInfo",
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.0",
  "CaptureTime": "2021-05-22T10:01:01Z",
  "Entries": [
    {
      "RackLocation": "Bay B/Row C/Rack D/Shelf E"
    }
  ]
}

```

- Untuk detail API, lihat [ListInventoryEntries](#) di Referensi AWS CLI Perintah.

## list-ops-item-related-items

Contoh kode berikut menunjukkan cara menggunakan `list-ops-item-related-items`.

### AWS CLI

Untuk membuat daftar sumber daya item terkait dari sebuah OpsItem

`list-ops-item-related-items` Contoh berikut mencantumkan sumber daya item terkait dari sebuah OpsItem

```
aws ssm list-ops-item-related-items \
  --ops-item-id "oi-f99f2EXAMPLE"
```

Output:

```
{
  "Summaries": [
    {
      "OpsItemId": "oi-f99f2EXAMPLE",
      "AssociationId": "e2036148-cccb-490e-ac2a-390e5EXAMPLE",
      "ResourceType": "AWS::SSMIncidents::IncidentRecord",
      "AssociationType": "IsParentOf",
      "ResourceUri": "arn:aws:ssm-incidents::111122223333:incident-record/
example-response/64bd9b45-1d0e-2622-840d-03a87a1451fa",
      "CreatedBy": {
        "Arn": "arn:aws:sts::111122223333:assumed-role/
AWSServiceRoleForIncidentManager/IncidentResponse"
      },
      "CreatedTime": "2021-08-11T18:47:14.994000+00:00",
      "LastModifiedBy": {
        "Arn": "arn:aws:sts::111122223333:assumed-role/
AWSServiceRoleForIncidentManager/IncidentResponse"
      },
      "LastModifiedTime": "2021-08-11T18:47:14.994000+00:00"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan insiden Manajer Insiden OpsCenter di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [ListOpsItemRelatedItems](#) di Referensi AWS CLI Perintah.

## list-resource-compliance-summaries

Contoh kode berikut menunjukkan cara menggunakan `list-resource-compliance-summaries`.

### AWS CLI

Untuk membuat daftar jumlah ringkasan kepatuhan tingkat sumber daya

Contoh ini mencantumkan jumlah ringkasan kepatuhan tingkat sumber daya.

Perintah:

```
aws ssm list-resource-compliance-summaries
```

Output:

```
{
  "ResourceComplianceSummaryItems": [
    {
      "ComplianceType": "Association",
      "ResourceType": "ManagedInstance",
      "ResourceId": "i-1234567890abcdef0",
      "Status": "COMPLIANT",
      "OverallSeverity": "UNSPECIFIED",
      "ExecutionSummary": {
        "ExecutionTime": 1550509273.0
      },
      "CompliantSummary": {
        "CompliantCount": 2,
        "SeveritySummary": {
          "CriticalCount": 0,
          "HighCount": 0,
          "MediumCount": 0,
          "LowCount": 0,
          "InformationalCount": 0,
          "UnspecifiedCount": 2
        }
      },
      "NonCompliantSummary": {
        "NonCompliantCount": 0,
        "SeveritySummary": {
          "CriticalCount": 0,
          "HighCount": 0,
          "MediumCount": 0,

```

```
        "LowCount": 0,
        "InformationalCount": 0,
        "UnspecifiedCount": 0
    }
}
},
{
    "ComplianceType": "Patch",
    "ResourceType": "ManagedInstance",
    "ResourceId": "i-9876543210abcdef0",
    "Status": "COMPLIANT",
    "OverallSeverity": "UNSPECIFIED",
    "ExecutionSummary": {
        "ExecutionTime": 1550248550.0,
        "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
        "ExecutionType": "Command"
    },
    "CompliantSummary": {
        "CompliantCount": 397,
        "SeveritySummary": {
            "CriticalCount": 0,
            "HighCount": 0,
            "MediumCount": 0,
            "LowCount": 0,
            "InformationalCount": 0,
            "UnspecifiedCount": 397
        }
    },
    "NonCompliantSummary": {
        "NonCompliantCount": 0,
        "SeveritySummary": {
            "CriticalCount": 0,
            "HighCount": 0,
            "MediumCount": 0,
            "LowCount": 0,
            "InformationalCount": 0,
            "UnspecifiedCount": 0
        }
    }
}
},
],
"NextToken": "--token string truncated--"
}
```



Untuk membuat daftar ringkasan kepatuhan tingkat sumber daya untuk jenis kepatuhan tertentu

Contoh ini mencantumkan ringkasan kepatuhan tingkat sumber daya untuk jenis kepatuhan Patch.

Perintah:

```
aws ssm list-resource-compliance-summaries --filters
"Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Untuk detail API, lihat [ListResourceComplianceSummaries](#) di Referensi AWS CLI Perintah.

## list-resource-data-sync

Contoh kode berikut menunjukkan cara menggunakan `list-resource-data-sync`.

AWS CLI

Untuk membuat daftar konfigurasi sinkronisasi data sumber daya

Contoh ini mengambil informasi tentang konfigurasi sinkronisasi data sumber daya Anda.

```
aws ssm list-resource-data-sync
```

Output:

```
{
  "ResourceDataSyncItems": [
    {
      "SyncName": "MyResourceDataSync",
      "S3Destination": {
        "BucketName": "ssm-resource-data-sync",
        "SyncFormat": "JsonSerDe",
        "Region": "us-east-1"
      },
      "LastSyncTime": 1550261472.003,
      "LastSuccessfulSyncTime": 1550261472.003,
      "LastStatus": "Successful",
      "SyncCreatedTime": 1543235736.72,
      "LastSyncStatusMessage": "The sync was successfully completed"
    }
  ]
}
```

```
]
}
```

- Untuk detail API, lihat [ListResourceDataSync](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk mencantumkan tag yang diterapkan ke baseline patch

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk baseline patch.

```
aws ssm list-tags-for-resource \
  --resource-type "PatchBaseline" \
  --resource-id "pb-0123456789abcdef0"
```

Output:

```
{
  "TagList": [
    {
      "Key": "Environment",
      "Value": "Production"
    },
    {
      "Key": "Region",
      "Value": "EMEA"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Menandai AWS Sumber Daya](#) di Referensi AWS Umum.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## modify-document-permission

Contoh kode berikut menunjukkan cara menggunakan `modify-document-permission`.

## AWS CLI

Untuk memodifikasi izin dokumen

`modify-document-permission` Contoh berikut membagikan dokumen Systems Manager secara publik.

```
aws ssm modify-document-permission \  
  --name "Example" \  
  --permission-type "Share" \  
  --account-ids-to-add "All"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Membagikan Dokumen Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ModifyDocumentPermission](#) di Referensi AWS CLI Perintah.

## `put-compliance-items`

Contoh kode berikut menunjukkan cara menggunakan `put-compliance-items`.

## AWS CLI

Untuk mendaftarkan jenis kepatuhan dan rincian kepatuhan ke instans yang ditunjuk

Contoh ini mendaftarkan jenis kepatuhan `Custom:AVCheck` ke instance terkelola yang ditentukan. Tidak ada output jika perintah berhasil.

Perintah:

```
aws ssm put-compliance-items --resource-id "i-1234567890abcdef0" --  
resource-type "ManagedInstance" --compliance-type "Custom:AVCheck"  
  --execution-summary "ExecutionTime=2019-02-18T16:00:00Z" --items  
  "Id=Version2.0,Title=ScanHost,Severity=CRITICAL,Status=COMPLIANT"
```

- Untuk detail API, lihat [PutComplianceItems](#) di Referensi AWS CLI Perintah.

## `put-inventory`

Contoh kode berikut menunjukkan cara menggunakan `put-inventory`.

## AWS CLI

Untuk menetapkan metadata pelanggan ke sebuah instance

Contoh ini menetapkan informasi lokasi rak ke instans. Tidak ada output jika perintah berhasil.

Perintah (Linux):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --items
' [{"TypeName": "Custom:RackInfo", "SchemaVersion": "1.0", "CaptureTime":
"2019-01-22T10:01:01Z", "Content": [{"RackLocation": "Bay B/Row C/Rack D/Shelf
E"}]} ]'
```

Perintah (Windows):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --items
"TypeName=Custom:RackInfo,SchemaVersion=1.0,CaptureTime=2019-01-22T10:01:01Z,Content=[{Rack
B/Row C/Rack D/Shelf F'}]"
```

- Untuk detail API, lihat [PutInventory](#) di Referensi AWS CLI Perintah.

## put-parameter

Contoh kode berikut menunjukkan cara menggunakan `put-parameter`.

### AWS CLI

Contoh 1: Untuk mengubah nilai parameter

`put-parameter` Contoh berikut mengubah nilai parameter yang ditentukan.

```
aws ssm put-parameter \
  --name "MyStringParameter" \
  --type "String" \
  --value "Vici" \
  --overwrite
```

Output:

```
{
  "Version": 2,
```

```
"Tier": "Standard"
}
```

Untuk informasi selengkapnya, lihat [Membuat parameter Systems Manager \(AWS CLI\)](https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html), 'Mengelola tingkatan parameter < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>', dan Bekerja dengan [kebijakan parameter di](#) Panduan Pengguna Systems Manager.AWS

Contoh 2: Untuk membuat parameter lanjutan

`put-parameter` Contoh berikut menciptakan parameter lanjutan.

```
aws ssm put-parameter \
  --name "MyAdvancedParameter" \
  --description "This is an advanced parameter" \
  --value "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat [truncated]" \
  --type "String" \
  --tier Advanced
```

Output:

```
{
  "Version": 1,
  "Tier": "Advanced"
}
```

Untuk informasi selengkapnya, lihat [Membuat parameter Systems Manager \(AWS CLI\)](https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html), 'Mengelola tingkatan parameter < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>', dan Bekerja dengan [kebijakan parameter di](#) Panduan Pengguna Systems Manager.AWS

Contoh 3: Untuk mengonversi parameter standar ke parameter lanjutan

`put-parameter` Contoh berikut mengubah parameter standar yang ada menjadi parameter lanjutan.

```
aws ssm put-parameter \
```

```
--name "MyConvertedParameter" \
--value "abc123" \
--type "String" \
--tier Advanced \
--overwrite
```

**Output:**

```
{
  "Version": 2,
  "Tier": "Advanced"
}
```

Untuk informasi selengkapnya, lihat [Membuat parameter Systems Manager \(AWS CLI\)](#), 'Mengelola tingkatan parameter < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>` \_\_, dan Bekerja dengan [kebijakan parameter di](#) Panduan Pengguna Systems Manager.AWS

Contoh 4: Untuk membuat parameter dengan kebijakan terlampir

`put-parameter`Contoh berikut membuat parameter lanjutan dengan kebijakan parameter terlampir.

```
aws ssm put-parameter \
  --name "/Finance/Payroll/q2accesskey" \
  --value "P@sSw)rd" \
  --type "SecureString" \
  --tier Advanced \
  --policies "[{"Type":"Expiration","Version":"1.0","Attributes":{"Timestamp":"2020-06-30T00:00:00.000Z"}}, {"Type":"ExpirationNotification","Version":"1.0","Attributes":{"Before":"5","Unit":"Days"}}, {"Type":"NoChangeNotification","Version":"1.0","Attributes":{"After":"60","Unit":"Days"}}]"
```

**Output:**

```
{
  "Version": 1,
  "Tier": "Advanced"
}
```

Untuk informasi selengkapnya, lihat [Membuat parameter Systems Manager \(AWS CLI\)](https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html), 'Mengelola tingkatan parameter < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>', dan Bekerja dengan [kebijakan parameter di](#) Panduan Pengguna Systems Manager.AWS

Contoh 5: Untuk menambahkan kebijakan ke parameter yang ada

put-parameterContoh berikut melampirkan kebijakan ke parameter lanjutan yang ada.

```
aws ssm put-parameter \
  --name "/Finance/Payroll/q2accesskey" \
  --value "N3wP@sSw)rd" \
  --type "SecureString" \
  --tier Advanced \
  --policies "[{"Type":"Expiration","Version":"1.0","Attributes":{"Timestamp":"2020-06-30T00:00:00.000Z"}}, {"Type":"ExpirationNotification","Version":"1.0","Attributes":{"Before":"5","Unit":"Days"}}, {"Type":"NoChangeNotification","Version":"1.0","Attributes":{"After":"60","Unit":"Days"}}]"
  --overwrite
```

Output:

```
{
  "Version": 2,
  "Tier": "Advanced"
}
```

Untuk informasi selengkapnya, lihat [Membuat parameter Systems Manager \(AWS CLI\)](https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html), 'Mengelola tingkatan parameter < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>', dan Bekerja dengan [kebijakan parameter di](#) Panduan Pengguna Systems Manager.AWS

- Untuk detail API, lihat [PutParameter](#) di Referensi AWS CLI Perintah.

## register-default-patch-baseline

Contoh kode berikut menunjukkan cara menggunakanregister-default-patch-baseline.

AWS CLI

Untuk mengatur baseline patch default

`register-default-patch-baseline` Contoh berikut mendaftarkan baseline patch kustom yang ditentukan sebagai baseline patch default untuk jenis sistem operasi yang didukungnya.

```
aws ssm register-default-patch-baseline \  
  --baseline-id "pb-abc123cf9bEXAMPLE"
```

Output:

```
{  
  "BaselineId": "pb-abc123cf9bEXAMPLE"  
}
```

`register-default-patch-baseline` Contoh berikut mendaftarkan baseline patch default yang disediakan oleh for AWS CentOS sebagai baseline patch default.

```
aws ssm register-default-patch-baseline \  
  --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/  
pb-0574b43a65ea646ed"
```

Output:

```
{  
  "BaselineId": "pb-abc123cf9bEXAMPLE"  
}
```

Untuk informasi [selengkapnya, lihat Tentang Garis Dasar Patch Standar dan Kustom di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [RegisterDefaultPatchBaseline](#) di Referensi AWS CLI Perintah.

## **register-patch-baseline-for-patch-group**

Contoh kode berikut menunjukkan cara menggunakan `register-patch-baseline-for-patch-group`.

### AWS CLI

Untuk mendaftarkan baseline patch untuk grup patch

`register-patch-baseline-for-patch-group` Contoh berikut mendaftarkan baseline patch untuk grup patch.



```
aws ssm register-patch-baseline-for-patch-group \  
  --baseline-id "pb-045f10b4f382baeda" \  
  --patch-group "Production"
```

Output:

```
{  
  "BaselineId": "pb-045f10b4f382baeda",  
  "PatchGroup": "Production"  
}
```

Untuk informasi selengkapnya, lihat Membuat Grup Patch < <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>> dan [Tambahkan Grup Patch ke Baseline Patch di](#) Panduan Pengguna Systems Manager AWS .

- Untuk detail API, lihat [RegisterPatchBaselineForPatchGroup](#) di Referensi AWS CLI Perintah.

## register-target-with-maintenance-window

Contoh kode berikut menunjukkan cara menggunakan `register-target-with-maintenance-window`.

AWS CLI

Contoh 1: Untuk mendaftarkan satu target dengan jendela pemeliharaan

`register-target-with-maintenance-window` Contoh berikut mendaftarkan sebuah instance dengan jendela pemeliharaan.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --target "Key=InstanceIds,Values=i-0000293ffd8c57862" \  
  --owner-information "Single instance" \  
  --resource-type "INSTANCE"
```

Output:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
```

```
}
```

Contoh 2: Untuk mendaftarkan beberapa target dengan jendela pemeliharaan menggunakan ID instance

`register-target-with-maintenance-window` Contoh berikut mendaftarkan dua instance dengan jendela pemeliharaan dengan menentukan ID instans mereka.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --target "Key=InstanceIds,Values=i-0000293ffd8c57862,i-0cb2b964d3e14fd9f" \  
  --owner-information "Two instances in a list" \  
  --resource-type "INSTANCE"
```

Output:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Contoh 3: Untuk mendaftarkan target dengan jendela pemeliharaan menggunakan tag sumber daya

`register-target-with-maintenance-window` Contoh berikut mendaftarkan instance dengan jendela pemeliharaan dengan menentukan tag sumber daya yang telah diterapkan ke instance.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-06cf17cbefcb4bf4f" \  
  --targets "Key=tag:Environment,Values=Prod" "Key=Role,Values=Web" \  
  --owner-information "Production Web Servers" \  
  --resource-type "INSTANCE"
```

Output:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Contoh 4: Untuk mendaftarkan target menggunakan sekelompok kunci tag

`register-target-with-maintenance-window` Contoh berikut mendaftarkan instance yang semuanya memiliki satu atau lebih kunci tag yang ditetapkan untuk mereka, terlepas dari nilai kunci mereka.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-0c50858d01EXAMPLE" \  
  --resource-type "INSTANCE" \  
  --target "Key=tag-key,Values=Name,Instance-Type,CostCenter"
```

Output:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Contoh 5: Untuk mendaftarkan target menggunakan nama grup sumber daya

`register-target-with-maintenance-window` Contoh berikut mendaftarkan grup sumber daya tertentu, terlepas dari jenis sumber daya yang dikandungnya.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-0c50858d01EXAMPLE" \  
  --resource-type "RESOURCE_GROUP" \  
  --target "Key=resource-groups:Name,Values=MyResourceGroup"
```

Output:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Instans Target dengan Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [RegisterTargetWithMaintenanceWindow](#) di Referensi AWS CLI Perintah.

## **register-task-with-maintenance-window**

Contoh kode berikut menunjukkan cara menggunakan `register-task-with-maintenance-window`.

## AWS CLI

Contoh 1: Untuk mendaftarkan tugas Otomasi dengan jendela pemeliharaan

`register-task-with-maintenance-window` Contoh berikut mendaftarkan tugas Otomasi dengan jendela pemeliharaan yang ditargetkan pada sebuah instance.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-082dcd7649EXAMPLE" \
  --targets Key=InstanceIds,Values=i-1234520122EXAMPLE \
  --task-arn AWS-RestartEC2Instance \
  --service-role-arn arn:aws:iam::111222333444:role/SSM --task-type AUTOMATION \
  --task-invocation-parameters "{\"Automation\":{\"DocumentVersion\":\"\$LATEST\"},\
  \"Parameters\":{\"InstanceId\":[\"{{RESOURCE_ID}}\"]}}\" \
  --priority 0 \
  --max-concurrency 1 \
  --max-errors 1 \
  --name "AutomationExample" \
  --description "Restarting EC2 Instance for maintenance"
```

Output:

```
{
  "WindowTaskId": "11144444-5555-6666-7777-88888888"
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Tugas dengan Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk mendaftarkan tugas Lambda dengan Jendela Pemeliharaan

`register-task-with-maintenance-window` Contoh berikut mendaftarkan tugas Lambda dengan Jendela Pemeliharaan yang ditargetkan pada sebuah instance.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-082dcd7649dee04e4" \
  --targets Key=InstanceIds,Values=i-12344d305eEXAMPLE \
  --task-arn arn:aws:lambda:us-east-1:111222333444:function:SSMTestLAMBDA \
  --service-role-arn arn:aws:iam::111222333444:role/SSM \
  --task-type LAMBDA \
  --task-invocation-parameters '{"Lambda":{"Payload":{"InstanceId":\
  \"{{RESOURCE_ID}}\"},\"targetType\":\"{{TARGET_TYPE}}\"},\"Qualifier\":\"$LATEST\"}' \
```

```
--priority 0 \
--max-concurrency 10 \
--max-errors 5 \
--name "Lambda_Example" \
--description "My Lambda Example"
```

**Output:**

```
{
  "WindowTaskId": "22244444-5555-6666-7777-88888888"
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Tugas dengan Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

Contoh 3: Untuk mendaftar tugas Run Command dengan jendela pemeliharaan

`register-task-with-maintenance-window` Contoh berikut mendaftar tugas Run Command dengan jendela pemeliharaan yang ditargetkan pada sebuah instance.

```
aws ssm register-task-with-maintenance-window \
--window-id "mw-082dcd7649dee04e4" \
--targets "Key=InstanceIds,Values=i-12344d305eEXAMPLE" \
--service-role-arn "arn:aws:iam::111222333444:role/SSM" \
--task-type "RUN_COMMAND" \
--name "SSMInstallPowerShellModule" \
--task-arn "AWS-InstallPowerShellModule" \
--task-invocation-parameters "{\"RunCommand\":{\"Comment\":"\"\",
\\\"OutputS3BucketName\\\":\"runcommandlogs\\\",\\\"Parameters\\\":{\"commands\":[\\\"Get-
Module -ListAvailable\\\"],\\\"executionTimeout\":[\\\"3600\\\"],\\\"source\":[\\\"https://
\\gallery.technet.microsoft.com/EZ0ut-33ae0fb7/file/110351/1/EZ0ut.zip\\\"],
\\\"workingDirectory\":[\\\"\\\\\\\\\\\"]},\\\"TimeoutSeconds\\\":600}}" \
--max-concurrency 1 \
--max-errors 1 \
--priority 10
```

**Output:**

```
{
  "WindowTaskId": "33344444-5555-6666-7777-88888888"
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Tugas dengan Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

Contoh 4: Untuk mendaftar tugas Step Functions dengan jendela pemeliharaan

`register-task-with-maintenance-window` Contoh berikut mendaftar tugas Step Functions dengan jendela pemeliharaan yang ditargetkan pada sebuah instance.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-1234d787d6EXAMPLE" \
  --targets Key=WindowTargetIds,Values=12347414-69c3-49f8-95b8-ed2dcEXAMPLE \
  --task-arn arn:aws:states:us-
east-1:111222333444:stateMachine:SSMTestStateMachine \
  --service-role-arn arn:aws:iam::111222333444:role/MaintenanceWindows \
  --task-type STEP_FUNCTIONS \
  --task-invocation-parameters '{"StepFunctions":{"Input":{"\"InstanceId\":
\"{{RESOURCE_ID}}\"}}}' \
  --priority 0 \
  --max-concurrency 10 \
  --max-errors 5 \
  --name "Step_Functions_Example" \
  --description "My Step Functions Example"
```

Output:

```
{
  "WindowTaskId": "44444444-5555-6666-7777-88888888"
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Tugas dengan Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

Contoh 5: Untuk mendaftar tugas menggunakan ID target windows pemeliharaan

`register-task-with-maintenance-window` Contoh berikut mendaftar tugas menggunakan ID target jendela pemeliharaan. ID target jendela pemeliharaan ada di output dari `aws ssm register-target-with-maintenance-window` perintah. Anda juga dapat mengambilnya dari output `aws ssm describe-maintenance-window-targets` perintah.

```
aws ssm register-task-with-maintenance-window \
```

```
--targets "Key=WindowTargetIds,Values=350d44e6-28cc-44e2-951f-4b2c9EXAMPLE" \  
--task-arn "AWS-RunShellScript" \  
--service-role-arn "arn:aws:iam::111222333444:role/MaintenanceWindowsRole" \  
--window-id "mw-ab12cd34eEXAMPLE" \  
--task-type "RUN_COMMAND" \  
--task-parameters "{\"commands\":{\"Values\":[\"df\"]}}" \  
--max-concurrency 1 \  
--max-errors 1 \  
--priority 10
```

Output:

```
{  
  "WindowTaskId": "33344444-5555-6666-7777-88888888"  
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan Tugas dengan Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [RegisterTaskWithMaintenanceWindow](#) di Referensi AWS CLI Perintah.

## **remove-tags-from-resource**

Contoh kode berikut menunjukkan cara menggunakan `remove-tags-from-resource`.

### AWS CLI

Untuk menghapus tag dari baseline patch

`remove-tags-from-resource` Contoh berikut menghapus tag dari baseline patch.

```
aws ssm remove-tags-from-resource \  
  --resource-type "PatchBaseline" \  
  --resource-id "pb-0123456789abcdef0" \  
  --tag-keys "Region"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai AWS Sumber Daya](#) di Referensi AWS Umum.

- Untuk detail API, lihat [RemoveTagsFromResource](#) di Referensi AWS CLI Perintah.

## reset-service-setting

Contoh kode berikut menunjukkan cara menggunakan `reset-service-setting`.

### AWS CLI

Untuk mengatur ulang setelan layanan untuk throughput Parameter Store

`reset-service-setting` Contoh berikut mengatur ulang setelan layanan untuk throughput Parameter Store di wilayah tertentu agar tidak lagi menggunakan peningkatan throughput.

```
aws ssm reset-service-setting \
  --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
  store/high-throughput-enabled
```

Output:

```
{
  "ServiceSetting": {
    "SettingId": "/ssm/parameter-store/high-throughput-enabled",
    "SettingValue": "false",
    "LastModifiedDate": 1555532818.578,
    "LastModifiedUser": "System",
    "ARN": "arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
    store/high-throughput-enabled",
    "Status": "Default"
  }
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Throughput Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ResetServiceSetting](#) di Referensi AWS CLI Perintah.

## resume-session

Contoh kode berikut menunjukkan cara menggunakan `resume-session`.

### AWS CLI

Untuk melanjutkan sesi Manajer Sesi



`resume-session` Contoh ini melanjutkan sesi Session Manager dengan instance setelah terputus. Perhatikan bahwa perintah interaktif ini memerlukan plugin Session Manager untuk diinstal pada mesin klien yang melakukan panggilan.

```
aws ssm resume-session \
  --session-id Mary-Major-07a16060613c408b5
```

Output:

```
{
  "SessionId": "Mary-Major-07a16060613c408b5",
  "TokenValue":
    "AAEAAVbTGsa0nyvcUoNGqifbv5r/8lgxuQljCuY8qVcv0noBAAAAAFxtd3jIXAFUUXGTJ7zF/
    AWJPwDvi0lF5p3d1AgrqVIV06IEXhkHLz0/1gXKRKEME71E6TL0p1LDJAMZ
    +kREejkZu4c5AxMkrQjMF+gtHP1bYJKTwtHQd1wjulPLex08SH17g5R/
    wekrj6WsDUpnEegFBfGftpAIz2GXQVfTJXKfkc5qepQ11C11D0IT2doz0qXgHwfQHfAKLErM5dWDZqKwyT1Z3iw7unQd
    +ihfGa6MEJJ97Jmat/a2TspEn0jNn9Mvu5iwXIW2yCvWZrGUj+/
    QI5Xr7s1XJBEEnSKR54o4fN0GV9RWl0RZsZm1m1ki0JJtiwwgZ",
  "StreamUrl": "wss://ssmmessages.us-east-2.amazonaws.com/v1/data-channel/Mary-
    Major-07a16060613c408b5?role=publish_subscribe"
}
```

Untuk informasi selengkapnya, lihat [Menginstal Plugin Session Manager untuk AWS CLI](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [ResumeSession](#) di Referensi AWS CLI Perintah.

## send-automation-signal

Contoh kode berikut menunjukkan cara menggunakan `send-automation-signal`.

### AWS CLI

Untuk mengirim sinyal ke eksekusi otomatisasi

`send-automation-signal` Contoh berikut mengirimkan sinyal Approve ke eksekusi Automation.

```
aws ssm send-automation-signal \
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
  --signal-type "Approve"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menjalankan Alur Kerja Otomasi dengan Penyetuju](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [SendAutomationSignal](#) di Referensi AWS CLI Perintah.

## send-command

Contoh kode berikut menunjukkan cara menggunakan `send-command`.

### AWS CLI

Contoh 1: Untuk menjalankan perintah pada satu atau beberapa instance jarak jauh

`send-command` Contoh berikut menjalankan `echo` perintah pada instance target.

```
aws ssm send-command \  
  --document-name "AWS-RunShellScript" \  
  --parameters 'commands=["echo HelloWorld"]' \  
  --targets "Key=instanceids,Values=i-1234567890abcdef0" \  
  --comment "echo HelloWorld"
```

### Output:

```
{  
  "Command": {  
    "CommandId": "92853adf-ba41-4cd6-9a88-142d1EXAMPLE",  
    "DocumentName": "AWS-RunShellScript",  
    "DocumentVersion": "",  
    "Comment": "echo HelloWorld",  
    "ExpiresAfter": 1550181014.717,  
    "Parameters": {  
      "commands": [  
        "echo HelloWorld"  
      ]  
    },  
    "InstanceIds": [  
      "i-0f00f008a2dcbef2"  
    ],  
    "Targets": [],  
    "RequestedDateTime": 1550173814.717,  
  },  
}
```

```
"Status": "Pending",
"StatusDetails": "Pending",
"OutputS3BucketName": "",
"OutputS3KeyPrefix": "",
"MaxConcurrency": "50",
"MaxErrors": "0",
"TargetCount": 1,
"CompletedCount": 0,
"ErrorCount": 0,
"DeliveryTimedOutCount": 0,
"ServiceRole": "",
"NotificationConfig": {
  "NotificationArn": "",
  "NotificationEvents": [],
  "NotificationType": ""
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Ujian 2: Untuk mendapatkan informasi IP tentang sebuah instance

send-command Contoh berikut mengambil informasi IP tentang sebuah instance.

```
aws ssm send-command \  
  --instance-ids "i-1234567890abcdef0" \  
  --document-name "AWS-RunShellScript" \  
  --comment "IP config" \  
  --parameters "commands=ifconfig"
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Contoh 3: Untuk menjalankan perintah pada instance dengan tag tertentu

send-commandContoh berikut menjalankan perintah pada instance yang memiliki kunci tag “ENV” dan nilai “Dev”.

```
aws ssm send-command \  
  --targets "Key=tag:ENV,Values=Dev" \  
  --document-name "AWS-RunShellScript" \  
  --parameters "commands=ifconfig"
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Contoh 4: Untuk menjalankan perintah yang mengirimkan notifikasi SNS

send-commandContoh berikut menjalankan perintah yang mengirimkan notifikasi SNS untuk semua peristiwa notifikasi dan jenis Command notifikasi.

```
aws ssm send-command \  
  --instance-ids "i-1234567890abcdef0" \  
  --document-name "AWS-RunShellScript" \  
  --comment "IP config" \  
  --parameters "commands=ifconfig" \  
  --service-role-arn "arn:aws:iam::123456789012:role/SNS_Role" \  
  --notification-config "NotificationArn=arn:aws:sns:us-  
east-1:123456789012:SNSTopicName,NotificationEvents=All,NotificationType=Command"
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Contoh 5: Untuk menjalankan perintah yang mengeluarkan ke S3 dan CloudWatch

send-commandContoh berikut menjalankan perintah yang mengeluarkan detail perintah ke bucket S3 dan ke grup CloudWatch log Log.

```
aws ssm send-command \  
  --instance-ids "i-1234567890abcdef0" \  
  --document-name "AWS-RunShellScript" \  
  --comment "IP config" \  
  --parameters "commands=ifconfig"
```

```
--parameters "commands=ifconfig" \  
--output-s3-bucket-name "s3-bucket-name" \  
--output-s3-key-prefix "runcommand" \  
--cloud-watch-output-config  
"CloudWatchOutputEnabled=true,CloudWatchLogGroupName=CWLGroupName"
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Contoh 6: Untuk menjalankan perintah pada beberapa instance dengan tag yang berbeda

send-commandContoh berikut menjalankan perintah pada instance dengan dua kunci tag dan nilai yang berbeda.

```
aws ssm send-command \  
  --document-name "AWS-RunPowerShellScript" \  
  --parameters commands=["echo helloWorld"] \  
  --targets Key=tag:Env,Values=Dev Key=tag:Role,Values=WebServers
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Contoh 7: Untuk menargetkan beberapa instance dengan kunci tag yang sama

send-commandContoh berikut menjalankan perintah pada instance yang memiliki kunci tag yang sama tetapi dengan nilai yang berbeda.

```
aws ssm send-command \  
  --document-name "AWS-RunPowerShellScript" \  
  --parameters commands=["echo helloWorld"] \  
  --targets Key=tag:Env,Values=Dev,Test
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menjalankan Perintah Menggunakan Systems Manager Run Command](#) di Panduan Pengguna AWS Systems Manager.

Contoh 8: Untuk menjalankan perintah yang menggunakan dokumen bersama

`send-command` Contoh berikut menjalankan dokumen bersama pada instance target.

```
aws ssm send-command \  
  --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument" \  
  --targets "Key=instanceids,Values=i-1234567890abcdef0"
```

Lihat contoh 1 untuk output sampel.

Untuk informasi selengkapnya, lihat [Menggunakan dokumen SSM bersama](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [SendCommand](#) di Referensi AWS CLI Perintah.

## **start-associations-once**

Contoh kode berikut menunjukkan cara menggunakan `start-associations-once`.

### AWS CLI

Untuk menjalankan asosiasi segera dan hanya satu kali

`start-associations-once` Contoh berikut menjalankan asosiasi yang ditentukan segera dan hanya sekali. Tidak ada output jika perintah berhasil.

```
aws ssm start-associations-once \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Melihat riwayat asosiasi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [StartAssociationsOnce](#) di Referensi AWS CLI Perintah.

## **start-automation-execution**

Contoh kode berikut menunjukkan cara menggunakan `start-automation-execution`.

### AWS CLI

Contoh 1: Untuk menjalankan dokumen otomatisasi

`start-automation-execution` Contoh berikut menjalankan dokumen Automation.

```
aws ssm start-automation-execution \  
  --document-name "AWS-UpdateLinuxAmi" \  
  --parameters "AutomationAssumeRole=arn:aws:iam::123456789012:role/  
SSMAutomationRole,SourceAmiId=ami-EXAMPLE,IamInstanceProfileName=EC2InstanceRole"
```

Output:

```
{  
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menjalankan Alur Kerja Otomasi Secara Manual](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk menjalankan dokumen otomasi bersama

`start-automation-execution` Contoh berikut menjalankan dokumen Otomasi bersama.

```
aws ssm start-automation-execution \  
  --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
```

Output:

```
{  
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"  
}
```

Untuk informasi selengkapnya, lihat [Menggunakan dokumen SSM bersama](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [StartAutomationExecution](#) di Referensi AWS CLI Perintah.

## **start-change-request-execution**

Contoh kode berikut menunjukkan cara menggunakan `start-change-request-execution`.

### AWS CLI

Contoh 1: Untuk memulai permintaan perubahan

`start-change-request-execution` Contoh berikut memulai permintaan perubahan dengan opsi minimal yang ditentukan.

```
aws ssm start-change-request-execution \
  --change-request-name MyChangeRequest \
  --document-name AWS-HelloWorldChangeTemplate \
  --runbooks '[{"DocumentName": "AWS-HelloWorld", "Parameters":
  {"AutomationAssumeRole": ["arn:aws:iam:us-east-2:1112223233444:role/
  MyChangeManagerAssumeRole"]}]}]' \
  --parameters
  Approver="JohnDoe",ApproverType="IamUser",ApproverSnsTopicArn="arn:aws:sns:us-
  east-2:1112223233444:MyNotificationTopic"
```

Output:

```
{
  "AutomationExecutionId": "9d32a4fc-f944-11e6-4105-0a1b2EXAMPLE"
}
```

Contoh 2: Untuk memulai permintaan perubahan menggunakan file JSON eksternal

`start-automation-execution` Contoh berikut memulai permintaan perubahan dengan beberapa opsi yang ditentukan dalam file JSON.

```
aws ssm start-change-request-execution \
  --cli-input-json file://MyChangeRequest.json
```

Isi dari `MyChangeRequest.json`:

```
{
  "ChangeRequestName": "MyChangeRequest",
  "DocumentName": "AWS-HelloWorldChangeTemplate",
  "DocumentVersion": "$DEFAULT",
  "ScheduledTime": "2021-12-30T03:00:00",
  "ScheduledEndTime": "2021-12-30T03:05:00",
  "Tags": [
    {
      "Key": "Purpose",
      "Value": "Testing"
    }
  ],
}
```



```

"Parameters": {
  "Approver": [
    "JohnDoe"
  ],
  "ApproverType": [
    "IamUser"
  ],
  "ApproverSnsTopicArn": [
    "arn:aws:sns:us-east-2:111222333444:MyNotificationTopic"
  ]
},
"Runbooks": [
  {
    "DocumentName": "AWS-HelloWorld",
    "DocumentVersion": "1",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Parameters": {
      "AutomationAssumeRole": [
        "arn:aws:iam::111222333444:role/MyChangeManagerAssumeRole"
      ]
    }
  }
],
"ChangeDetails": "### Document Name: HelloWorldChangeTemplate\n\n## What does this document do?\n\nThis change template demonstrates the feature set available for creating change templates for Change Manager. This template starts a Runbook workflow for the Automation document called AWS-HelloWorld.\n\n## Input Parameters\n\n* ApproverSnsTopicArn: (Required) Amazon Simple Notification Service ARN for approvers.\n* Approver: (Required) The name of the approver to send this request to.\n* ApproverType: (Required) The type of reviewer.\n  * Allowed Values: IamUser, IamGroup, IamRole, SSOGroup, SSOUser\n\n## Output Parameters\n\nThis document has no outputs \n"
}

```

### Output:

```

{
  "AutomationExecutionId": "9d32a4fc-f944-11e6-4105-0a1b2EXAMPLE"
}

```

Untuk informasi selengkapnya, lihat [Membuat permintaan perubahan](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [StartChangeRequestExecution](#) di Referensi AWS CLI Perintah.

## start-session

Contoh kode berikut menunjukkan cara menggunakan `start-session`.

### AWS CLI

Contoh 1: Untuk memulai sesi Manajer Sesi

`start-session` Contoh ini membuat koneksi dengan instance untuk sesi Session Manager. Perhatikan bahwa perintah interaktif ini memerlukan plugin Session Manager untuk diinstal pada mesin klien yang melakukan panggilan.

```
aws ssm start-session \  
  --target "i-1234567890abcdef0"
```

Output:

```
Starting session with SessionId: Jane-Roe-07a16060613c408b5
```

Contoh 2: Untuk memulai sesi Session Manager menggunakan SSH

`start-session` Contoh ini membuat koneksi dengan instance untuk sesi Session Manager menggunakan SSH. Perhatikan bahwa perintah interaktif ini memerlukan plugin Session Manager untuk diinstal pada mesin klien yang membuat panggilan, dan bahwa perintah tersebut menggunakan pengguna default pada instance, seperti `ec2-user` untuk instans EC2 untuk Linux.

```
ssh -i /path/my-key-pair.pem ec2-user@i-02573cafcfEXAMPLE
```

Output:

```
Starting session with SessionId: ec2-user-07a16060613c408b5
```

Untuk informasi selengkapnya, lihat [Memulai Sesi](#) dan [Menginstal Plugin Session Manager untuk AWS CLI](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [StartSession](#) di Referensi AWS CLI Perintah.

## stop-automation-execution

Contoh kode berikut menunjukkan cara menggunakan `stop-automation-execution`.

### AWS CLI

Untuk menghentikan eksekusi otomatisasi

`stop-automation-execution` Contoh berikut menghentikan dokumen Otomasi.

```
aws ssm stop-automation-execution
  --automation-execution-id "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menjalankan Alur Kerja Otomasi Secara Manual](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [StopAutomationExecution](#) di Referensi AWS CLI Perintah.

## terminate-session

Contoh kode berikut menunjukkan cara menggunakan `terminate-session`.

### AWS CLI

Untuk mengakhiri sesi Manajer Sesi

`terminate-session` Contoh ini secara permanen mengakhiri sesi yang dibuat oleh pengguna "Shirley-Rodriguez" dan menutup koneksi data antara klien Session Manager dan SSM Agent pada instance.

```
aws ssm terminate-session \
  --session-id "Shirley-Rodriguez-07a16060613c408b5"
```

Output:

```
{
  "SessionId": "Shirley-Rodriguez-07a16060613c408b5"
}
```

Untuk informasi selengkapnya, lihat [Mengakhiri Sesi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [TerminateSession](#) di Referensi AWS CLI Perintah.

## **unlabel-parameter-version**

Contoh kode berikut menunjukkan cara menggunakan `unlabel-parameter-version`.

### AWS CLI

Untuk menghapus label parameter

`unlabel-parameter-version` Contoh berikut menghapus label yang ditentukan dari versi parameter yang diberikan.

```
aws ssm unlabel-parameter-version \  
  --name "parameterName" \  
  --parameter-version "version" \  
  --labels "label_1" "label_2" "label_3"
```

Output:

```
{  
  "RemovedLabels": [  
    "label_1"  
    "label_2"  
    "label_3"  
  ],  
  "InvalidLabels": []  
}
```

Untuk informasi selengkapnya, lihat [Menghapus label parameter \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UnlabelParameterVersion](#) di Referensi AWS CLI Perintah.

## **update-association-status**

Contoh kode berikut menunjukkan cara menggunakan `update-association-status`.

## AWS CLI

Untuk memperbarui status asosiasi

update-association-status Contoh berikut memperbarui status asosiasi asosiasi antara instance dan dokumen.

```
aws ssm update-association-status \
  --name "AWS-UpdateSSMAgent" \
  --instance-id "i-1234567890abcdef0" \
  --association-status
  "Date=1424421071.939,Name=Pending,Message=temp_status_change,AdditionalInfo=Additional-
  Config-Needed"
```

Output:

```
{
  "AssociationDescription": {
    "Name": "AWS-UpdateSSMAgent",
    "InstanceId": "i-1234567890abcdef0",
    "AssociationVersion": "1",
    "Date": 1550507529.604,
    "LastUpdateAssociationDate": 1550507806.974,
    "Status": {
      "Date": 1424421071.0,
      "Name": "Pending",
      "Message": "temp_status_change",
      "AdditionalInfo": "Additional-Config-Needed"
    },
    "Overview": {
      "Status": "Success",
      "AssociationStatusAggregatedCount": {
        "Success": 1
      }
    },
    "DocumentVersion": "$DEFAULT",
    "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
    "Targets": [
      {
        "Key": "InstanceIds",
        "Values": [
          "i-1234567890abcdef0"
        ]
      }
    ]
  }
}
```

```
    }
  ],
  "LastExecutionDate": 1550507808.0,
  "LastSuccessfulExecutionDate": 1550507808.0
}
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan asosiasi di Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateAssociationStatus](#) di Referensi AWS CLI Perintah.

## update-association

Contoh kode berikut menunjukkan cara menggunakan `update-association`.

### AWS CLI

Contoh 1: Untuk memperbarui asosiasi dokumen

`update-association` Contoh berikut memperbarui asosiasi dengan versi dokumen baru.

```
aws ssm update-association \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --document-version "$LATEST"
```

Output:

```
{
  "AssociationDescription": {
    "Name": "AWS-UpdateSSMAgent",
    "AssociationVersion": "2",
    "Date": 1550508093.293,
    "LastUpdateAssociationDate": 1550508106.596,
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Creating"
    }
  },
  "DocumentVersion": "$LATEST",
  "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
  "Targets": [
    {
```

```

        "Key": "tag:Name",
        "Values": [
            "Linux"
        ]
    },
    "LastExecutionDate": 1550508094.879,
    "LastSuccessfulExecutionDate": 1550508094.879
}
}

```

Untuk informasi selengkapnya, lihat [Mengedit dan membuat versi baru asosiasi](#) di Panduan Pengguna AWS Systems Manager.

Contoh 2: Untuk memperbarui ekspresi jadwal asosiasi

update-association Contoh berikut memperbarui ekspresi jadwal untuk asosiasi tertentu.

```

aws ssm update-association \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --schedule-expression "cron(0 0 0/4 1/1 * ? *)"

```

Output:

```

{
  "AssociationDescription": {
    "Name": "AWS-HelloWorld",
    "AssociationVersion": "2",
    "Date": "2021-02-08T13:54:19.203000-08:00",
    "LastUpdateAssociationDate": "2021-06-29T11:51:07.933000-07:00",
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Creating"
    }
  },
  "DocumentVersion": "$DEFAULT",
  "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
  "Targets": [
    {
      "Key": "aws:NoOpAutomationTag",
      "Values": [
        "AWS-NoOpAutomationTarget-Value"
      ]
    }
  ]
}

```

```
    }
  ],
  "ScheduleExpression": "cron(0 0 0/4 1/1 * ? *)",
  "LastExecutionDate": "2021-06-26T19:00:48.110000-07:00",
  "ApplyOnlyAtCronInterval": false
}
}
```

Untuk informasi selengkapnya, lihat [Mengedit dan membuat versi baru asosiasi](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateAssociation](#) di Referensi AWS CLI Perintah.

## update-document-default-version

Contoh kode berikut menunjukkan cara menggunakan `update-document-default-version`.

### AWS CLI

Untuk memperbarui versi default dokumen

`update-document-default-version` Contoh berikut memperbarui versi default dari dokumen Systems Manager.

```
aws ssm update-document-default-version \
  --name "Example" \
  --document-version "2"
```

Output:

```
{
  "Description": {
    "Name": "Example",
    "DefaultVersion": "2"
  }
}
```

Untuk informasi selengkapnya, lihat [Menulis Konten Dokumen SSM](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateDocumentDefaultVersion](#) di Referensi AWS CLI Perintah.



## update-document-metadata

Contoh kode berikut menunjukkan cara menggunakan `update-document-metadata`.

### AWS CLI

Contoh: Untuk menyetujui versi terbaru dari template perubahan

Berikut ini `update-document-metadata` memberikan persetujuan untuk versi terbaru dari template perubahan yang telah dikirimkan untuk ditinjau.

```
aws ssm update-document-metadata \  
  --name MyChangeManagerTemplate \  
  --document-reviews 'Action=Approve,Comment=[{Type=Comment,Content=Approved!}]'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Meninjau dan menyetujui atau menolak templat perubahan di Panduan Pengguna AWS Systems Manager](#).

- Untuk detail API, lihat [UpdateDocumentMetadata](#) di Referensi AWS CLI Perintah.

## update-document

Contoh kode berikut menunjukkan cara menggunakan `update-document`.

### AWS CLI

Untuk membuat versi baru dokumen

`update-document` Contoh berikut membuat versi baru dokumen saat dijalankan di komputer Windows. Dokumen yang ditentukan oleh `--document` harus dalam format JSON. Perhatikan bahwa `file://` harus direferensikan diikuti oleh jalur file konten. Karena di awal `--document-version` parameter, \$ Pada Windows Anda harus mengelilingi nilai dengan tanda kutip ganda. Di Linux, macOS, atau pada PowerShell prompt, Anda harus mengelilingi nilainya dengan tanda kutip tunggal.

Versi Windows:

```
aws ssm update-document \  
  --document file://C:\temp\mytemplate.json \  
  --document-version 1
```

```
--name "RunShellScript" \  
--content "file://RunShellScript.json" \  
--document-version "$LATEST"
```

### Versi Linux/Mac:

```
aws ssm update-document \  
  --name "RunShellScript" \  
  --content "file://RunShellScript.json" \  
  --document-version '$LATEST'
```

### Output:

```
{  
  "DocumentDescription": {  
    "Status": "Updating",  
    "Hash": "f775e5df4904c6fa46686c4722fae9de1950dace25cd9608ff8d622046b68d9b",  
    "Name": "RunShellScript",  
    "Parameters": [  
      {  
        "Type": "StringList",  
        "Name": "commands",  
        "Description": "(Required) Specify a shell script or a command to  
run."  
      }  
    ],  
    "DocumentType": "Command",  
    "PlatformTypes": [  
      "Linux"  
    ],  
    "DocumentVersion": "2",  
    "HashType": "Sha256",  
    "CreateDate": 1487899655.152,  
    "Owner": "809632081692",  
    "SchemaVersion": "2.0",  
    "DefaultVersion": "1",  
    "LatestVersion": "2",  
    "Description": "Run an updated script"  
  }  
}
```

- Untuk detail API, lihat [UpdateDocument](#) di Referensi AWS CLI Perintah.

## update-maintenance-window-target

Contoh kode berikut menunjukkan cara menggunakan `update-maintenance-window-target`.

### AWS CLI

Untuk memperbarui target jendela pemeliharaan

`update-maintenance-window-target` Contoh berikut hanya memperbarui nama target jendela pemeliharaan.

```
aws ssm update-maintenance-window-target \
  --window-id "mw-0c5ed765acEXAMPLE" \
  --window-target-id "57e8344e-fe64-4023-8191-6bf05EXAMPLE" \
  --name "NewName" \
  --no-replace
```

Output:

```
{
  "Description": "",
  "OwnerInformation": "",
  "WindowTargetId": "57e8344e-fe64-4023-8191-6bf05EXAMPLE",
  "WindowId": "mw-0c5ed765acEXAMPLE",
  "Targets": [
    {
      "Values": [
        "i-1234567890EXAMPLE"
      ],
      "Key": "InstanceIds"
    }
  ],
  "Name": "NewName"
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateMaintenanceWindowTarget](#) di Referensi AWS CLI Perintah.

## update-maintenance-window-task

Contoh kode berikut menunjukkan cara menggunakan `update-maintenance-window-task`.

## AWS CLI

Untuk memperbarui tugas jendela pemeliharaan

update-maintenance-window-task Contoh berikut memperbarui peran layanan untuk tugas jendela pemeliharaan.

```
aws ssm update-maintenance-window-task \  
  --window-id "mw-0c5ed765acEXAMPLE" \  
  --window-task-id "23d3809e-9fbe-4ddf-b41a-b49d7EXAMPLE" \  
  --service-role-arn "arn:aws:iam::111222333444:role/aws-service-role/  
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM"
```

Output:

```
{  
  "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/  
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",  
  "MaxErrors": "1",  
  "TaskArn": "AWS-UpdateEC2Config",  
  "MaxConcurrency": "1",  
  "WindowTaskId": "23d3809e-9fbe-4ddf-b41a-b49d7EXAMPLE",  
  "TaskParameters": {},  
  "Priority": 1,  
  "TaskInvocationParameters": {  
    "RunCommand": {  
      "TimeoutSeconds": 600,  
      "Parameters": {  
        "allowDowngrade": [  
          "false"  
        ]  
      }  
    }  
  },  
  "WindowId": "mw-0c5ed765acEXAMPLE",  
  "Description": "UpdateEC2Config",  
  "Targets": [  
    {  
      "Values": [  
        "57e8344e-fe64-4023-8191-6bf05EXAMPLE"  
      ],  
      "Key": "WindowTargetIds"  
    }  
  ]  
}
```

```
  ],  
  "Name": "UpdateEC2Config"  
}
```

Untuk informasi selengkapnya, lihat [Memperbarui Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateMaintenanceWindowTask](#) di Referensi AWS CLI Perintah.

## update-maintenance-window

Contoh kode berikut menunjukkan cara menggunakan `update-maintenance-window`.

### AWS CLI

Contoh 1: Untuk memperbarui jendela pemeliharaan

`update-maintenance-window` Contoh berikut memperbarui nama jendela pemeliharaan.

```
aws ssm update-maintenance-window \  
  --window-id "mw-1a2b3c4d5e6f7g8h9" \  
  --name "My-Renamed-MW"
```

Output:

```
{  
  "Cutoff": 1,  
  "Name": "My-Renamed-MW",  
  "Schedule": "cron(0 16 ? * TUE *)",  
  "Enabled": true,  
  "AllowUnassociatedTargets": true,  
  "WindowId": "mw-1a2b3c4d5e6f7g8h9",  
  "Duration": 4  
}
```

Contoh 2: Untuk menonaktifkan jendela pemeliharaan

`update-maintenance-window` Contoh berikut menonaktifkan jendela pemeliharaan.

```
aws ssm update-maintenance-window \  
  --window-id "mw-1a2b3c4d5e6f7g8h9" \  
  --no-enabled
```

### Contoh 3: Untuk mengaktifkan jendela pemeliharaan

`update-maintenance-window` Contoh berikut memungkinkan jendela pemeliharaan.

```
aws ssm update-maintenance-window \  
  --window-id "mw-1a2b3c4d5e6f7g8h9" \  
  --enabled
```

Untuk informasi selengkapnya, lihat [Memperbarui Jendela Pemeliharaan \(AWS CLI\)](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateMaintenanceWindow](#) di Referensi AWS CLI Perintah.

### `update-managed-instance-role`

Contoh kode berikut menunjukkan cara menggunakan `update-managed-instance-role`.

#### AWS CLI

Untuk memperbarui peran IAM dari instans terkelola

`update-managed-instance-role` Contoh berikut memperbarui profil instans IAM dari instance terkelola.

```
aws ssm update-managed-instance-role \  
  --instance-id "mi-08ab247cdfEXAMPLE" \  
  --iam-role "ExampleRole"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Langkah 4: Membuat Profil Instans IAM untuk Systems Manager](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateManagedInstanceRole](#) di Referensi AWS CLI Perintah.

### `update-ops-item`

Contoh kode berikut menunjukkan cara menggunakan `update-ops-item`.

#### AWS CLI

Untuk memperbarui OpsItem

`update-ops-item` Contoh berikut memperbarui deskripsi, prioritas, dan kategori untuk file `OpsItem`. Selain itu, perintah menentukan topik SNS tempat notifikasi dikirim saat ini `OpsItem` diedit atau diubah.

```
aws ssm update-ops-item \  
  --ops-item-id "oi-287b5EXAMPLE" \  
  --description "Primary OpsItem for failover event 2020-01-01-fh398yf" \  
  --priority 2 \  
  --category "Security" \  
  --notifications "Arn=arn:aws:sns:us-east-2:111222333444:my-us-east-2-topic"
```

Output:

```
This command produces no output.
```

Untuk informasi selengkapnya, lihat [Bekerja dengan OpsItems](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateOpsItem](#) di Referensi AWS CLI Perintah.

## update-patch-baseline

Contoh kode berikut menunjukkan cara menggunakan `update-patch-baseline`.

AWS CLI

Contoh 1: Untuk memperbarui baseline patch

`update-patch-baseline` Contoh berikut menambahkan dua patch yang ditentukan sebagai ditolak dan satu patch yang disetujui untuk baseline patch yang ditentukan.

```
aws ssm update-patch-baseline \  
  --baseline-id "pb-0123456789abcdef0" \  
  --rejected-patches "KB2032276" "MS10-048" \  
  --approved-patches "KB2124261"
```

Output:

```
{  
  "BaselineId": "pb-0123456789abcdef0",  
  "Name": "WindowsPatching",
```

```

"OperatingSystem": "WINDOWS",
"GlobalFilters": {
  "PatchFilters": []
},
"ApprovalRules": {
  "PatchRules": [
    {
      "PatchFilterGroup": {
        "PatchFilters": [
          {
            "Key": "PRODUCT",
            "Values": [
              "WindowsServer2016"
            ]
          }
        ]
      },
      "ComplianceLevel": "CRITICAL",
      "ApproveAfterDays": 0,
      "EnableNonSecurity": false
    }
  ]
},
"ApprovedPatches": [
  "KB2124261"
],
"ApprovedPatchesComplianceLevel": "UNSPECIFIED",
"ApprovedPatchesEnableNonSecurity": false,
"RejectedPatches": [
  "KB2032276",
  "MS10-048"
],
"RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
"CreateDate": 1550244180.465,
"ModifiedDate": 1550244180.465,
"Description": "Patches for Windows Servers",
"Sources": []
}

```

Contoh 2: Untuk mengganti nama baseline patch

update-patch-baselineContoh berikut mengganti nama baseline patch yang ditentukan.

```
aws ssm update-patch-baseline \
```



```
--baseline-id "pb-0713accee01234567" \  
--name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Untuk informasi selengkapnya, lihat [Memperbarui atau Menghapus Garis Dasar Patch](https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-baseline-update-or-delete.html) < <https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-baseline-update-or-delete.html>>`\_\_ di Panduan Pengguna Systems Manager.AWS

- Untuk detail API, lihat [UpdatePatchBaseline](#) di Referensi AWS CLI Perintah.

## update-resource-data-sync

Contoh kode berikut menunjukkan cara menggunakan `update-resource-data-sync`.

### AWS CLI

Untuk memperbarui sinkronisasi data sumber daya

`update-resource-data-sync` Contoh berikut memperbarui sinkronisasi data `SyncFromSource` sumber daya.

```
aws ssm update-resource-data-sync \  
  --sync-name exampleSync \  
  --sync-type SyncFromSource \  
  --sync-source '{"SourceType":"SingleAccountMultiRegions", "SourceRegions":["us-east-1", "us-west-2"]}'
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menyiapkan Systems Manager Explorer untuk Menampilkan Data dari Beberapa Akun dan Wilayah](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateResourceDataSync](#) di Referensi AWS CLI Perintah.

## update-service-setting

Contoh kode berikut menunjukkan cara menggunakan `update-service-setting`.

### AWS CLI

Untuk memperbarui setelan layanan untuk throughput Parameter Store

`update-service-setting` Contoh berikut memperbarui setelan layanan saat ini untuk throughput Parameter Store di wilayah tertentu untuk menggunakan peningkatan throughput.

```
aws ssm update-service-setting \  
  --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-  
store/high-throughput-enabled \  
  --setting-value true
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Meningkatkan Throughput Parameter Store](#) di Panduan Pengguna AWS Systems Manager.

- Untuk detail API, lihat [UpdateServiceSetting](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Texttract menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Amazon Textract. AWS Command Line Interface

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **analyze-document**

Contoh kode berikut menunjukkan cara menggunakan `analyze-document`.

AWS CLI

Untuk menganalisis teks dalam dokumen

`analyze-document` Contoh berikut menunjukkan cara menganalisis teks dalam dokumen.

## Linux/macOS:

```
aws textract analyze-document \  
  --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \  
  --feature-types ['TABLES','FORMS']
```

## Windows:

```
aws textract analyze-document \  
  --document "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\"}}" \  
  --feature-types [\"TABLES\", \"FORMS\"] \  
  --region region-name
```

## Output:

```
{  
  "Blocks": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 1.0,  
          "Top": 0.0,  
          "Left": 0.0,  
          "Height": 1.0  
        },  
        "Polygon": [  
          {  
            "Y": 0.0,  
            "X": 0.0  
          },  
          {  
            "Y": 0.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 0.0  
          }  
        ]  
      }  
    }  
  ]  
}
```

```

    ]
  },
  "Relationships": [
    {
      "Type": "CHILD",
      "Ids": [
        "87586964-d50d-43e2-ace5-8a890657b9a0",
        "a1e72126-21d9-44f4-a8d6-5c385f9002ba",
        "e889d012-8a6b-4d2e-b7cd-7a8b327d876a"
      ]
    }
  ],
  "BlockType": "PAGE",
  "Id": "c2227f12-b25d-4e1f-baea-1ee180d926b2"
}
],
"DocumentMetadata": {
  "Pages": 1
}
}

```

Untuk informasi selengkapnya, lihat [Menganalisis Teks Dokumen dengan Amazon Textract](#) di Panduan Pengembang Amazon Texttract

- Untuk detail API, lihat [AnalyzeDocument](#) di Referensi AWS CLI Perintah.

## detect-document-text

Contoh kode berikut menunjukkan cara menggunakan `detect-document-text`.

### AWS CLI

Untuk mendeteksi teks dalam dokumen

Berikut `detect-document-text` ini Contoh berikut menunjukkan cara mendeteksi teks dalam dokumen.

Linux/macOS:

```
aws textract detect-document-text \
  --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}'
```

Windows:

```
aws textract detect-document-text \  
  --document "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\"}}\" \  
  --region region-name
```

**Output:**

```
{  
  "Blocks": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 1.0,  
          "Top": 0.0,  
          "Left": 0.0,  
          "Height": 1.0  
        },  
        "Polygon": [  
          {  
            "Y": 0.0,  
            "X": 0.0  
          },  
          {  
            "Y": 0.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 0.0  
          }  
        ]  
      },  
      "Relationships": [  
        {  
          "Type": "CHILD",  
          "Ids": [  
            "896a9f10-9e70-4412-81ce-49ead73ed881",  
            "0da18623-dc4c-463d-a3d1-9ac050e9e720",  
            "167338d7-d38c-4760-91f1-79a8ec457bb2"  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```
    }
  ],
  "BlockType": "PAGE",
  "Id": "21f0535e-60d5-4bc7-adf2-c05dd851fa25"
},
{
  "Relationships": [
    {
      "Type": "CHILD",
      "Ids": [
        "62490c26-37ea-49fa-8034-7a9ff9369c9c",
        "1e4f3f21-05bd-4da9-ba10-15d01e66604c"
      ]
    }
  ]
},
  ],
  "Confidence": 89.11581420898438,
  "Geometry": {
    "BoundingBox": {
      "Width": 0.33642634749412537,
      "Top": 0.17169663310050964,
      "Left": 0.13885067403316498,
      "Height": 0.49159330129623413
    },
    "Polygon": [
      {
        "Y": 0.17169663310050964,
        "X": 0.13885067403316498
      },
      {
        "Y": 0.17169663310050964,
        "X": 0.47527703642845154
      },
      {
        "Y": 0.6632899641990662,
        "X": 0.47527703642845154
      },
      {
        "Y": 0.6632899641990662,
        "X": 0.13885067403316498
      }
    ]
  },
  "Text": "He llo,",
  "BlockType": "LINE",
```

```
    "Id": "896a9f10-9e70-4412-81ce-49ead73ed881"
  },
  {
    "Relationships": [
      {
        "Type": "CHILD",
        "Ids": [
          "19b28058-9516-4352-b929-64d7cef29daf"
        ]
      }
    ],
    "Confidence": 85.5694351196289,
    "Geometry": {
      "BoundingBox": {
        "Width": 0.33182239532470703,
        "Top": 0.23131252825260162,
        "Left": 0.5091826915740967,
        "Height": 0.3766750991344452
      },
      "Polygon": [
        {
          "Y": 0.23131252825260162,
          "X": 0.5091826915740967
        },
        {
          "Y": 0.23131252825260162,
          "X": 0.8410050868988037
        },
        {
          "Y": 0.607987642288208,
          "X": 0.8410050868988037
        },
        {
          "Y": 0.607987642288208,
          "X": 0.5091826915740967
        }
      ]
    },
    "Text": "worlc",
    "BlockType": "LINE",
    "Id": "0da18623-dc4c-463d-a3d1-9ac050e9e720"
  }
],
"DocumentMetadata": {
```

```
    "Pages": 1
  }
}
```

Untuk informasi selengkapnya, lihat Mendeteksi Teks Dokumen dengan Amazon Textract di Panduan Pengembang Amazon Textract

- Untuk detail API, lihat [DetectDocumentText](#) di Referensi AWS CLI Perintah.

## get-document-analysis

Contoh kode berikut menunjukkan cara menggunakan `get-document-analysis`.

### AWS CLI

Untuk mendapatkan hasil analisis teks asinkron dari dokumen multi-halaman

`get-document-analysis` Contoh berikut menunjukkan cara mendapatkan hasil analisis teks asinkron dari dokumen multi-halaman.

```
aws textract get-document-analysis \
  --job-id df7cf32ebbd2a5de113535fcf4d921926a701b09b4e7d089f3aebadb41e0712b \
  --max-results 1000
```

Output:

```
{
  "Blocks": [
    {
      "Geometry": {
        "BoundingBox": {
          "Width": 1.0,
          "Top": 0.0,
          "Left": 0.0,
          "Height": 1.0
        },
        "Polygon": [
          {
            "Y": 0.0,
            "X": 0.0
          },
          {
            "Y": 0.0,
```



```

        "X": 1.0
      },
      {
        "Y": 1.0,
        "X": 1.0
      },
      {
        "Y": 1.0,
        "X": 0.0
      }
    ]
  },
  "Relationships": [
    {
      "Type": "CHILD",
      "Ids": [
        "75966e64-81c2-4540-9649-d66ec341cd8f",
        "bb099c24-8282-464c-a179-8a9fa0a057f0",
        "5ebf522d-f9e4-4dc7-bfae-a288dc094595"
      ]
    }
  ],
  "BlockType": "PAGE",
  "Id": "247c28ee-b63d-4aeb-9af0-5f7ea8ba109e",
  "Page": 1
}
],
"NextToken": "cY1W3eTFvoB0cH7YrKVudI4Gb0H8J0xAYLo8xI/JunCIPWCthaKQ+07n/
ElyutsSy0+1VOImoTRmP1zw4P0RFtaeV9Bzhnfedpx1YqwB4xaGDA==",
"DocumentMetadata": {
  "Pages": 1
},
"JobStatus": "SUCCEEDED"
}

```

Untuk informasi selengkapnya, lihat Mendeteksi dan Menganalisis Teks dalam Dokumen Multi-Halaman di Panduan Pengembang Amazon Texttract

- Untuk detail API, lihat [GetDocumentAnalysis](#) di Referensi AWS CLI Perintah.

## get-document-text-detection

Contoh kode berikut menunjukkan cara menggunakan `get-document-text-detection`.

## AWS CLI

Untuk mendapatkan hasil deteksi teks asinkron dalam dokumen multi-halaman

`get-document-text-detection` Contoh berikut menunjukkan cara mendapatkan hasil deteksi teks asinkron dalam dokumen multi-halaman.

```
aws textract get-document-text-detection \  
  --job-id 57849a3dc627d4df74123dca269d69f7b89329c870c65bb16c9fd63409d200b9 \  
  --max-results 1000
```

### Output

```
{  
  "Blocks": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 1.0,  
          "Top": 0.0,  
          "Left": 0.0,  
          "Height": 1.0  
        },  
        "Polygon": [  
          {  
            "Y": 0.0,  
            "X": 0.0  
          },  
          {  
            "Y": 0.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 0.0  
          }  
        ]  
      },  
      "Relationships": [  

```

```

        {
            "Type": "CHILD",
            "Ids": [
                "1b926a34-0357-407b-ac8f-ec473160c6a9",
                "0c35dc17-3605-4c9d-af1a-d9451059df51",
                "dea3db8a-52c2-41c0-b50c-81f66f4aa758"
            ]
        },
        {
            "BlockType": "PAGE",
            "Id": "84671a5e-8c99-43be-a9d1-6838965da33e",
            "Page": 1
        }
    ],
    "NextToken": "GcqyoAJuZwuj0T35EN4LCI3EUzMtiLq3nKyFFHvU5q1SaIdEBcSty+njNgoWwuMP/
muqc96S4o5NzDqehhXvhkodMyV050JGyms5lsrCxibWJw==",
    "DocumentMetadata": {
        "Pages": 1
    },
    "JobStatus": "SUCCEEDED"
}

```

Untuk informasi selengkapnya, lihat Mendeteksi dan Menganalisis Teks dalam Dokumen Multi-Halaman di Panduan Pengembang Amazon Texttract

- Untuk detail API, lihat [GetDocumentTextDetection](#) di Referensi AWS CLI Perintah.

## start-document-analysis

Contoh kode berikut menunjukkan cara menggunakan `start-document-analysis`.

### AWS CLI

Untuk mulai menganalisis teks dalam dokumen multi-halaman

`start-document-analysis` Contoh berikut menunjukkan bagaimana memulai analisis asinkron teks dalam dokumen multi-halaman.

Linux/macOS:

```

aws textract start-document-analysis \
  --document-location '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
  --feature-types ['TABLES','FORMS'] \

```

```
--notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

### Windows:

```
aws textract start-document-analysis \
  --document-location "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\"}}\" \
  --feature-types "[\"TABLES\", \"FORMS\"]" \
  --region region-name \
  --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

### Output:

```
{
  "JobId": "df7cf32ebbd2a5de113535fcf4d921926a701b09b4e7d089f3aebadb41e0712b"
}
```

Untuk informasi selengkapnya, lihat Mendeteksi dan Menganalisis Teks dalam Dokumen Multi-Halaman di Panduan Pengembang Amazon Texttract

- Untuk detail API, lihat [StartDocumentAnalysis](#) di Referensi AWS CLI Perintah.

## start-document-text-detection

Contoh kode berikut menunjukkan cara menggunakan `start-document-text-detection`.

### AWS CLI

Untuk mulai mendeteksi teks dalam dokumen multi-halaman

`start-document-text-detection` Contoh berikut menunjukkan bagaimana memulai deteksi asinkron teks dalam dokumen multi-halaman.

### Linux/macOS:

```
aws textract start-document-text-detection \
  --document-location '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
  --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleARN"
```

### Windows:

```
aws textract start-document-text-detection \  
  --document-location "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\  
  \"}}" \  
  --region region-name \  
  --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Output:

```
{  
  "JobId": "57849a3dc627d4df74123dca269d69f7b89329c870c65bb16c9fd63409d200b9"  
}
```

Untuk informasi selengkapnya, lihat Mendeteksi dan Menganalisis Teks dalam Dokumen Multi-Halaman di Panduan Pengembang Amazon Texttract

- Untuk detail API, lihat [StartDocumentTextDetection](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Transcribe menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum menggunakan AWS Command Line Interface with Amazon Transcribe.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-language-model**

Contoh kode berikut menunjukkan cara menggunakan `create-language-model`.

## AWS CLI

Contoh 1: Untuk membuat model bahasa khusus menggunakan data pelatihan dan penyetelan.

`create-language-model` Contoh berikut membuat model bahasa kustom. Anda dapat menggunakan model bahasa khusus untuk meningkatkan kinerja transkripsi untuk domain seperti hukum, perhotelan, keuangan, dan asuransi. Untuk kode bahasa, masukkan kode bahasa yang valid. Untuk `base-model-name`, tentukan model dasar yang paling cocok untuk laju sampel audio yang ingin Anda transkripsikan dengan model bahasa khusus Anda. Untuk `model-name`, tentukan nama yang ingin Anda panggil model bahasa kustom.

```
aws transcribe create-language-model \
  --language-code language-code \
  --base-model-name base-model-name \
  --model-name cli-clm-example \
  --input-data-config S3Uri="s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-for-
training-data",TuningDataS3Uri="s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-for-
tuning-data",DataAccessRoleArn="arn:aws:iam::AWS-account-number:role/IAM-role-with-
permissions-to-create-a-custom-language-model"
```

Output:

```
{
  "LanguageCode": "language-code",
  "BaseModelName": "base-model-name",
  "ModelName": "cli-clm-example",
  "InputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",
    "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",
    "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-with-
permissions-create-a-custom-language-model"
  },
  "ModelStatus": "IN_PROGRESS"
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Akurasi Transkripsi Khusus Domain dengan Model Bahasa Khusus](#) di Panduan Pengembang Amazon Transcribe.

Contoh 2: Untuk membuat model bahasa kustom hanya menggunakan data pelatihan.

`create-language-model` Contoh berikut mentranskripsikan file audio Anda. Anda dapat menggunakan model bahasa khusus untuk meningkatkan kinerja transkripsi untuk domain seperti

hukum, perhotelan, keuangan, dan asuransi. Untuk kode bahasa, masukkan kode bahasa yang valid. Untuk base-model-name, tentukan model dasar yang paling cocok untuk laju sampel audio yang ingin Anda transkripsikan dengan model bahasa khusus Anda. Untuk model-name, tentukan nama yang ingin Anda panggil model bahasa kustom.

```
aws transcribe create-language-model \
  --language-code en-US \
  --base-model-name base-model-name \
  --model-name cli-clm-example \
  --input-data-config S3Uri="s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-For-
  Training-Data",DataAccessRoleArn="arn:aws:iam::AWS-account-number:role/IAM-role-
  with-permissions-to-create-a-custom-language-model"
```

Output:

```
{
  "LanguageCode": "en-US",
  "BaseModelName": "base-model-name",
  "ModelName": "cli-clm-example",
  "InputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-For-Training-Data/",
    "DataAccessRoleArn": "arn:aws:iam::your-AWS-account-number:role/IAM-role-
    with-permissions-to-create-a-custom-language-model"
  },
  "ModelStatus": "IN_PROGRESS"
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Akurasi Transkripsi Khusus Domain dengan Model Bahasa Khusus](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [CreateLanguageModel](#) di Referensi AWS CLI Perintah.

## create-medical-vocabulary

Contoh kode berikut menunjukkan cara menggunakan `create-medical-vocabulary`.

### AWS CLI

Untuk membuat kosakata khusus medis

`create-medical-vocabulary` Contoh berikut membuat kosakata kustom. Untuk membuat kosakata khusus, Anda harus membuat file teks dengan semua istilah yang ingin Anda

transkripsikan dengan lebih akurat. Untuk `vocabulary-file-uri`, tentukan URI Amazon Simple Storage Service (Amazon S3) dari file teks tersebut. Untuk kode bahasa, tentukan kode bahasa yang sesuai dengan bahasa kosakata kustom Anda. Untuk nama kosakata, tentukan apa yang ingin Anda sebut kosakata khusus Anda.

```
aws transcribe create-medical-vocabulary \  
  --vocabulary-name cli-medical-vocab-example \  
  --language-code language-code \  
  --vocabulary-file-uri https://DOC-EXAMPLE-BUCKET.AWS-Region.amazonaws.com/the-  
text-file-for-the-medical-custom-vocabulary.txt
```

Output:

```
{  
  "VocabularyName": "cli-medical-vocab-example",  
  "LanguageCode": "language-code",  
  "VocabularyState": "PENDING"  
}
```

Untuk informasi selengkapnya, lihat [Kosakata Kustom Medis di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [CreateMedicalVocabulary](#) di Referensi AWS CLI Perintah.

## **create-vocabulary-filter**

Contoh kode berikut menunjukkan cara menggunakan `create-vocabulary-filter`.

### AWS CLI

Untuk membuat filter kosakata

`create-vocabulary-filter` Contoh berikut membuat filter kosakata yang menggunakan file teks yang berisi daftar kata yang tidak ingin Anda tampilkan dalam transkripsi. Untuk kode bahasa, tentukan kode bahasa yang sesuai dengan bahasa filter kosakata Anda. Untuk `vocabulary-filter-file-uri`, tentukan URI Amazon Simple Storage Service (Amazon S3) dari file teks. Untuk `vocabulary-filter-name`, tentukan nama filter kosakata Anda.

```
aws transcribe create-vocabulary-filter \  
  --language-code language-code \  
  --vocabulary-filter-file-uri s3://DOC-EXAMPLE-BUCKET/vocabulary-filter.txt \  
  --vocabulary-filter-name vocabulary-filter
```



```
--vocabulary-filter-name cli-vocabulary-filter-example
```

Output:

```
{
  "VocabularyFilterName": "cli-vocabulary-filter-example",
  "LanguageCode": "language-code"
}
```

Untuk informasi selengkapnya, lihat [Memfilter Kata yang Tidak Diinginkan](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [CreateVocabularyFilter](#) di Referensi AWS CLI Perintah.

## create-vocabulary

Contoh kode berikut menunjukkan cara menggunakan `create-vocabulary`.

### AWS CLI

Untuk membuat kosakata khusus

`create-vocabulary` Contoh berikut membuat kosakata kustom. Untuk membuat kosakata khusus, Anda harus membuat file teks dengan semua istilah yang ingin Anda transkripsikan dengan lebih akurat. Untuk `vocabulary-file-uri`, tentukan URI Amazon Simple Storage Service (Amazon S3) dari file teks tersebut. Untuk kode bahasa, tentukan kode bahasa yang sesuai dengan bahasa kosakata kustom Anda. Untuk nama kosakata, tentukan apa yang ingin Anda sebut kosakata khusus Anda.

```
aws transcribe create-vocabulary \
  --language-code language-code \
  --vocabulary-name cli-vocab-example \
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/the-text-file-
for-the-custom-vocabulary.txt
```

Output:

```
{
  "VocabularyName": "cli-vocab-example",
  "LanguageCode": "language-code",
  "VocabularyState": "PENDING"
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Kosakata Khusus di Panduan Pengembang Amazon Transcribe](#).

- Untuk detail API, lihat [CreateVocabulary](#) di Referensi AWS CLI Perintah.

## **delete-language-model**

Contoh kode berikut menunjukkan cara menggunakan `delete-language-model`.

### AWS CLI

Untuk menghapus model bahasa kustom

`delete-language-model` Contoh berikut menghapus model bahasa kustom.

```
aws transcribe delete-language-model \  
  --model-name model-name
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Meningkatkan Akurasi Transkripsi Khusus Domain dengan Model Bahasa Khusus](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DeleteLanguageModel](#) di Referensi AWS CLI Perintah.

## **delete-medical-transcription-job**

Contoh kode berikut menunjukkan cara menggunakan `delete-medical-transcription-job`.

### AWS CLI

Untuk menghapus pekerjaan transkripsi medis

`delete-medical-transcription-job` Contoh berikut menghapus pekerjaan transkripsi medis.

```
aws transcribe delete-medical-transcription-job \  
  --medical-transcription-job-name medical-transcription-job-name
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteMedicalTranscriptionJob](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DeleteMedicalTranscriptionJob](#) di Referensi AWS CLI Perintah.

## **delete-medical-vocabulary**

Contoh kode berikut menunjukkan cara menggunakan `delete-medical-vocabulary`.

### AWS CLI

Untuk menghapus kosakata khusus medis

`delete-medical-vocabulary` Contoh berikut menghapus kosakata kustom medis. Untuk nama kosakata, tentukan nama kosakata khusus medis.

```
aws transcribe delete-vocabulary \  
  --vocabulary-name medical-custom-vocabulary-name
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kosakata Kustom Medis di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DeleteMedicalVocabulary](#) di Referensi AWS CLI Perintah.

## **delete-transcription-job**

Contoh kode berikut menunjukkan cara menggunakan `delete-transcription-job`.

### AWS CLI

Untuk menghapus salah satu pekerjaan transkripsi Anda

`delete-transcription-job` Contoh berikut menghapus salah satu pekerjaan transkripsi Anda.

```
aws transcribe delete-transcription-job \  
  --transcription-job-name your-transcription-job
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [DeleteTranscriptionJob](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DeleteTranscriptionJob](#) di Referensi AWS CLI Perintah.

## **delete-vocabulary-filter**

Contoh kode berikut menunjukkan cara menggunakan `delete-vocabulary-filter`.

### AWS CLI

Untuk menghapus filter kosakata

`delete-vocabulary-filter` Contoh berikut menghapus filter kosakata.

```
aws transcribe delete-vocabulary-filter \  
  --vocabulary-filter-name vocabulary-filter-name
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memfilter Kata yang Tidak Diinginkan](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DeleteVocabularyFilter](#) di Referensi AWS CLI Perintah.

## **delete-vocabulary**

Contoh kode berikut menunjukkan cara menggunakan `delete-vocabulary`.

### AWS CLI

Untuk menghapus kosakata kustom

`delete-vocabulary` Contoh berikut menghapus kosakata kustom.

```
aws transcribe delete-vocabulary \  
  --vocabulary-name vocabulary-name
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kosakata Khusus di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DeleteVocabulary](#) di Referensi AWS CLI Perintah.

## describe-language-model

Contoh kode berikut menunjukkan cara menggunakan `describe-language-model`.

### AWS CLI

Untuk mendapatkan informasi tentang model bahasa khusus tertentu

`describe-language-model` Contoh berikut mendapatkan informasi tentang model bahasa kustom tertentu. Misalnya, di bawah `BaseModelName` Anda dapat melihat apakah model Anda dilatih menggunakan `WideBand` model `NarrowBand` atau. Model bahasa khusus dengan model `NarrowBand` dasar dapat mentranskripsikan audio dengan laju sampel kurang dari 16 kHz. Model bahasa yang menggunakan model `WideBand` dasar dapat mentranskripsikan audio dengan laju sampel lebih besar dari 16 kHz. Parameter `S3Uri` menunjukkan awalan Amazon S3 yang Anda gunakan untuk mengakses data pelatihan untuk membuat model bahasa kustom.

```
aws transcribe describe-language-model \
  --model-name cli-clm-example
```

Output:

```
{
  "LanguageModel": {
    "ModelName": "cli-clm-example",
    "CreateTime": "2020-09-25T17:57:38.504000+00:00",
    "LastModifiedTime": "2020-09-25T17:57:48.585000+00:00",
    "LanguageCode": "language-code",
    "BaseModelName": "base-model-name",
    "ModelStatus": "IN_PROGRESS",
    "UpgradeAvailability": false,
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",
      "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",
      "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-with-permissions-to-create-a-custom-language-model"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Akurasi Transkripsi Khusus Domain dengan Model Bahasa Khusus](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [DescribeLanguageModel](#) di Referensi AWS CLI Perintah.

## get-medical-transcription-job

Contoh kode berikut menunjukkan cara menggunakan `get-medical-transcription-job`.

### AWS CLI

Untuk mendapatkan informasi tentang pekerjaan transkripsi medis tertentu

`get-medical-transcription-job` Contoh berikut mendapatkan informasi tentang pekerjaan transkripsi medis tertentu. Untuk mengakses hasil transkripsi, gunakan `TranscriptFileUri` parameter. Jika Anda telah mengaktifkan fitur tambahan untuk pekerjaan transkripsi, Anda dapat melihatnya di objek Pengaturan. Parameter khusus menunjukkan spesialisasi medis penyedia. Parameter Tipe menunjukkan apakah ucapan dalam pekerjaan transkripsi adalah percakapan medis, atau dikte medis.

```
aws transcribe get-medical-transcription-job \  
  --medical-transcription-job-name vocabulary-dictation-medical-transcription-job
```

### Output:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-  
job",  
    "TranscriptionJobStatus": "COMPLETED",  
    "LanguageCode": "en-US",  
    "MediaSampleRateHertz": 48000,  
    "MediaFormat": "mp4",  
    "Media": {  
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-audio-file.file-extension"  
    },  
    "Transcript": {  
      "TranscriptFileUri": "https://s3.Region.amazonaws.com/Amazon-S3-Prefix/  
vocabulary-dictation-medical-transcription-job.json"  
    },  
    "StartTime": "2020-09-21T21:17:27.045000+00:00",  
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",  
  }  
}
```

```
    "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
    "Settings": {
      "ChannelIdentification": false,
      "ShowAlternatives": false,
      "VocabularyName": "cli-medical-vocab-example"
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

Untuk informasi selengkapnya, lihat [Transkripsi Batch](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [GetMedicalTranscriptionJob](#) di Referensi AWS CLI Perintah.

## get-medical-vocabulary

Contoh kode berikut menunjukkan cara menggunakan `get-medical-vocabulary`.

### AWS CLI

Untuk mendapatkan informasi tentang kosakata khusus medis

`get-medical-vocabulary` Contoh berikut mendapatkan informasi tentang kosakata khusus medis. Anda dapat menggunakan `VocabularyState` parameter untuk melihat status pemrosesan kosakata. Jika sudah SIAP, Anda bisa menggunakannya dalam `StartMedicalTranscriptionJob` operasi. :

```
aws transcribe get-medical-vocabulary \
  --vocabulary-name medical-vocab-example
```

### Output:

```
{
  "VocabularyName": "medical-vocab-example",
  "LanguageCode": "en-US",
  "VocabularyState": "READY",
  "LastModifiedTime": "2020-09-19T23:59:04.349000+00:00",
  "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-
medical-custom-vocabulary"
}
```

Untuk informasi selengkapnya, lihat [Kosakata Kustom Medis di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [GetMedicalVocabulary](#) di Referensi AWS CLI Perintah.

## get-transcription-job

Contoh kode berikut menunjukkan cara menggunakan `get-transcription-job`.

### AWS CLI

Untuk mendapatkan informasi tentang pekerjaan transkripsi tertentu

`get-transcription-job` Contoh berikut mendapatkan informasi tentang pekerjaan transkripsi tertentu. Untuk mengakses hasil transkripsi, gunakan `TranscriptFileUri` parameter. Gunakan `MediaFileUri` parameter untuk melihat file audio mana yang Anda transkripsikan dengan pekerjaan ini. Anda dapat menggunakan objek Pengaturan untuk melihat fitur opsional yang telah Anda aktifkan dalam pekerjaan transkripsi.

```
aws transcribe get-transcription-job \  
  --transcription-job-name your-transcription-job
```

### Output:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "your-transcription-job",  
    "TranscriptionJobStatus": "COMPLETED",  
    "LanguageCode": "language-code",  
    "MediaSampleRateHertz": 48000,  
    "MediaFormat": "mp4",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"  
    },  
    "Transcript": {  
      "TranscriptFileUri": "https://Amazon-S3-file-location-of-transcription-  
output"  
    },  
    "StartTime": "2020-09-18T22:27:23.970000+00:00",  
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",  
    "CompletionTime": "2020-09-18T22:28:21.197000+00:00",  
    "Settings": {
```



```
        "ChannelIdentification": false,
        "ShowAlternatives": false
    },
    "IdentifyLanguage": true,
    "IdentifiedLanguageScore": 0.8672199249267578
}
}
```

Untuk informasi selengkapnya, lihat [Memulai \(Antarmuka Baris AWS Perintah\)](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [GetTranscriptionJob](#) di Referensi AWS CLI Perintah.

## get-vocabulary-filter

Contoh kode berikut menunjukkan cara menggunakan `get-vocabulary-filter`.

### AWS CLI

Untuk mendapatkan informasi tentang filter kosakata

`get-vocabulary-filter` Contoh berikut mendapatkan informasi tentang filter kosakata. Anda dapat menggunakan `DownloadUri` parameter untuk mendapatkan daftar kata yang Anda gunakan untuk membuat filter kosakata.

```
aws transcribe get-vocabulary-filter \
  --vocabulary-filter-name testFilter
```

Output:

```
{
  "VocabularyFilterName": "testFilter",
  "LanguageCode": "language-code",
  "LastModifiedTime": "2020-05-07T22:39:32.147000+00:00",
  "DownloadUri": "https://Amazon-S3-location-to-download-your-vocabulary-filter"
}
```

Untuk informasi selengkapnya, lihat [Memfilter Kata yang Tidak Diinginkan](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [GetVocabularyFilter](#) di Referensi AWS CLI Perintah.

## get-vocabulary

Contoh kode berikut menunjukkan cara menggunakan `get-vocabulary`.

### AWS CLI

Untuk mendapatkan informasi tentang kosakata khusus

`get-vocabulary` Contoh berikut mendapatkan informasi tentang kosakata kustom yang dibuat sebelumnya.

```
aws transcribe get-vocabulary \  
  --vocabulary-name cli-vocab-1
```

Output:

```
{  
  "VocabularyName": "cli-vocab-1",  
  "LanguageCode": "language-code",  
  "VocabularyState": "READY",  
  "LastModifiedTime": "2020-09-19T23:22:32.836000+00:00",  
  "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-  
  custom-vocabulary"  
}
```

Untuk informasi selengkapnya, lihat [Kosakata Khusus di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [GetVocabulary](#) di Referensi AWS CLI Perintah.

## list-language-models

Contoh kode berikut menunjukkan cara menggunakan `list-language-models`.

### AWS CLI

Untuk membuat daftar model bahasa kustom

`list-language-models` Contoh berikut mencantumkan model bahasa kustom yang terkait dengan AWS akun dan Wilayah Anda. Anda dapat menggunakan `TuningDataS3Uri` parameter `S3Uri` dan untuk menemukan awalan Amazon S3 yang telah Anda gunakan sebagai data

pelatihan, atau data penyetalan Anda. Ini `BaseModelName` memberi tahu Anda apakah Anda telah menggunakan `NarrowBand`, atau `WideBand` model untuk membuat model bahasa khusus. Anda dapat mentranskripsikan audio dengan laju sampel kurang dari 16 kHz dengan model bahasa khusus menggunakan model dasar `NarrowBand`. Anda dapat mentranskripsikan audio 16 kHz atau lebih besar dengan model bahasa khusus menggunakan model dasar `WideBand`. `ModelStatusParameter` menunjukkan apakah Anda dapat menggunakan model bahasa khusus dalam pekerjaan transkripsi. Jika nilainya `SELESAI`, Anda dapat menggunakannya dalam pekerjaan transkripsi.

```
aws transcribe list-language-models
```

Output:

```
{
  "Models": [
    {
      "ModelName": "cli-clm-2",
      "CreateTime": "2020-09-25T17:57:38.504000+00:00",
      "LastModifiedTime": "2020-09-25T17:57:48.585000+00:00",
      "LanguageCode": "language-code",
      "BaseModelName": "WideBand",
      "ModelStatus": "IN_PROGRESS",
      "UpgradeAvailability": false,
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-training-data/",
        "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-tuning-data/",
        "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-used-to-create-the-custom-language-model"
      }
    },
    {
      "ModelName": "cli-clm-1",
      "CreateTime": "2020-09-25T17:16:01.835000+00:00",
      "LastModifiedTime": "2020-09-25T17:16:15.555000+00:00",
      "LanguageCode": "language-code",
      "BaseModelName": "WideBand",
      "ModelStatus": "IN_PROGRESS",
      "UpgradeAvailability": false,
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-training-data/",
        "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-used-to-create-the-custom-language-model"
      }
    }
  ]
}
```

```

    }
  },
  {
    "ModelName": "clm-console-1",
    "CreateTime": "2020-09-24T19:26:28.076000+00:00",
    "LastModifiedTime": "2020-09-25T04:25:22.271000+00:00",
    "LanguageCode": "language-code",
    "BaseModelName": "NarrowBand",
    "ModelStatus": "COMPLETED",
    "UpgradeAvailability": false,
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-training-data/",
      "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-used-to-create-the-custom-language-model"
    }
  }
]
}

```

Untuk informasi selengkapnya, lihat [Meningkatkan Akurasi Transkripsi Khusus Domain dengan Model Bahasa Khusus](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [ListLanguageModels](#) di Referensi AWS CLI Perintah.

## list-medical-transcription-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-medical-transcription-jobs`.

### AWS CLI

Untuk membuat daftar pekerjaan transkripsi medis Anda

`list-medical-transcription-jobs` Contoh berikut mencantumkan pekerjaan transkripsi medis yang terkait dengan AWS akun dan Wilayah Anda. Untuk mendapatkan informasi lebih lanjut tentang pekerjaan transkripsi tertentu, salin nilai `MedicalTranscriptionJobName` parameter dalam output transkripsi, dan tentukan nilai itu untuk `MedicalTranscriptionJobName` opsi perintah. `get-medical-transcription-job` Untuk melihat lebih banyak pekerjaan transkripsi Anda, salin nilai `NextToken` parameter, jalankan `list-medical-transcription-jobs` perintah lagi, dan tentukan nilai itu di `--next-token` opsi.

```
aws transcribe list-medical-transcription-jobs
```

## Output:

```

{
  "NextToken": "3/PblzkiGhzjER3KHuQt2fmbPLF7cDYafjFMEoGn440N/
gsuUSTIkGyanvRE6WMXfd/ZTEc2EZj+P9eii/
z102FDY1i6RLI0WoRX4RwMisVrh9G0Kie0Y8ikBCdtqLZB10Wa9McC+eb01
+LaDtZPC4u6ttoHLRlEfzqstHXSgapXg3tEBtm9piIaPB6M0M5BB6t86+qtmocTR/
qrteHZBBudhTfbCwhsxaqujHiiUvFdm3BQbKKWIW06yV9b+4f38oD2lVIan
+vfUs3gBYA15VTDmXXzQPBQ0HPjtwmFI+IWX15nSUjWuN3TUy1HgPWzDaYT8qBtu0Z+3UG4V6b
+K2CC0XszXg5rBq9hYgNzy4XoFh/6s5DoSznzq49Q9xHgHdT2yBADFmvFK7myZBsJ75+2vQZ0SVpWUPy3WT/32zFAcoEL
+mFYfUjtTZ8n/jq7aQEjQ42A
+X/7K6Jg0cdVPtEg8P1Dr5kgYYG3q30mYXX37U3FZuJmnTI63VtIXsNn0U5eGoY0btpk00Nq9UkzgjSxqj84ZD5n
+S0EGy9ZUYBJRRcGeYUM3Q4DbSJfUwSAqcFdLIWZdp8qIREMQIBWy7BLwSdyqsQo2vRrd53hm5aWM7SVf6pPq6X/
IXR5+1eU00D8/coaTT4ES2DerbV6RkV4o0VT1d0SdVX/
MmtkNG8nYj8PqU07w7988quh1ZP6D80veJS1q73tUUR9MjnGernW2tAnvnLNhdefBcD
+sZVfYq3iBMFY7wTy1P1G6NqW9GrYDY0X3tTPW1D7phpbVSYkrh/
PdYrps5UxnsGoA1b7L/FfAXDfUoGrGUB4N3JsPYXX9D++g+6gV1qBBs/
WFF934aKqfD6UTggm/zV3GA0WiBpfvAZRvEb924i6yGHYMC7y5401ZAwSBupmI
+FFd13CaP04kN1vJlth6Am5vUPXg4BpyUhtbRhwd/KxCvf9K0tLJGyL1A==",
  "MedicalTranscriptionJobSummaries": [
    {
      "MedicalTranscriptionJobName": "vocabulary-dictation-medical-
transcription-job",
      "CreationTime": "2020-09-21T21:17:27.016000+00:00",
      "StartTime": "2020-09-21T21:17:27.045000+00:00",
      "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "CUSTOMER_BUCKET",
      "Specialty": "PRIMARYCARE",
      "Type": "DICTATION"
    },
    {
      "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
      "CreationTime": "2020-09-21T21:01:14.569000+00:00",
      "StartTime": "2020-09-21T21:01:14.592000+00:00",
      "CompletionTime": "2020-09-21T21:01:43.606000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "CUSTOMER_BUCKET",
      "Specialty": "PRIMARYCARE",
      "Type": "DICTATION"
    }
  ],
}

```

```
{
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
  "CreationTime": "2020-09-21T19:09:18.171000+00:00",
  "StartTime": "2020-09-21T19:09:18.199000+00:00",
  "CompletionTime": "2020-09-21T19:10:22.516000+00:00",
  "LanguageCode": "en-US",
  "TranscriptionJobStatus": "COMPLETED",
  "OutputLocationType": "CUSTOMER_BUCKET",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
},
{
  "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
  "CreationTime": "2020-09-21T18:43:37.157000+00:00",
  "StartTime": "2020-09-21T18:43:37.265000+00:00",
  "CompletionTime": "2020-09-21T18:44:21.192000+00:00",
  "LanguageCode": "en-US",
  "TranscriptionJobStatus": "COMPLETED",
  "OutputLocationType": "CUSTOMER_BUCKET",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
},
{
  "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
  "CreationTime": "2020-09-20T23:46:44.053000+00:00",
  "StartTime": "2020-09-20T23:46:44.081000+00:00",
  "CompletionTime": "2020-09-20T23:47:35.851000+00:00",
  "LanguageCode": "en-US",
  "TranscriptionJobStatus": "COMPLETED",
  "OutputLocationType": "CUSTOMER_BUCKET",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
}
]
```

Untuk informasi selengkapnya, lihat <https://docs.aws.amazon.com/transcribe/latest/dg/batch-med-transcription.html> di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [ListMedicalTranscriptionJobs](#) di Referensi AWS CLI Perintah.

## list-medical-vocabularies

Contoh kode berikut menunjukkan cara menggunakan `list-medical-vocabularies`.

### AWS CLI

Untuk membuat daftar kosakata khusus medis Anda

`list-medical-vocabularies` Contoh berikut mencantumkan kosakata khusus medis yang terkait dengan AWS akun dan Wilayah Anda. Untuk mendapatkan informasi lebih lanjut tentang pekerjaan transkripsi tertentu, salin nilai `MedicalTranscriptionJobName` parameter dalam output transkripsi, dan tentukan nilai itu untuk `MedicalTranscriptionJobName` opsi perintah. `get-medical-transcription-job` Untuk melihat lebih banyak pekerjaan transkripsi Anda, salin nilai `NextToken` parameter, jalankan `list-medical-transcription-jobs` perintah lagi, dan tentukan nilai itu di `--next-token` opsi.

```
aws transcribe list-medical-vocabularies
```

Output:

```
{
  "Vocabularies": [
    {
      "VocabularyName": "cli-medical-vocab-2",
      "LanguageCode": "en-US",
      "LastModifiedTime": "2020-09-21T21:44:59.521000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "cli-medical-vocab-1",
      "LanguageCode": "en-US",
      "LastModifiedTime": "2020-09-19T23:59:04.349000+00:00",
      "VocabularyState": "READY"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kosakata Kustom Medis di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [ListMedicalVocabularies](#) di Referensi AWS CLI Perintah.

## list-transcription-jobs

Contoh kode berikut menunjukkan cara menggunakan `list-transcription-jobs`.

### AWS CLI

Untuk membuat daftar pekerjaan transkripsi Anda

`list-transcription-jobs` Contoh berikut mencantumkan pekerjaan transkripsi yang terkait dengan AWS akun dan Wilayah Anda.

```
aws transcribe list-transcription-jobs
```

Output:

```
{
  "NextToken": "NextToken",
  "TranscriptionJobSummaries": [
    {
      "TranscriptionJobName": "speak-id-job-1",
      "CreationTime": "2020-08-17T21:06:15.391000+00:00",
      "StartTime": "2020-08-17T21:06:15.416000+00:00",
      "CompletionTime": "2020-08-17T21:07:05.098000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "SERVICE_BUCKET"
    },
    {
      "TranscriptionJobName": "job-1",
      "CreationTime": "2020-08-17T20:50:24.207000+00:00",
      "StartTime": "2020-08-17T20:50:24.230000+00:00",
      "CompletionTime": "2020-08-17T20:52:18.737000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "SERVICE_BUCKET"
    },
    {
      "TranscriptionJobName": "sdk-test-job-4",
      "CreationTime": "2020-08-17T20:32:27.917000+00:00",
      "StartTime": "2020-08-17T20:32:27.956000+00:00",
      "CompletionTime": "2020-08-17T20:33:15.126000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",

```



```

        "OutputLocationType": "SERVICE_BUCKET"
    },
    {
        "TranscriptionJobName": "Diarization-speak-id",
        "CreationTime": "2020-08-10T22:10:09.066000+00:00",
        "StartTime": "2020-08-10T22:10:09.116000+00:00",
        "CompletionTime": "2020-08-10T22:26:48.172000+00:00",
        "LanguageCode": "language-code",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "SERVICE_BUCKET"
    },
    {
        "TranscriptionJobName": "your-transcription-job-name",
        "CreationTime": "2020-07-29T17:45:09.791000+00:00",
        "StartTime": "2020-07-29T17:45:09.826000+00:00",
        "CompletionTime": "2020-07-29T17:46:20.831000+00:00",
        "LanguageCode": "language-code",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "SERVICE_BUCKET"
    }
]
}

```

Untuk informasi selengkapnya, lihat [Memulai \(Antarmuka Baris AWS Perintah\)](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [ListTranscriptionJobs](#) di Referensi AWS CLI Perintah.

## list-vocabularies

Contoh kode berikut menunjukkan cara menggunakan `list-vocabularies`.

### AWS CLI

Untuk membuat daftar kosakata kustom Anda

`list-vocabularies` Contoh berikut mencantumkan kosakata kustom yang terkait dengan AWS akun dan Wilayah Anda.

```
aws transcribe list-vocabularies
```

Output:

```
{
  "NextToken": "NextToken",
  "Vocabularies": [
    {
      "VocabularyName": "ards-test-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-27T22:00:27.330000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "sample-test",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T23:04:11.044000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-test-3-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T22:12:22.277000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-test-2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T21:53:50.455000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-1-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T21:39:33.356000+00:00",
      "VocabularyState": "READY"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Kosakata Khusus di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [ListVocabularies](#) di Referensi AWS CLI Perintah.

## list-vocabulary-filters

Contoh kode berikut menunjukkan cara menggunakan `list-vocabulary-filters`.

### AWS CLI

Untuk membuat daftar filter kosakata Anda

`list-vocabulary-filters` Contoh berikut mencantumkan filter kosakata yang terkait dengan AWS akun dan Wilayah Anda.

```
aws transcribe list-vocabulary-filters
```

Output:

```
{
  "NextToken": "NextToken": [
    {
      "VocabularyFilterName": "testFilter",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-07T22:39:32.147000+00:00"
    },
    {
      "VocabularyFilterName": "testFilter2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-21T23:29:35.174000+00:00"
    },
    {
      "VocabularyFilterName": "filter2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-08T20:18:26.426000+00:00"
    },
    {
      "VocabularyFilterName": "filter-review",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-06-03T18:52:30.448000+00:00"
    },
    {
      "VocabularyFilterName": "crlf-filt",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-22T19:42:42.737000+00:00"
    }
  ]
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Memfilter Kata yang Tidak Diinginkan](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [ListVocabularyFilters](#) di Referensi AWS CLI Perintah.

## start-medical-transcription-job

Contoh kode berikut menunjukkan cara menggunakan `start-medical-transcription-job`.

### AWS CLI

Contoh 1: Untuk mentranskripsikan dikte medis yang disimpan sebagai file audio

`start-medical-transcription-job` Contoh berikut mentranskripsikan file audio. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfile.json
```

Isi dari `myfile.json`:

```
{  
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "DICTATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  }  
}
```

Output:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",
```

```

    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T00:35:22.256000+00:00",
    "CreationTime": "2020-09-20T00:35:22.218000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}

```

Untuk informasi selengkapnya, lihat [Ikhtisar Transkripsi Batch](#) di Panduan Pengembang Amazon Transcribe.

Contoh 2: Untuk mentranskripsikan dialog klinis-pasien yang disimpan sebagai file audio

`start-medical-transcription-job` Contoh berikut mentranskripsikan file audio yang berisi dialog klinis-pasien. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://mysecondfile.json

```

Isi dari `mysecondfile.json`:

```

{
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  }
}

```

Output:

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "simple-conversation-medical-transcription-
job",

```

```

    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T23:19:49.965000+00:00",
    "CreationTime": "2020-09-20T23:19:49.941000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}

```

Untuk informasi selengkapnya, lihat [Ikhtisar Transkripsi Batch](#) di Panduan Pengembang Amazon Transcribe.

Contoh 3: Untuk mentranskripsikan file audio multisaluran dari dialog dokter-pasien

`start-medical-transcription-job` Contoh berikut mentranskripsi audio dari setiap saluran dalam file audio dan menggabungkan transkripsi terpisah dari setiap saluran menjadi satu output transkripsi. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://mythirdfile.json

```

Isi dari `mythirdfile.json`:

```

{
  "MedicalTranscriptionJobName": "multichannel-conversation-medical-transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "ChannelIdentification": true
  }
}

```

Output:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T23:46:44.081000+00:00",
    "CreationTime": "2020-09-20T23:46:44.053000+00:00",
    "Settings": {
      "ChannelIdentification": true
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}
```

Untuk informasi selengkapnya, lihat [Identifikasi Saluran](#) di Panduan Pengembang Amazon Transcribe.

Contoh 4: Untuk mentranskripsikan file audio dari dialog klinis-pasien dan mengidentifikasi pembicara dalam output transkripsi

`start-medical-transcription-job` Contoh berikut mentranskripsikan file audio dan memberi label ucapan setiap pembicara dalam output transkripsi. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.

```
aws transcribe start-medical-transcription-job \
  --cli-input-json file://myfourthfile.json
```

Isi dari `myfourthfile.json`:

```
{
  "MedicalTranscriptionJobName": "speaker-id-conversation-medical-transcription-
job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
```

```

    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
  }
}

```

### Output:

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T18:43:37.265000+00:00",
    "CreationTime": "2020-09-21T18:43:37.157000+00:00",
    "Settings": {
      "ShowSpeakerLabels": true,
      "MaxSpeakerLabels": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengidentifikasi Pembicara](#) di Panduan Pengembang Amazon Transcribe.

Contoh 5: Untuk mentranskripsikan percakapan medis yang disimpan sebagai file audio dengan hingga dua alternatif transkripsi

`start-medical-transcription-job` Contoh berikut membuat hingga dua transkripsi alternatif dari satu file audio. Setiap transkripsi memiliki tingkat kepercayaan yang terkait dengannya. Secara default, Amazon Transcribe mengembalikan transkripsi dengan tingkat kepercayaan tertinggi. Anda dapat menentukan bahwa Amazon Transcribe mengembalikan transkripsi tambahan dengan tingkat kepercayaan yang lebih rendah. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.



```
aws transcribe start-medical-transcription-job \  
--cli-input-json file://myfifthfile.json
```

Isi dari myfifthfile.json:

```
{  
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-transcription-  
job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Output:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
    "StartTime": "2020-09-21T19:09:18.199000+00:00",  
    "CreationTime": "2020-09-21T19:09:18.171000+00:00",  
    "Settings": {  
      "ShowAlternatives": true,  
      "MaxAlternatives": 2  
    },  
    "Specialty": "PRIMARYCARE",  
    "Type": "CONVERSATION"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Transkripsi Alternatif](#) di Panduan Pengembang Amazon Transcribe.

Contoh 6: Untuk mentranskripsikan file audio dikte medis dengan hingga dua transkripsi alternatif `start-medical-transcription-job` Contoh berikut mentranskripsikan file audio dan menggunakan filter kosakata untuk menutupi kata-kata yang tidak diinginkan. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://mysixthfile.json
```

Isi dari `mysixthfile.json`:

```
{  
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-transcription-  
job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "DICTATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Output:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "alternatives-dictation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
    "StartTime": "2020-09-21T21:01:14.592000+00:00",  
  }  
}
```

```

    "CreationTime": "2020-09-21T21:01:14.569000+00:00",
    "Settings": {
      "ShowAlternatives": true,
      "MaxAlternatives": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}

```

Untuk informasi selengkapnya, lihat [Transkripsi Alternatif](#) di Panduan Pengembang Amazon Transcribe.

Contoh 7: Untuk mentranskripsikan file audio dikte medis dengan akurasi yang meningkat dengan menggunakan kosakata khusus

start-medical-transcription-job Contoh berikut mentranskripsikan file audio dan menggunakan kosakata khusus medis yang telah Anda buat sebelumnya untuk meningkatkan akurasi transkripsi. Anda menentukan lokasi output transkripsi dalam `OutputBucketName` parameter.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myseventhfile.json

```

Isi dari `mysixthfile.json`:

```

{
  "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "DICTATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "VocabularyName": "cli-medical-vocab-1"
  }
}

```

Output:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-
job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T21:17:27.045000+00:00",
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",
    "Settings": {
      "VocabularyName": "cli-medical-vocab-1"
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

Untuk informasi selengkapnya, lihat [Kosakata Kustom Medis di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [StartMedicalTranscriptionJob](#) di Referensi AWS CLI Perintah.

## start-transcription-job

Contoh kode berikut menunjukkan cara menggunakan `start-transcription-job`.

### AWS CLI

Contoh 1: Untuk mentranskripsikan file audio

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myfile.json
```

Isi dari `myfile.json`:

```
{
  "TranscriptionJobName": "cli-simple-transcription-job",
  "LanguageCode": "the-language-of-your-transcription-job",
```

```
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
}
```

Untuk informasi selengkapnya, lihat [Memulai \(Antarmuka Baris AWS Perintah\)](#) di Panduan Pengembang Amazon Transcribe.

Contoh 2: Untuk mentranskripsikan file audio multi-saluran

`start-transcription-job` Contoh berikut mentranskripsikan file audio multi-saluran Anda.

```
aws transcribe start-transcription-job \
  --cli-input-json file://mysecondfile.json
```

Isi dari `mysecondfile.json`:

```
{
  "TranscriptionJobName": "cli-channelid-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings":{
    "ChannelIdentification":true
  }
}
```

Output:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-channelid-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-17T16:07:56.817000+00:00",
```

```
    "CreationTime": "2020-09-17T16:07:56.784000+00:00",
    "Settings": {
      "ChannelIdentification": true
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Mentranskripsikan Audio Multi-Saluran](#) di Panduan Pengembang Amazon Transcribe.

Contoh 3: Untuk mentranskripsikan file audio dan mengidentifikasi speaker yang berbeda

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dan mengidentifikasi speaker dalam output transkripsi.

```
aws transcribe start-transcription-job \
  --cli-input-json file://mythirdfile.json
```

Isi dari `mythirdfile.json`:

```
{
  "TranscriptionJobName": "cli-speakerid-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings": {
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
  }
}
```

Output:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-speakerid-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
```

```

        "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-17T16:22:59.696000+00:00",
    "CreationTime": "2020-09-17T16:22:59.676000+00:00",
    "Settings": {
        "ShowSpeakerLabels": true,
        "MaxSpeakerLabels": 2
    }
}
}

```

Untuk informasi selengkapnya, lihat [Mengidentifikasi Pembicara](#) di Panduan Pengembang Amazon Transcribe.

Contoh 4: Untuk mentranskripsikan file audio dan menutupi kata-kata yang tidak diinginkan dalam output transkripsi

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dan menggunakan filter kosakata yang sebelumnya Anda buat untuk menutupi kata-kata yang tidak diinginkan.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myfourthfile.json

```

Isi dari `myfourthfile.json`:

```

{
  "TranscriptionJobName": "cli-filter-mask-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings":{
    "VocabularyFilterName": "your-vocabulary-filter",
    "VocabularyFilterMethod": "mask"
  }
}

```

Output:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-filter-mask-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyFilterName": "your-vocabulary-filter",
      "VocabularyFilterMethod": "mask"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Memfilter Transkripsi di Panduan Pengembang Amazon Transcribe](#).

Contoh 5: Untuk mentranskripsikan file audio dan menghapus kata-kata yang tidak diinginkan dalam output transkripsi

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dan menggunakan filter kosakata yang sebelumnya Anda buat untuk menutupi kata-kata yang tidak diinginkan.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myfifthfile.json
```

Isi dari `myfifthfile.json`:

```
{
  "TranscriptionJobName": "cli-filter-remove-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-name.file-extension"
  },
  "Settings": {
    "VocabularyFilterName": "your-vocabulary-filter",
    "VocabularyFilterMethod": "remove"
  }
}
```



```
}  
}
```

### Output:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-filter-remove-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "the-language-of-your-transcription-job",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
    },  
    "StartTime": "2020-09-18T16:36:18.568000+00:00",  
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",  
    "Settings": {  
      "VocabularyFilterName": "your-vocabulary-filter",  
      "VocabularyFilterMethod": "remove"  
    }  
  }  
}
```

Untuk informasi selengkapnya, lihat [Memfilter Transkripsi di Panduan Pengembang Amazon Transcribe](#).

Contoh 6: Untuk mentranskripsikan file audio dengan akurasi yang meningkat menggunakan kosakata khusus

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dan menggunakan filter kosakata yang sebelumnya Anda buat untuk menutupi kata-kata yang tidak diinginkan.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://mysixthfile.json
```

Isi dari `mysixthfile.json`:

```
{  
  "TranscriptionJobName": "cli-vocab-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {
```

```

    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings":{
    "VocabularyName": "your-vocabulary"
  }
}

```

### Output:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-vocab-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyName": "your-vocabulary"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Memfilter Transkripsi di Panduan](#) Pengembang Amazon Transcribe.

Contoh 7: Untuk mengidentifikasi bahasa file audio dan menuliskannya

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dan menggunakan filter kosakata yang sebelumnya Anda buat untuk menutupi kata-kata yang tidak diinginkan.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myseventhfile.json

```

Isi dari `myseventhfile.json`:

```

{

```

```
"TranscriptionJobName": "cli-identify-language-transcription-job",
"IdentifyLanguage": true,
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
}
}
```

### Output:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-identify-language-transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-18T22:27:23.970000+00:00",
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",
    "IdentifyLanguage": true
  }
}
```

Untuk informasi selengkapnya, lihat [Mengidentifikasi Bahasa](#) di Panduan Pengembang Amazon Transcribe.

Contoh 8: Untuk mentranskripsikan file audio dengan informasi yang dapat diidentifikasi secara pribadi disunting

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dan menyunting informasi identitas pribadi apa pun dalam keluaran transkripsi.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myeighthfile.json
```

Isi dari `myeighthfile.json`:

```
{
  "TranscriptionJobName": "cli-redaction-job",
  "LanguageCode": "language-code",
  "Media": {
```

```

    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
  },
  "ContentRedaction": {
    "RedactionOutput": "redacted",
    "RedactionType": "PII"
  }
}

```

### Output:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-redaction-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
    },
    "StartTime": "2020-09-25T23:49:13.195000+00:00",
    "CreationTime": "2020-09-25T23:49:13.176000+00:00",
    "ContentRedaction": {
      "RedactionType": "PII",
      "RedactionOutput": "redacted"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Redaksi Konten Otomatis](#) di Panduan Pengembang Amazon Transcribe.

Contoh 9: Untuk menghasilkan transkrip dengan informasi identitas pribadi (PII) yang disunting dan transkrip yang tidak disunting

`start-transcription-job` Contoh berikut menghasilkan dua transkripsi file audio Anda, satu dengan informasi yang dapat diidentifikasi secara pribadi disunting, dan yang lainnya tanpa redaksi apa pun.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myninthfile.json

```

Isi dari `myninthfile.json`:

```
{
  "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
  },
  "ContentRedaction": {
    "RedactionOutput": "redacted_and_unredacted",
    "RedactionType": "PII"
  }
}
```

Output:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
    },
    "StartTime": "2020-09-25T23:59:47.677000+00:00",
    "CreationTime": "2020-09-25T23:59:47.653000+00:00",
    "ContentRedaction": {
      "RedactionType": "PII",
      "RedactionOutput": "redacted_and_unredacted"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Redaksi Konten Otomatis](#) di Panduan Pengembang Amazon Transcribe.

Contoh 10: Untuk menggunakan model bahasa kustom yang sebelumnya Anda buat untuk mentranskripsikan file audio.

`start-transcription-job` Contoh berikut mentranskripsikan file audio Anda dengan model bahasa khusus yang telah Anda buat sebelumnya.

```
aws transcribe start-transcription-job \
```

```
--cli-input-json file://mytenthfile.json
```

Isi dari mytenthfile.json:

```
{
  "TranscriptionJobName": "cli-clm-2-job-1",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"
  },
  "ModelSettings": {
    "LanguageModelName": "cli-clm-2"
  }
}
```

Output:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-clm-2-job-1",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"
    },
    "StartTime": "2020-09-28T17:56:01.835000+00:00",
    "CreationTime": "2020-09-28T17:56:01.801000+00:00",
    "ModelSettings": {
      "LanguageModelName": "cli-clm-2"
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Meningkatkan Akurasi Transkripsi Khusus Domain dengan Model Bahasa Khusus](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [StartTranscriptionJob](#) di Referensi AWS CLI Perintah.

## update-medical-vocabulary

Contoh kode berikut menunjukkan cara menggunakan `update-medical-vocabulary`.

## AWS CLI

Untuk memperbarui kosakata khusus medis dengan istilah baru.

`update-medical-vocabulary` Contoh berikut menggantikan istilah yang digunakan dalam kosakata khusus medis dengan yang baru. Prasyarat: untuk mengganti istilah dalam kosakata khusus medis, Anda memerlukan file dengan istilah baru.

```
aws transcribe update-medical-vocabulary \  
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/medical-custom-  
vocabulary.txt \  
  --vocabulary-name medical-custom-vocabulary \  
  --language-code language
```

Output:

```
{  
  "VocabularyName": "medical-custom-vocabulary",  
  "LanguageCode": "en-US",  
  "VocabularyState": "PENDING"  
}
```

Untuk informasi selengkapnya, lihat [Kosakata Kustom Medis di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [UpdateMedicalVocabulary](#) di Referensi AWS CLI Perintah.

## update-vocabulary-filter

Contoh kode berikut menunjukkan cara menggunakan `update-vocabulary-filter`.

### AWS CLI

Untuk mengganti kata-kata dalam filter kosakata

`update-vocabulary-filter` Contoh berikut menggantikan kata-kata dalam filter kosakata dengan yang baru. Prasyarat: Untuk memperbarui filter kosakata dengan kata-kata baru, Anda harus menyimpan kata-kata itu sebagai file teks.

```
aws transcribe update-vocabulary-filter \  
  --vocabulary-name medical-custom-vocabulary \  
  --filter-name medical-custom-filter \  
  --filter-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/medical-custom-  
filter.txt
```

```
--vocabulary-filter-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/your-text-  
file-to-update-your-vocabulary-filter.txt \  
--vocabulary-filter-name vocabulary-filter-name
```

### Output:

```
{  
  "VocabularyFilterName": "vocabulary-filter-name",  
  "LanguageCode": "language-code",  
  "LastModifiedTime": "2020-09-23T18:40:35.139000+00:00"  
}
```

Untuk informasi selengkapnya, lihat [Memfilter Kata yang Tidak Diinginkan](#) di Panduan Pengembang Amazon Transcribe.

- Untuk detail API, lihat [UpdateVocabularyFilter](#) di Referensi AWS CLI Perintah.

## update-vocabulary

Contoh kode berikut menunjukkan cara menggunakan `update-vocabulary`.

### AWS CLI

Untuk memperbarui kosakata khusus dengan istilah baru.

`update-vocabulary` Contoh berikut menimpa istilah yang digunakan untuk membuat kosakata khusus dengan yang baru yang Anda berikan. Prasyarat: untuk mengganti istilah dalam kosakata khusus, Anda memerlukan file dengan istilah baru.

```
aws transcribe update-vocabulary \  
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/custom-  
vocabulary.txt \  
  --vocabulary-name custom-vocabulary \  
  --language-code language-code
```

### Output:

```
{  
  "VocabularyName": "custom-vocabulary",  
  "LanguageCode": "language",  
  "VocabularyState": "PENDING"
```



```
}
```

Untuk informasi selengkapnya, lihat [Kosakata Khusus di Panduan](#) Pengembang Amazon Transcribe.

- Untuk detail API, lihat [UpdateVocabulary](#) di Referensi AWS CLI Perintah.

## Contoh Amazon Translate menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Amazon Translate.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **import-terminology**

Contoh kode berikut menunjukkan cara menggunakan `import-terminology`.

AWS CLI

Untuk mengimpor terminologi kustom dari file

`import-terminology` Contoh berikut membuat terminologi yang disebut `MyTestTerminology` dari `test-terminology.csv` file:

```
aws translate import-terminology \  
  --name MyTestTerminology \  
  --source test-terminology.csv
```

```
--description "Creating a test terminology in AWS Translate" \  
--merge-strategy OVERWRITE \  
--data-file fileb://test-terminology.csv \  
--terminology-data Format=CSV
```

Isi dari test-terminology.csv:

en, fr, es, zh Halo dunia! , Bonjour memuji-muji le monde! , Hola Mundo! ,???? Amazon, Amazon, Amazon, Amazon

Output:

```
{  
  "TerminologyProperties": {  
    "SourceLanguageCode": "en",  
    "Name": "MyTestTerminology",  
    "TargetLanguageCodes": [  
      "fr",  
      "es",  
      "zh"  
    ],  
    "SizeBytes": 97,  
    "LastUpdatedAt": 1571089500.851,  
    "CreatedAt": 1571089500.851,  
    "TermCount": 6,  
    "Arn": "arn:aws:translate:us-west-2:123456789012:terminology/  
MyTestTerminology/LATEST",  
    "Description": "Creating a test terminology in AWS Translate"  
  }  
}
```

- Untuk detail API, lihat [ImportTerminology](#) di Referensi AWS CLI Perintah.

## Trusted Advisor contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Trusted Advisor.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **get-organization-recommendation**

Contoh kode berikut menunjukkan cara menggunakan `get-organization-recommendation`.

AWS CLI

Untuk mendapatkan rekomendasi organisasi

`get-organization-recommendation` Contoh berikut mendapat rekomendasi organisasi oleh pengenalnya.

```
aws trustedadvisor get-organization-recommendation \
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-
  recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5
```

Output:

```
{
  "organizationRecommendation": {
    "arn": "arn:aws:trustedadvisor::organization-recommendation/9534ec9b-
    bf3a-44e8-8213-2ed68b39d9d5",
    "name": "Lambda Runtime Deprecation Warning",
    "description": "One or more lambdas are using a deprecated runtime",
    "awsServices": [
      "lambda"
    ],
    "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
    "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
    "lifecycleStage": "resolved",
```

```

    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 0,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
  }
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [GetOrganizationRecommendation](#) di Referensi AWS CLI Perintah.

## get-recommendation

Contoh kode berikut menunjukkan cara menggunakan `get-recommendation`.

### AWS CLI

Untuk mendapatkan rekomendasi

`get-recommendation` Contoh berikut mendapat rekomendasi oleh pengenalnya.

```

aws trustedadvisor get-recommendation \
  --recommendation-identifier
  arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
  bbb7-491a-833b-5773e9589578

```

Output:

```

{
  "recommendation": {
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
    bbb7-491a-833b-5773e9589578",
    "name": "MFA Recommendation",
    "description": "Enable multi-factor authentication",

```

```
    "awsServices": [
      "iam"
    ],
    "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",
    "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
    "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
    "pillarSpecificAggregates": {
      "costOptimizing": {
        "estimatedMonthlySavings": 0.0,
        "estimatedPercentMonthlySavings": 0.0
      }
    },
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 1,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "error",
    "type": "standard"
  }
}
```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [GetRecommendation](#) di Referensi AWS CLI Perintah.

## list-checks

Contoh kode berikut menunjukkan cara menggunakan `list-checks`.

### AWS CLI

Untuk daftar cek Trusted Advisor

`list-checks` Contoh berikut mencantumkan semua cek Trusted Advisor.

```
aws trustedadvisor list-checks
```

## Output:

```
{
  "checkSummaries": [
    {
      "arn": "arn:aws:trustedadvisor:::check/1iG5NDGVre",
      "awsServices": [
        "EC2"
      ],
      "description": "Checks security groups for rules that allow unrestricted
access to a resource. Unrestricted access increases opportunities for malicious
activity (hacking, denial-of-service attacks, loss of data)",
      "id": "1iG5NDGVre",
      "metadata": {
        "0": "Region",
        "1": "Security Group Name",
        "2": "Security Group ID",
        "3": "Protocol",
        "4": "Port",
        "5": "Status",
        "6": "IP Range"
      },
      "name": "Security Groups - Unrestricted Access",
      "pillars": [
        "security"
      ],
      "source": "ta_check"
    },
    {
      "arn": "arn:aws:trustedadvisor:::check/1qazXsw23e",
      "awsServices": [
        "RDS"
      ],
      "description": "Checks your usage of RDS and provides recommendations
on purchase of Reserved Instances to help reduce costs incurred from using RDS
On-Demand. AWS generates these recommendations by analyzing your On-Demand usage
for the past 30 days. We then simulate every combination of reservations in the
generated category of usage in order to identify the best number of each type
of Reserved Instance to purchase to maximize your savings. This check covers
recommendations based on partial upfront payment option with 1-year or 3-year
commitment. This check is not available to accounts linked in Consolidated Billing.
Recommendations are only available for the Paying Account.",
      "id": "1qazXsw23e",
      "metadata": {
```

```

    "0": "Region",
    "1": "Family",
    "2": "Instance Type",
    "3": "License Model",
    "4": "Database Edition",
    "5": "Database Engine",
    "6": "Deployment Option",
    "7": "Recommended number of Reserved Instances to purchase",
    "8": "Expected Average Reserved Instance Utilization",
    "9": "Estimated Savings with Recommendation (monthly)"
    "10": "Upfront Cost of Reserved Instances",
    "11": "Estimated cost of Reserved Instances (monthly)",
    "12": "Estimated On-Demand Cost Post Recommended Reserved Instance
Purchase (monthly)",
    "13": "Estimated Break Even (months)",
    "14": "Lookback Period (days)",
    "15": "Term (years)"
  },
  "name": "Amazon Relational Database Service (RDS) Reserved Instance
Optimization",
  "pillars": [
    "cost_optimizing"
  ],
  "source": "ta_check"
},
{
  "arn": "arn:aws:trustedadvisor:::check/1qw23er45t",
  "awsServices": [
    "Redshift"
  ],
  "description": "Checks your usage of Redshift and provides
recommendations on purchase of Reserved Nodes to help reduce costs incurred from
using Redshift On-Demand. AWS generates these recommendations by analyzing your
On-Demand usage for the past 30 days. We then simulate every combination of
reservations in the generated category of usage in order to identify the best
number of each type of Reserved Nodes to purchase to maximize your savings. This
check covers recommendations based on partial upfront payment option with 1-year or
3-year commitment. This check is not available to accounts linked in Consolidated
Billing. Recommendations are only available for the Paying Account.",
  "id": "1qw23er45t",
  "metadata": {
    "0": "Region",
    "1": "Family",
    "2": "Node Type",

```

```

        "3": "Recommended number of Reserved Nodes to purchase",
        "4": "Expected Average Reserved Node Utilization",
        "5": "Estimated Savings with Recommendation (monthly)",
        "6": "Upfront Cost of Reserved Nodes",
        "7": "Estimated cost of Reserved Nodes (monthly)",
        "8": "Estimated On-Demand Cost Post Recommended Reserved Nodes
Purchase (monthly)",
        "9": "Estimated Break Even (months)",
        "10": "Lookback Period (days)",
        "11": "Term (years)",
    },
    "name": "Amazon Redshift Reserved Node Optimization",
    "pillars": [
        "cost_optimizing"
    ],
    "source": "ta_check"
},
],
"nextToken": "REDACTED"
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [ListChecks](#) di Referensi AWS CLI Perintah.

## list-organization-recommendation-accounts

Contoh kode berikut menunjukkan cara menggunakan `list-organization-recommendation-accounts`.

### AWS CLI

Untuk membuat daftar akun rekomendasi organisasi

`list-organization-recommendation-accounts` Contoh berikut mencantumkan semua ringkasan rekomendasi akun untuk rekomendasi organisasi dengan pengenalnya.

```

aws trustedadvisor list-organization-recommendation-accounts \
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-
  recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5

```

Output:



```
{
  "accountRecommendationLifecycleSummaries": [{
    "accountId": "000000000000",
    "accountRecommendationArn":
"arn:aws:trustedadvisor::000000000000:recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
    "lifecycleStage": "resolved",
    "updateReason": "Resolved issue",
    "updateReasonCode": "valid_business_case",
    "lastUpdatedAt": "2023-01-17T18:25:44.552Z"
  }],
  "nextToken": "REDACTED"
}
```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [ListOrganizationRecommendationAccounts](#) di Referensi AWS CLI Perintah.

## list-organization-recommendation-resources

Contoh kode berikut menunjukkan cara menggunakan `list-organization-recommendation-resources`.

### AWS CLI

Untuk membuat daftar sumber daya rekomendasi organisasi

`list-organization-recommendation-resources` Contoh berikut mencantumkan semua sumber daya untuk rekomendasi organisasi dengan pengenalnya.

```
aws trustedadvisor list-organization-recommendation-resources \
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0
```

Output:

```
{
  "organizationRecommendationResourceSummaries": [
    {
```

```

    "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/5a694939-2e54-45a2-ae72-730598fa89d0/
bb38affc0ce0681d9a6cd13f30238ba03a8f63dfe7a379dc403c619119d86af",
    "awsResourceId": "database-1-instance-1",
    "id":
"bb38affc0ce0681d9a6cd13f302383ba03a8f63dfe7a379dc403c619119d86af",
    "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
    "metadata": {
      "0": "14",
      "1": "208.79999999999998",
      "2": "database-1-instance-1",
      "3": "db.r5.large",
      "4": "false",
      "5": "us-west-2",
      "6": "arn:aws:rds:us-west-2:000000000000:db:database-1-instance-1",
      "7": "1"
    },
    "recommendationArn": "arn:aws:trustedadvisor::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
    "regionCode": "us-west-2",
    "status": "warning"
  },
  {
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/5a694939-2e54-45a2-
ae72-730598fa89d0/51fded4d7a3278818df9cfe344ff5762cec46c095a6763d1ba1ba53bd0e1b0e6",
    "awsResourceId": "database-1",
    "id":
"51fded4d7a3278818df9cfe344ff5762cec46c095a6763d1ba1ba53bd0e1b0e6",
    "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
    "metadata": {
      "0": "14",
      "1": "31.679999999999996",
      "2": "database-1",
      "3": "db.t3.small",
      "4": "false",
      "5": "us-west-2",
      "6": "arn:aws:rds:us-west-2:000000000000:db:database-1",
      "7": "20"
    },
    "recommendationArn": "arn:aws:trustedadvisor::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
    "regionCode": "us-west-2",
    "status": "warning"
  }

```

```

    },
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/5a694939-2e54-45a2-ae72-730598fa89d0/
f4d01bd20f4cd5372062aafc8786c489e48f0ead7cdab121463bf9f89e40a36b",
      "awsResourceId": "database-2-instance-1-us-west-2a",
      "id":
"f4d01bd20f4cd5372062aafc8786c489e48f0ead7cdab121463bf9f89e40a36b",
      "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
      "metadata": {
        "0": "14",
        "1": "187.200000000000002",
        "2": "database-2-instance-1-us-west-2a",
        "3": "db.r6g.large",
        "4": "true",
        "5": "us-west-2",
        "6": "arn:aws:rds:us-west-2:000000000000:db:database-2-instance-1-
us-west-2a",
        "7": "1"
      },
      "recommendationArn": "arn:aws:trustedadvisor::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
      "regionCode": "us-west-2",
      "status": "warning"
    },
  ],
  "nextToken": "REDACTED"
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [ListOrganizationRecommendationResources](#) di Referensi AWS CLI Perintah.

## list-organization-recommendations

Contoh kode berikut menunjukkan cara menggunakan `list-organization-recommendations`.

### AWS CLI

Contoh 1: Untuk membuat daftar rekomendasi organisasi

`list-organization-recommendations` Contoh berikut mencantumkan semua rekomendasi organisasi dan tidak menyertakan filter.

```
aws trustedadvisor list-organization-recommendations
```

Output:

```
{
  "organizationRecommendationSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::organization-recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
      "name": "Lambda Runtime Deprecation Warning",
      "awsServices": [
        "lambda"
      ],
      "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
      "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
      "lifecycleStage": "resolved",
      "pillars": [
        "security"
      ],
      "resourcesAggregates": {
        "errorCount": 0,
        "okCount": 0,
        "warningCount": 0
      },
      "source": "ta_check",
      "status": "warning",
      "type": "priority"
    },
    {
      "arn": "arn:aws:trustedadvisor::organization-
recommendation/4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
      "name": "Lambda Runtime Deprecation Warning",
      "awsServices": [
        "lambda"
      ],
      "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
      "id": "4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
      "lifecycleStage": "resolved",
      "pillars": [
        "security"
      ]
    }
  ]
}
```

```

    ],
    "resourcesAggregates": {
      "errorCount": 0,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
  },
],
"nextToken": "REDACTED"
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

Contoh 2: Untuk membuat daftar rekomendasi organisasi dengan filter

`list-organization-recommendations` Contoh berikut memfilter dan mengembalikan maksimal satu rekomendasi organisasi yang merupakan bagian dari pilar “keamanan”.

```

aws trustedadvisor list-organization-recommendations \
  --pillar security \
  --max-items 100

```

Output:

```

{
  "organizationRecommendationSummaries": [{
    "arn": "arn:aws:trustedadvisor::organization-recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
    "name": "Lambda Runtime Deprecation Warning",
    "awsServices": [
      "lambda"
    ],
    "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
    "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
    "lifecycleStage": "resolved",
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {

```

```

        "errorCount": 0,
        "okCount": 0,
        "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
}],
"nextToken": "REDACTED"
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

Contoh 3: Untuk membuat daftar rekomendasi organisasi dengan token pagination

`list-organization-recommendations` Contoh berikut menggunakan “NextToken” yang dikembalikan dari permintaan sebelumnya untuk mengambil halaman rekomendasi organisasi berikutnya.

```

aws trustedadvisor list-organization-recommendations \
  --pillar security \
  --max-items 100 \
  --starting-token <next-token>

```

Output:

```

{
  "organizationRecommendationSummaries": [{
    "arn": "arn:aws:trustedadvisor::organization-recommendation/4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
    "name": "Lambda Runtime Deprecation Warning",
    "awsServices": [
      "lambda"
    ],
    "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
    "id": "4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
    "lifecycleStage": "resolved",
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 0,

```

```

        "okCount": 0,
        "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
  ]
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [ListOrganizationRecommendations](#) di Referensi AWS CLI Perintah.

## list-recommendation-resources

Contoh kode berikut menunjukkan cara menggunakan `list-recommendation-resources`.

### AWS CLI

Untuk membuat daftar sumber rekomendasi

`list-recommendation-resources` Contoh berikut mencantumkan semua sumber daya untuk rekomendasi oleh pengenalnya.

```

aws trustedadvisor list-recommendation-resources \
  --recommendation-identifier
  arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
  bbb7-491a-833b-5773e9589578

```

Output:

```

{
  "recommendationResourceSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
      resource/55fa4d2e-
      bbb7-491a-833b-5773e9589578/18959a1f1973cff8e706e9d9bde28bba36cd602a6b2cb86c8b61252835236010",
      "id":
      "18959a1f1973cff8e706e9d9bde28bba36cd602a6b2cb86c8b61252835236010",
      "awsResourceId": "webcms-dev-01",
      "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
      "metadata": {

```

```

        "0": "14",
        "1": "123.12000000000002",
        "2": "webcms-dev-01",
        "3": "db.m6i.large",
        "4": "false",
        "5": "us-east-1",
        "6": "arn:aws:rds:us-east-1:000000000000:db:webcms-dev-01",
        "7": "20"
    },
    "recommendationArn":
    "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
    "regionCode": "us-east-1",
    "status": "warning"
},
{
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/55fa4d2e-bbb7-491a-833b-5773e9589578/
e6367ff500ac90db8e4adeb4892e39ee9c36bbf812dcfce4b9e4fefcec9eb63e",
    "id":
    "e6367ff500ac90db8e4adeb4892e39ee9c36bbf812dcfce4b9e4fefcec9eb63e",
    "awsResourceId": "aws-dev-db-stack-instance-1",
    "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
    "metadata": {
        "0": "14",
        "1": "29.52",
        "2": "aws-dev-db-stack-instance-1",
        "3": "db.t2.small",
        "4": "false",
        "5": "us-east-1",
        "6": "arn:aws:rds:us-east-1:000000000000:db:aws-dev-db-stack-
instance-1",
        "7": "1"
    },
    "recommendationArn":
    "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
    "regionCode": "us-east-1",
    "status": "warning"
},
{
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/55fa4d2e-
bbb7-491a-833b-5773e9589578/31aa78ba050a5015d2d38cca7f5f1ce88f70857c4e1c3ad03f8f9fd95dad7459

```



```

      "id":
        "31aa78ba050a5015d2d38cca7f5f1ce88f70857c4e1c3ad03f8f9fd95dad7459",
        "awsResourceId": "aws-awesome-apps-stack-db",
        "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
        "metadata": {
          "0": "14",
          "1": "114.48000000000002",
          "2": "aws-awesome-apps-stack-db",
          "3": "db.m6g.large",
          "4": "false",
          "5": "us-east-1",
          "6": "arn:aws:rds:us-east-1:000000000000:db:aws-awesome-apps-stack-
db",
          "7": "100"
        },
        "recommendationArn":
        "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
        "regionCode": "us-east-1",
        "status": "warning"
      }
    ],
    "nextToken": "REDACTED"
  }

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [ListRecommendationResources](#) di Referensi AWS CLI Perintah.

## list-recommendations

Contoh kode berikut menunjukkan cara menggunakan `list-recommendations`.

### AWS CLI

Contoh 1: Untuk daftar rekomendasi

`list-recommendations` Contoh berikut mencantumkan semua rekomendasi dan tidak menyertakan filter.

```
aws trustedadvisor list-recommendations
```

## Output:

```
{
  "recommendationSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
      "name": "MFA Recommendation",
      "awsServices": [
        "iam"
      ],
      "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",
      "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
      "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
      "pillarSpecificAggregates": {
        "costOptimizing": {
          "estimatedMonthlySavings": 0.0,
          "estimatedPercentMonthlySavings": 0.0
        }
      },
      "pillars": [
        "security"
      ],
      "resourcesAggregates": {
        "errorCount": 1,
        "okCount": 0,
        "warningCount": 0
      },
      "source": "ta_check",
      "status": "error",
      "type": "standard"
    },
    {
      "arn":
"arn:aws:trustedadvisor::000000000000:recommendation/8b602b6f-452d-4cb2-8a9e-
c7650955d9cd",
      "name": "RDS clusters quota warning",
      "awsServices": [
        "rds"
      ],
      "checkArn": "arn:aws:trustedadvisor:::check/gjqMBn6pjz",
      "id": "8b602b6f-452d-4cb2-8a9e-c7650955d9cd",
      "lastUpdatedAt": "2023-11-01T15:58:17.397Z",
      "pillarSpecificAggregates": {
```

```

        "costOptimizing": {
            "estimatedMonthlySavings": 0.0,
            "estimatedPercentMonthlySavings": 0.0
        },
    ],
    "pillars": [
        "service_limits"
    ],
    "resourcesAggregates": {
        "errorCount": 0,
        "okCount": 3,
        "warningCount": 6
    },
    "source": "ta_check",
    "status": "warning",
    "type": "standard"
}
],
"nextToken": "REDACTED"
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

Contoh 2: Untuk daftar rekomendasi dengan filter

`list-recommendations` Contoh berikut mencantumkan rekomendasi dan menyertakan filter.

```

aws trustedadvisor list-recommendations \
  --aws-service iam \
  --max-items 100

```

Output:

```

{
  "recommendationSummaries": [{
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
    "name": "MFA Recommendation",
    "awsServices": [
      "iam"
    ],
    "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",

```

```

    "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
    "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
    "pillarSpecificAggregates": {
      "costOptimizing": {
        "estimatedMonthlySavings": 0.0,
        "estimatedPercentMonthlySavings": 0.0
      }
    },
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 1,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "error",
    "type": "standard"
  ]],
  "nextToken": "REDACTED"
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

Contoh 3: Untuk membuat daftar rekomendasi dengan token pagination

`list-recommendations` Contoh berikut menggunakan “NextToken” yang dikembalikan dari permintaan sebelumnya untuk mengambil halaman berikutnya dari Rekomendasi yang difilter.

```

aws trustedadvisor list-recommendations \
  --aws-service rds \
  --max-items 100 \
  --starting-token <next-token>

```

Output:

```

{
  "recommendationSummaries": [{
    "arn":
    "arn:aws:trustedadvisor::000000000000:recommendation/8b602b6f-452d-4cb2-8a9e-
    c7650955d9cd",

```

```

    "name": "RDS clusters quota warning",
    "awsServices": [
      "rds"
    ],
    "checkArn": "arn:aws:trustedadvisor:::check/gjqMBn6pjz",
    "id": "8b602b6f-452d-4cb2-8a9e-c7650955d9cd",
    "lastUpdatedAt": "2023-11-01T15:58:17.397Z",
    "pillarSpecificAggregates": {
      "costOptimizing": {
        "estimatedMonthlySavings": 0.0,
        "estimatedPercentMonthlySavings": 0.0
      }
    },
    "pillars": [
      "service_limits"
    ],
    "resourcesAggregates": {
      "errorCount": 0,
      "okCount": 3,
      "warningCount": 6
    },
    "source": "ta_check",
    "status": "warning",
    "type": "standard"
  }
}

```

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [ListRecommendations](#) di Referensi AWS CLI Perintah.

## update-organization-recommendation-lifecycle

Contoh kode berikut menunjukkan cara menggunakan `update-organization-recommendation-lifecycle`.

### AWS CLI

Untuk memperbarui siklus hidup rekomendasi organisasi

`update-organization-recommendation-lifecycle` Contoh berikut memperbarui siklus hidup rekomendasi organisasi dengan pengenalnya.

```
aws trustedadvisor update-organization-recommendation-lifecycle \  
  --organization-recommendation-identifier arn:aws:trustedadvisor:::organization-  
recommendation/96b5e5ca-7930-444c-90c6-06d386128100 \  
  --lifecycle-stage dismissed \  
  --update-reason-code not_applicable
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [UpdateOrganizationRecommendationLifecycle](#) di Referensi AWS CLI Perintah.

## update-recommendation-lifecycle

Contoh kode berikut menunjukkan cara menggunakan `update-recommendation-lifecycle`.

### AWS CLI

Untuk memperbarui siklus hidup rekomendasi

`update-recommendation-lifecycle` Contoh berikut memperbarui siklus hidup rekomendasi dengan pengenalnya.

```
aws trustedadvisor update-recommendation-lifecycle \  
  --recommendation-identifier  
arn:aws:trustedadvisor:::000000000000:recommendation/861c9c6e-  
f169-405a-8b59-537a8cacc7a \  
  --lifecycle-stage resolved \  
  --update-reason-code valid_business_case
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai API Trusted Advisor di Panduan Pengguna Trusted Advisor.AWS](#)

- Untuk detail API, lihat [UpdateRecommendationLifecycle](#) di Referensi AWS CLI Perintah.

## Contoh Izin Terverifikasi menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan menerapkan skenario umum dengan menggunakan Izin Terverifikasi AWS Command Line Interface dengan.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-identity-source**

Contoh kode berikut menunjukkan cara menggunakan `create-identity-source`.

AWS CLI

Untuk membuat sumber identitas

`create-identity-source` Contoh berikut membuat sumber identitas yang memungkinkan Anda mereferensikan identitas yang disimpan di kumpulan pengguna Amazon Cognito yang ditentukan. Identitas tersebut tersedia di Izin Terverifikasi sebagai entitas tipe. User

```
aws verifiedpermissions create-identity-source \  
  --configuration file://config.txt \  
  --principal-entity-type "User" \  
  --policy-store-id PSEXAMPLEEabcdefg111111
```

Isi dari `config.txt`:

```
{  
  "cognitoUserPoolConfiguration": {
```

```
    "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_1a2b3c4d5",
    "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"]
  }
}
```

Output:

```
{
  "createdDate": "2023-05-19T20:30:28.214829+00:00",
  "identitySourceId": "ISEXAMPLEabcdefgh111111",
  "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
  "policyStoreId": "PSEXAMPLEabcdefgh111111"
}
```

Untuk informasi selengkapnya tentang sumber identitas, lihat [Menggunakan Izin Terverifikasi Amazon dengan penyedia identitas](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [CreateIdentitySource](#) di Referensi AWS CLI Perintah.

## create-policy-store

Contoh kode berikut menunjukkan cara menggunakan `create-policy-store`.

### AWS CLI

Untuk membuat toko kebijakan

`create-policy-store` Contoh berikut membuat penyimpanan kebijakan di AWS Wilayah saat ini.

```
aws verifiedpermissions create-policy-store \
  --validation-settings "mode=STRICT"
```

Output:

```
{
  "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefgh111111",
  "createdDate": "2023-05-16T17:41:29.103459+00:00",
  "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",
  "policyStoreId": "PSEXAMPLEabcdefgh111111"
}
```



```
}
```

Untuk informasi selengkapnya tentang toko kebijakan, lihat penyimpanan [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [CreatePolicyStore](#) di Referensi AWS CLI Perintah.

## create-policy-template

Contoh kode berikut menunjukkan cara menggunakan `create-policy-template`.

### AWS CLI

Contoh 1: Untuk membuat template kebijakan

`create-policy-template` Contoh berikut membuat template kebijakan dengan pernyataan yang berisi placeholder untuk prinsipal.

```
aws verifiedpermissions create-policy-template \  
  --definition file://template1.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Isi file `template1.txt`:

```
permit(  
  principal in ?principal,  
  action == Action::"view",  
  resource == Photo::"VacationPhoto94.jpg"  
);
```

Output:

```
{  
  "createdDate": "2023-06-12T20:47:42.804511+00:00",  
  "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefg111111",  
  "policyTemplateId": "PTEXAMPLEabcdefg111111"  
}
```

Untuk informasi selengkapnya tentang templat kebijakan, lihat [templat kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [CreatePolicyTemplate](#) di Referensi AWS CLI Perintah.

## create-policy

Contoh kode berikut menunjukkan cara menggunakan `create-policy`.

### AWS CLI

Contoh 1: Untuk membuat kebijakan statis

`create-policy` Contoh berikut membuat kebijakan statis dengan cakupan kebijakan yang menentukan prinsipal dan sumber daya.

```
aws iam create-policy \
  --definition file://definition1.txt \
  --policy-store-id PSEXAMPLEEabcdefg111111
```

Isi file `definition1.txt`:

```
{
  "static": {
    "description": "Grant everyone of janeFriends UserGroup access to the
vacationFolder Album",
    "statement": "permit(principal in UserGroup::\\"janeFriends\\", action,
resource in Album::\\"vacationFolder\\" );"
  }
}
```

Output:

```
{
  "createdDate": "2023-06-12T20:33:37.382907+00:00",
  "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
  "policyId": "SPEXAMPLEEabcdefg111111",
  "policyStoreId": "PSEXAMPLEEabcdefg111111",
  "policyType": "STATIC",
  "principal": {
    "entityId": "janeFriends",
    "entityType": "UserGroup"
  },
  "resource": {
```

```

    "entityId": "vacationFolder",
    "entityType": "Album"
  }
}

```

Contoh 2: Untuk membuat kebijakan statis yang memberikan akses ke sumber daya kepada semua orang

`create-policy` Contoh berikut membuat kebijakan statis dengan cakupan kebijakan yang hanya menentukan sumber daya.

```

aws verifiedpermissions create-policy \
  --definition file://definition2.txt \
  --policy-store-id PSEXAMPLEEabcdefg111111

```

Isi file `definition2.txt`:

```

{
  "static": {
    "description": "Grant everyone access to the publicFolder Album",
    "statement": "permit(principal, action, resource in Album:\""publicFolder
\"");"
  }
}

```

Output:

```

{
  "createdDate": "2023-06-12T20:39:44.975897+00:00",
  "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
  "policyId": "PbfR73F8oh5MMfr9uRtFDB",
  "policyStoreId": "PSEXAMPLEEabcdefg222222",
  "policyType": "STATIC",
  "resource": {
    "entityId": "publicFolder",
    "entityType": "Album"
  }
}

```

Contoh 3: Untuk membuat kebijakan terkait template yang dikaitkan dengan template yang ditentukan

`create-policy` Contoh berikut membuat kebijakan terkait templat menggunakan templat kebijakan yang ditentukan dan mengaitkan prinsipal yang ditentukan untuk digunakan dengan kebijakan terkait templat yang baru.

```
aws verifiedpermissions create-policy \  
  --definition file://definition.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Isi dari `definition.txt`:

```
{  
  "templateLinked": {  
    "policyTemplateId": "PTEXAMPLEabcdefg111111",  
    "principal": {  
      "entityType": "User",  
      "entityId": "alice"  
    }  
  }  
}
```

Output:

```
{  
  "createdDate": "2023-06-12T20:49:51.490211+00:00",  
  "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",  
  "policyId": "TPEXAMPLEabcdefg111111",  
  "policyStoreId": "PSEXAMPLEabcdefg111111",  
  "policyType": "TEMPLATE_LINKED",  
  "principal": {  
    "entityId": "alice",  
    "entityType": "User"  
  },  
  "resource": {  
    "entityId": "VacationPhoto94.jpg",  
    "entityType": "Photo"  
  }  
}
```

Untuk informasi selengkapnya tentang kebijakan, lihat [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [CreatePolicy](#) di Referensi AWS CLI Perintah.

## delete-identity-source

Contoh kode berikut menunjukkan cara menggunakan `delete-identity-source`.

### AWS CLI

Untuk menghapus sumber identitas

`delete-identity-source` Contoh berikut menghapus sumber identitas yang memiliki Id tertentu.

```
aws verifiedpermissions delete-identity-source \  
  --identity-source-id IEXAMPLEabcdefg111111 \  
  --policy-store-id PEXAMPLEabcdefg111111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang sumber identitas, lihat [Menggunakan Izin Terverifikasi Amazon dengan penyedia identitas](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [DeleteIdentitySource](#) di Referensi AWS CLI Perintah.

## delete-policy-store

Contoh kode berikut menunjukkan cara menggunakan `delete-policy-store`.

### AWS CLI

Untuk menghapus toko kebijakan

`delete-policy-store` Contoh berikut menghapus penyimpanan kebijakan yang memiliki Id yang ditentukan.

```
aws verifiedpermissions delete-policy-store \  
  --policy-store-id PEXAMPLEabcdefg111111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang toko kebijakan, lihat penyimpanan [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [DeletePolicyStore](#) di Referensi AWS CLI Perintah.

## delete-policy-template

Contoh kode berikut menunjukkan cara menggunakan `delete-policy-template`.

### AWS CLI

Untuk menghapus templat kebijakan

`delete-policy-template` Contoh berikut menghapus template kebijakan yang memiliki Id tertentu.

```
aws verifiedpermissions delete-policy \  
  --policy-template-id PTEXTAMPEabcdefg111111 \  
  --policy-store-id PSEXAMPEabcdefg111111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang templat kebijakan, lihat [templat kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [DeletePolicyTemplate](#) di Referensi AWS CLI Perintah.

## delete-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-policy`.

### AWS CLI

Untuk menghapus kebijakan statis atau terkait templat

`delete-policy` Contoh berikut menghapus kebijakan yang memiliki Id yang ditentukan.

```
aws verifiedpermissions delete-policy \  
  --policy-id SPEXAMPEabcdefg111111 \  
  --policy-store-id PSEXAMPEabcdefg111111
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya tentang kebijakan, lihat [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## get-identity-source

Contoh kode berikut menunjukkan cara menggunakan `get-identity-source`.

### AWS CLI

Untuk mengambil detail tentang sumber identitas

`get-identity-source` Contoh berikut menampilkan rincian untuk sumber identitas dengan Id yang ditentukan.

```
aws verifiedpermissions get-identity-source \  
  --identity-source ISEXAMPLEabcdefgh111111 \  
  --policy-store-id PSEXAMPLEabcdefgh111111
```

Output:

```
{  
  "createdDate": "2023-06-12T22:27:49.150035+00:00",  
  "details": {  
    "clientIds": [ "a1b2c3d4e5f6g7h8i9j0kalbmc" ],  
    "discoveryUrl": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_1a2b3c4d5",  
    "openIdIssuer": "COGNITO",  
    "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-west-2_1a2b3c4d5"  
  },  
  "identitySourceId": "ISEXAMPLEabcdefgh111111",  
  "lastUpdatedDate": "2023-06-12T22:27:49.150035+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefgh111111",  
  "principalEntityType": "User"  
}
```

Untuk informasi selengkapnya tentang sumber identitas, lihat [Menggunakan Izin Terverifikasi Amazon dengan penyedia identitas](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [GetIdentitySource](#) di Referensi AWS CLI Perintah.

## get-policy-store

Contoh kode berikut menunjukkan cara menggunakan `get-policy-store`.

## AWS CLI

Untuk mengambil detail tentang toko kebijakan

`get-policy-store` Contoh berikut menampilkan detail untuk penyimpanan kebijakan dengan Id yang ditentukan.

```
aws verifiedpermissions get-policy-store \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{  
  "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEabcdefg111111",  
  "createdDate": "2023-06-05T20:16:46.225598+00:00",  
  "lastUpdatedDate": "2023-06-08T20:40:23.173691+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefg111111",  
  "validationSettings": { "mode": "OFF" }  
}
```

Untuk informasi selengkapnya tentang toko kebijakan, lihat penyimpanan [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [GetPolicyStore](#) di Referensi AWS CLI Perintah.

## `get-policy-template`

Contoh kode berikut menunjukkan cara menggunakan `get-policy-template`.

## AWS CLI

Untuk mengambil detail tentang templat kebijakan

`get-policy-template` Contoh berikut menampilkan rincian untuk template kebijakan dengan ID yang ditentukan.

```
aws verifiedpermissions get-policy-template \  
  --policy-template-id PTEXAMPLEabcdefg111111 \  
  --policy-store-id PSEXAMPLEabcdefg111111
```



**Output:**

```
{
  "createdDate": "2023-06-12T20:47:42.804511+00:00",
  "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "policyTemplateId": "PTEXAMPLEabcdefg111111",
  "statement": "permit(\n  principal in ?principal,\n  action == Action::\n  \"view\", \n  resource == Photo::\"VacationPhoto94.jpg\"\\n);"
}
```

Untuk informasi selengkapnya tentang templat kebijakan, lihat [templat kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [GetPolicyTemplate](#) di Referensi AWS CLI Perintah.

**get-policy**

Contoh kode berikut menunjukkan cara menggunakan `get-policy`.

**AWS CLI**

Untuk mengambil detail tentang kebijakan

`get-policy` Contoh berikut menampilkan detail kebijakan dengan ID yang ditentukan.

```
aws verifiedpermissions get-policy \
  --policy-id PSEXAMPLEabcdefg111111 \
  --policy-store-id PSEXAMPLEabcdefg111111
```

**Output:**

```
{
  "createdDate": "2023-06-12T20:33:37.382907+00:00",
  "definition": {
    "static": {
      "description": "Grant everyone of janeFriends UserGroup access to the vacationFolder Album",
      "statement": "permit(principal in UserGroup::\"janeFriends\", action, resource in Album::\"vacationFolder\" );"
    }
  },
}
```

```

"lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
"policyId": "SPEXAMPLEabcdefg111111",
"policyStoreId": "PSEXAMPLEabcdefg111111",
"policyType": "STATIC",
"principal": {
  "entityId": "janeFriends",
  "entityType": "UserGroup"
},
"resource": {
  "entityId": "vacationFolder",
  "entityType": "Album"
}
}

```

Untuk informasi selengkapnya tentang kebijakan, lihat [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [GetPolicy](#) di Referensi AWS CLI Perintah.

## get-schema

Contoh kode berikut menunjukkan cara menggunakan `get-schema`.

### AWS CLI

Untuk mengambil skema di toko kebijakan

`get-schema` Contoh berikut menampilkan rincian skema di toko kebijakan yang ditentukan.

```

aws verifiedpermissions get-schema \
  --policy-store-id PSEXAMPLEabcdefg111111

```

Output:

```

{
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "schema": "{\n\"MySampleNamespace\":{\n\"entityTypes\":{\n\"Employee\":{\n\"shape\n\n\":{\n\"attributes\":{\n\"jobLevel\":{\n\"type\":\n\n\"Long\n\n\"},\n\n\"name\":{\n\n\"type\":\n\n\"String\n\n\"}\n\n}},\n\n\"type\":\n\n\"Record\n\n\"}}},\n\n\"actions\":{\n\n\"remoteAccess\":{\n\n\"appliesTo\":\n\n{\n\n\"principalTypes\":[\n\n\"Employee\n\n\"]\n\n}}\n\n}}\n\n}}",
  "createdDate": "2023-06-14T17:47:13.999885+00:00",
  "lastUpdatedDate": "2023-06-14T17:47:13.999885+00:00"
}

```

```
}
```

Untuk informasi selengkapnya tentang skema, lihat [Skema penyimpanan kebijakan](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [GetSchema](#) di Referensi AWS CLI Perintah.

## **is-authorized-with-token**

Contoh kode berikut menunjukkan cara menggunakan `is-authorized-with-token`.

### AWS CLI

Contoh 1: Untuk meminta keputusan otorisasi untuk permintaan pengguna (izinkan)

`is-authorized-with-token` Contoh berikut meminta keputusan otorisasi untuk pengguna yang diautentikasi oleh Amazon Cognito. Permintaan menggunakan token identitas yang disediakan oleh Cognito daripada token akses. Dalam contoh ini, penyimpanan informasi yang ditentukan dikonfigurasi untuk mengembalikan prinsipal sebagai entitas tipe `CognitoUser`

```
aws verifiedpermissions is-authorized-with-token \  
  --action actionId="View",actionType="Action" \  
  --resource entityId="vacationPhoto94.jpg",entityType="Photo" \  
  --policy-store-id PSEXAMPLEabcdefg111111 \  
  --identity-token "AbCdE12345...long.string...54321EdCbA"
```

Toko kebijakan berisi kebijakan dengan pernyataan berikut yang menerima identitas dari kumpulan pengguna Cognito dan Id aplikasi yang ditentukan.

```
permit(  
  principal == CognitoUser::"us-east-1_1a2b3c4d5|a1b2c3d4e5f6g7h8i9j0kalbmc",  
  action,  
  resource == Photo::"VacationPhoto94.jpg"  
);
```

Output:

```
{  
  "decision":"Allow",  
  "determiningPolicies":[
```

```
    {
      "determiningPolicyId": "SPEXAMPLEabcdefg111111"
    }
  ],
  "errors": []
}
```

Untuk informasi selengkapnya tentang menggunakan identitas dari kumpulan pengguna Cognito, lihat [Menggunakan Izin Terverifikasi Amazon dengan penyedia identitas di Panduan Pengguna Izin Terverifikasi Amazon](#).

- Untuk detail API, lihat [IsAuthorizedWithToken](#) di Referensi AWS CLI Perintah.

## is-authorized

Contoh kode berikut menunjukkan cara menggunakan `is-authorized`.

### AWS CLI

Contoh 1: Untuk meminta keputusan otorisasi untuk permintaan pengguna (izinkan)

`is-authorized` Contoh berikut meminta keputusan otorisasi untuk kepala sekolah tipe User bernama Alice, yang ingin melakukan `updatePhoto` operasi, pada sumber daya tipe Photo bernama `VacationPhoto94.jpg`.

Tanggapan menunjukkan bahwa permintaan diizinkan oleh satu kebijakan.

```
aws verifiedpermissions is-authorized \
  --principal entityType=User,entityId=alice \
  --action actionType=Action,actionId=view \
  --resource entityType=Photo,entityId=VacationPhoto94.jpg \
  --policy-store-id SPEXAMPLEabcdefg111111
```

Output:

```
{
  "decision": "ALLOW",
  "determiningPolicies": [
    {
      "policyId": "SPEXAMPLEabcdefg111111"
    }
  ]
}
```

```
    ],  
    "errors": []  
}
```

Contoh 2: Untuk meminta keputusan otorisasi untuk permintaan pengguna (tolak)

Contoh berikut adalah sama dengan contoh sebelumnya, kecuali bahwa prinsipal adalah `User::"Bob"`. Toko kebijakan tidak berisi kebijakan apa pun yang memungkinkan pengguna tersebut mengakses `Album::"alice_folder"`.

Output menunjukkan Deny bahwa implisit karena daftar `DeterminingPolicies` kosong.

```
aws verifiedpermissions create-policy \  
  --definition file://definition2.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{  
  "decision": "DENY",  
  "determiningPolicies": [],  
  "errors": []  
}
```

Untuk informasi selengkapnya, lihat [Panduan Pengguna Izin Terverifikasi Amazon](#).

- Untuk detail API, lihat [IsAuthorized](#) di Referensi AWS CLI Perintah.

## list-identity-sources

Contoh kode berikut menunjukkan cara menggunakan `list-identity-sources`.

### AWS CLI

Untuk daftar sumber identitas yang tersedia

`list-identity-sources` Contoh berikut mencantumkan semua sumber identitas di toko kebijakan yang ditentukan.

```
aws verifiedpermissions list-identity-sources \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

```
--policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
  "identitySources": [
    {
      "createdDate": "2023-06-12T22:27:49.150035+00:00",
      "details": {
        "clientIds": [ "a1b2c3d4e5f6g7h8i9j0kalbmc" ],
        "discoveryUrl": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_1a2b3c4d5",
        "openIdIssuer": "COGNITO",
        "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/
us-west-2_1a2b3c4d5"
      },
      "identitySourceId": "ISEXAMPLEabcdefg111111",
      "lastUpdatedDate": "2023-06-12T22:27:49.150035+00:00",
      "policyStoreId": "PSEXAMPLEabcdefg111111",
      "principalEntityType": "User"
    }
  ]
}
```

Untuk informasi selengkapnya tentang sumber identitas, lihat [Menggunakan Izin Terverifikasi Amazon dengan penyedia identitas](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [ListIdentitySources](#) di Referensi AWS CLI Perintah.

## list-policies

Contoh kode berikut menunjukkan cara menggunakan `list-policies`.

### AWS CLI

Untuk daftar kebijakan yang tersedia

`list-policies` Contoh berikut mencantumkan semua kebijakan di toko kebijakan yang ditentukan.

```
aws verifiedpermissions list-policies \
  --policy-store-id PSEXAMPLEabcdefg111111
```

## Output:

```
{
  "policies": [
    {
      "createdDate": "2023-06-12T20:33:37.382907+00:00",
      "definition": {
        "static": {
          "description": "Grant everyone of janeFriends UserGroup access
to the vacationFolder Album"
        }
      },
      "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
      "policyId": "SPEXAMPLEabcdefg111111",
      "policyStoreId": "PSEXAMPLEabcdefg111111",
      "policyType": "STATIC",
      "principal": {
        "entityId": "janeFriends",
        "entityType": "UserGroup"
      },
      "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
      }
    },
    {
      "createdDate": "2023-06-12T20:39:44.975897+00:00",
      "definition": {
        "static": {
          "description": "Grant everyone access to the publicFolder Album"
        }
      },
      "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
      "policyId": "SPEXAMPLEabcdefg222222",
      "policyStoreId": "PSEXAMPLEabcdefg111111",
      "policyType": "STATIC",
      "resource": {
        "entityId": "publicFolder",
        "entityType": "Album"
      }
    },
    {
      "createdDate": "2023-06-12T20:49:51.490211+00:00",
      "definition": {
```

```
        "templateLinked": {
            "policyTemplateId": "PTEXAMPLEabcdefg111111"
        }
    },
    "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",
    "policyId": "SPEXAMPLEabcdefg333333",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "TEMPLATE_LINKED",
    "principal": {
        "entityId": "alice",
        "entityType": "User"
    },
    "resource": {
        "entityId": "VacationPhoto94.jpg",
        "entityType": "Photo"
    }
}
]
```

Untuk informasi selengkapnya tentang kebijakan, lihat [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [ListPolicies](#) di Referensi AWS CLI Perintah.

## list-policy-stores

Contoh kode berikut menunjukkan cara menggunakan `list-policy-stores`.

### AWS CLI

Untuk daftar toko kebijakan yang tersedia

`list-policy-stores` Contoh berikut mencantumkan semua toko kebijakan di AWS Wilayah. Semua perintah untuk Izin Terverifikasi kecuali `create-policy-store` dan `list-policy-stores` mengharuskan Anda menentukan Id penyimpanan kebijakan yang ingin Anda gunakan.

```
aws verifiedpermissions list-policy-stores
```

Output:

```
{
```



```
"policyStores": [  
  {  
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEabcdefg111111",  
    "createdDate": "2023-06-05T20:16:46.225598+00:00",  
    "policyStoreId": "PSEXAMPLEabcdefg111111"  
  },  
  {  
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEabcdefg222222",  
    "createdDate": "2023-06-08T18:09:37.364356+00:00",  
    "policyStoreId": "PSEXAMPLEabcdefg222222"  
  },  
  {  
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEabcdefg333333",  
    "createdDate": "2023-06-08T18:09:46.920600+00:00",  
    "policyStoreId": "PSEXAMPLEabcdefg333333"  
  }  
]
```

Untuk informasi selengkapnya tentang toko kebijakan, lihat penyimpanan [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [ListPolicyStores](#) di Referensi AWS CLI Perintah.

## list-policy-templates

Contoh kode berikut menunjukkan cara menggunakan `list-policy-templates`.

### AWS CLI

Untuk membuat daftar templat kebijakan yang tersedia

`list-policy-templates` Contoh berikut mencantumkan semua templat kebijakan di toko kebijakan yang ditentukan.

```
aws verifiedpermissions list-policy-templates \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
  "policyTemplates": [
    {
      "createdDate": "2023-06-12T20:47:42.804511+00:00",
      "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
      "policyStoreId": "PSEXAMPLEabcdefg111111",
      "policyTemplateId": "PTEXAMPLEabcdefg111111"
    }
  ]
}
```

Untuk informasi selengkapnya tentang templat kebijakan, lihat [templat kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [ListPolicyTemplates](#) di Referensi AWS CLI Perintah.

## put -schema

Contoh kode berikut menunjukkan cara menggunakan `put -schema`.

### AWS CLI

Untuk menyimpan skema ke toko kebijakan

`put -schema` Contoh berikut membuat atau menggantikan skema di toko kebijakan tertentu.

`cedarJsonParameter` dalam file input mengambil representasi string dari objek JSON. Ini berisi tanda kutip tertanam (") dalam pasangan tanda kutip terluar. Ini mengharuskan Anda untuk mengonversi JSON menjadi string dengan mendahului semua tanda kutip yang disematkan dengan karakter garis miring terbalik (") dan menggabungkan semua baris menjadi satu baris teks tanpa jeda baris.

Contoh string dapat ditampilkan dibungkus di beberapa baris di sini untuk keterbacaan, tetapi operasi memerlukan parameter dikirimkan sebagai string baris tunggal.

```
aws verifiedpermissions put-schema --definition file: //schema.txt -- PSExampleabcdefg111111
policy-store-id
```

Isi dari `schema.txt`:

```
{
```

```
"cedarJson": "{\\"MySampleNamespace\\": {\\"actions\\": {\\"remoteAccess\\": {
  \\"appliesTo\\": {\\"principalTypes\\": [\\"Employee\\"]}}},\\"entityTypes\\": {
  \\"Employee\\": {\\"shape\\": {\\"attributes\\": {\\"jobLevel\\": {\\"type\\":
  \\"Long\\"},\\"name\\": {\\"type\\": \\"String\\"}},\\"type\\": \\"Record\\"}}}}}"
}
```

### Output:

```
{
  "policyStoreId": "PSEXAMPLEabcdefghijklmnop111111",
  "namespaces": [
    "MySampleNamespace"
  ],
  "createdDate": "2023-06-14T17:47:13.999885+00:00",
  "lastUpdatedDate": "2023-06-14T17:47:13.999885+00:00"
}
```

Untuk informasi selengkapnya tentang skema, lihat [Skema penyimpanan kebijakan](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [PutSchema](#) di Referensi AWS CLI Perintah.

## update-identity-source

Contoh kode berikut menunjukkan cara menggunakan `update-identity-source`.

### AWS CLI

Untuk memperbarui sumber identitas

`update-identity-source` Contoh berikut memodifikasi sumber identitas yang ditentukan dengan menyediakan konfigurasi kumpulan pengguna Cognito baru dan mengubah jenis entitas yang dikembalikan oleh sumber identitas.

```
aws verifiedpermissions update-identity-source
  --identity-source-id ISEXAMPLEabcdefghijklmnop111111 \
  --update-configuration file://config.txt \
  --principal-entity-type "Employee" \
  --policy-store-id PSEXAMPLEabcdefghijklmnop111111
```

Isi dari `config.txt`:

```
{
  "cognitoUserPoolConfiguration": {
    "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-west-2_1a2b3c4d5",
    "clientIds": ["a1b2c3d4e5f6g7h8i9j0kalbmc"]
  }
}
```

Output:

```
{
  "createdDate": "2023-05-19T20:30:28.214829+00:00",
  "identitySourceId": "ISEXAMPLEabcdefgh111111",
  "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
  "policyStoreId": "PSEXAMPLEabcdefgh111111"
}
```

Untuk informasi selengkapnya tentang sumber identitas, lihat [Menggunakan Izin Terverifikasi Amazon dengan penyedia identitas](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [UpdateIdentitySource](#) di Referensi AWS CLI Perintah.

## update-policy-store

Contoh kode berikut menunjukkan cara menggunakan `update-policy-store`.

### AWS CLI

Untuk memperbarui toko kebijakan

`update-policy-store` Contoh berikut memodifikasi penyimpanan kebijakan dengan mengubah setelan validasinya.

```
aws verifiedpermissions update-policy-store \
  --validation-settings "mode=STRICT" \
  --policy-store-id PSEXAMPLEabcdefgh111111
```

Output:

```
{
```

```
"arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
  "createdDate": "2023-05-16T17:41:29.103459+00:00",
  "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",
  "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

Untuk informasi selengkapnya tentang toko kebijakan, lihat penyimpanan [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [UpdatePolicyStore](#) di Referensi AWS CLI Perintah.

## update-policy-template

Contoh kode berikut menunjukkan cara menggunakan `update-policy-template`.

### AWS CLI

Contoh 1: Untuk memperbarui templat kebijakan

`update-policy-template` Contoh berikut memodifikasi kebijakan terkait templat yang ditentukan untuk menggantikan pernyataan kebijakannya.

```
aws verifiedpermissions update-policy-template \
  --policy-template-id PTEXAMPLEabcdefg111111 \
  --statement file://template1.txt \
  --policy-store-id PSEXAMPLEabcdefg111111
```

Isi file `template1.txt`:

```
permit(
  principal in ?principal,
  action == Action::"view",
  resource == Photo::"VacationPhoto94.jpg"
);
```

Output:

```
{
  "createdDate": "2023-06-12T20:47:42.804511+00:00",
  "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
  "policyStoreId": "PSEXAMPLEabcdefg111111",
```

```
"policyTemplateId": "PTEXAMPLEabcdefg111111"  
}
```

Untuk informasi selengkapnya tentang templat kebijakan, lihat [templat kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [UpdatePolicyTemplate](#) di Referensi AWS CLI Perintah.

## update-policy

Contoh kode berikut menunjukkan cara menggunakan `update-policy`.

### AWS CLI

Contoh 1: Untuk membuat kebijakan statis

`create-policy` Contoh berikut membuat kebijakan statis dengan cakupan kebijakan yang menentukan prinsipal dan sumber daya.

```
aws iam create-policy \
  --definition file://definition.txt \
  --policy-store-id PSEXAMPLEabcdefg111111
```

`statementParameter` mengambil representasi string dari objek JSON. Ini berisi tanda kutip tertanam (") dalam pasangan tanda kutip terluar. Ini mengharuskan Anda untuk mengonversi JSON menjadi string dengan mendahului semua tanda kutip yang disematkan dengan karakter garis miring terbalik (") dan menggabungkan semua baris menjadi satu baris teks tanpa jeda baris.

Contoh string dapat ditampilkan dibungkus di beberapa baris di sini untuk keterbacaan, tetapi operasi memerlukan parameter dikirimkan sebagai string baris tunggal.

Isi file `definition.txt`:

```
{  
  "static": {  
    "description": "Grant everyone of janeFriends UserGroup access to the  
vacationFolder Album",  
    "statement": "permit(principal in UserGroup::\\"janeFriends\\", action,  
resource in Album::\\"vacationFolder\\" );"  
  }  
}
```

```
}
```

**Output:**

```
{
  "createdDate": "2023-06-12T20:33:37.382907+00:00",
  "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
  "policyId": "SPEXAMPLEabcdefg111111",
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "policyType": "STATIC",
  "principal": {
    "entityId": "janeFriends",
    "entityType": "UserGroup"
  },
  "resource": {
    "entityId": "vacationFolder",
    "entityType": "Album"
  }
}
```

Contoh 2: Untuk membuat kebijakan statis yang memberikan akses ke sumber daya kepada semua orang

`create-policy` Contoh berikut membuat kebijakan statis dengan cakupan kebijakan yang hanya menentukan sumber daya.

```
aws verifiedpermissions create-policy \
  --definition file://definition2.txt \
  --policy-store-id PSEXAMPLEabcdefg111111
```

Isi file `definition2.txt`:

```
{
  "static": {
    "description": "Grant everyone access to the publicFolder Album",
    "statement": "permit(principal, action, resource in Album:\""publicFolder
  \");"
  }
}
```

**Output:**

```
{
  "createdDate": "2023-06-12T20:39:44.975897+00:00",
  "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
  "policyId": "PbfR73F8oh5MMfr9uRtFDB",
  "policyStoreId": "PSEXAMPLEEabcdefg222222",
  "policyType": "STATIC",
  "resource": {
    "entityId": "publicFolder",
    "entityType": "Album"
  }
}
```

Contoh 3: Untuk membuat kebijakan terkait template yang dikaitkan dengan template yang ditentukan

`create-policy` Contoh berikut membuat kebijakan terkait templat menggunakan templat kebijakan yang ditentukan dan mengaitkan prinsipal yang ditentukan untuk digunakan dengan kebijakan terkait templat yang baru.

```
aws verifiedpermissions create-policy \
  --definition file://definition2.txt \
  --policy-store-id PSEXAMPLEEabcdefg111111
```

Isi dari `definition3.txt`:

```
{
  "templateLinked": {
    "policyTemplateId": "PTEXAMPLEEabcdefg111111",
    "principal": {
      "entityType": "User",
      "entityId": "alice"
    }
  }
}
```

Output:

```
{
  "createdDate": "2023-06-12T20:49:51.490211+00:00",
  "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",
  "policyId": "TPEXAMPLEEabcdefg111111",
```



```
"policyStoreId": "PSEXAMPLEabcdefg111111",
"policyType": "TEMPLATE_LINKED",
"principal": {
  "entityId": "alice",
  "entityType": "User"
},
"resource": {
  "entityId": "VacationPhoto94.jpg",
  "entityType": "Photo"
}
}
```

Untuk informasi selengkapnya tentang kebijakan, lihat [kebijakan Izin Terverifikasi Amazon](#) di Panduan Pengguna Izin Terverifikasi Amazon.

- Untuk detail API, lihat [UpdatePolicy](#) di Referensi AWS CLI Perintah.

## Contoh VPC Lattice menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With VPC Lattice.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-listener**

Contoh kode berikut menunjukkan cara menggunakan `create-listener`.

## AWS CLI

Untuk membuat pendengar

`create-listener` Contoh berikut membuat pendengar HTTPS dengan aturan default yang meneruskan lalu lintas ke grup target VPC Lattice yang ditentukan.

```
aws vpc-lattice create-listener \  
  --name my-service-listener \  
  --protocol HTTPS \  
  --port 443 \  
  --service-identifier svc-0285b53b2eEXAMPLE \  
  --default-action file://listener-config.json
```

Isi dari `listener-config.json`:

```
{  
  "forward": {  
    "targetGroups": [  
      {  
        "targetGroupIdentifier": "tg-0eaa4b9ab4EXAMPLE"  
      }  
    ]  
  }  
}
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
svc-0285b53b2eEXAMPLE/listener/listener-07cc7fb0abEXAMPLE",  
  "defaultAction": {  
    "forward": {  
      "targetGroups": [  
        {  
          "targetGroupIdentifier": "tg-0eaa4b9ab4EXAMPLE",  
          "weight": 100  
        }  
      ]  
    }  
  },  
  "id": "listener-07cc7fb0abEXAMPLE",
```

```
"name": "my-service-listener",
"port": 443,
"protocol": "HTTPS",
"serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
"serviceId": "svc-0285b53b2eEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Pendengar](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [CreateListener](#) di Referensi AWS CLI Perintah.

## create-service-network-service-association

Contoh kode berikut menunjukkan cara menggunakan `create-service-network-service-association`.

### AWS CLI

Untuk membuat asosiasi layanan

`create-service-network-service-association` Contoh berikut mengaitkan layanan yang ditentukan dengan jaringan layanan yang ditentukan.

```
aws vpc-lattice create-service-network-service-association \
  --service-identifier svc-0285b53b2eEXAMPLE \
  --service-network-identifier sn-080ec7dc93EXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkserviceassociation/snsa-0e16955a8cEXAMPLE",
  "createdBy": "123456789012",
  "dnsEntry": {
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CEXAMPLE"
  },
  "id": "snsa-0e16955a8cEXAMPLE",
  "status": "CREATE_IN_PROGRESS"
}
```

```
}
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [CreateServiceNetworkServiceAssociation](#) di Referensi AWS CLI Perintah.

## **create-service-network-vpc-association**

Contoh kode berikut menunjukkan cara menggunakan `create-service-network-vpc-association`.

### AWS CLI

Untuk membuat asosiasi VPC

`create-service-network-vpc-association` Contoh berikut mengaitkan vpc yang ditentukan dengan jaringan layanan yang ditentukan. Grup keamanan yang ditentukan mengontrol sumber daya mana di VPC dapat mengakses jaringan layanan dan layanannya.

```
aws vpc-lattice create-service-network-vpc-association \  
  --vpc-identifier vpc-0a1b2c3d4eEXAMPLE \  
  --service-network-identifier sn-080ec7dc93EXAMPLE \  
  --security-group-ids sg-0aee16bc6cEXAMPLE
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/  
snva-0821fc8631EXAMPLE",  
  "createdBy": "123456789012",  
  "id": "snva-0821fc8631EXAMPLE",  
  "securityGroupIds": [  
    "sg-0aee16bc6cEXAMPLE"  
  ],  
  "status": "CREATE_IN_PROGRESS"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi VPC](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [CreateServiceNetworkVpcAssociation](#) di Referensi AWS CLI Perintah.

## **create-service-network**

Contoh kode berikut menunjukkan cara menggunakan `create-service-network`.

### AWS CLI

Untuk membuat jaringan layanan

`create-service-network` Contoh berikut membuat jaringan layanan dengan nama yang ditentukan.

```
aws vpc-lattice create-service-network \  
  --name my-service-network
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/  
sn-080ec7dc93EXAMPLE",  
  "authType": "NONE",  
  "id": "sn-080ec7dc93EXAMPLE",  
  "name": "my-service-network"  
}
```

Untuk informasi selengkapnya, lihat [Jaringan layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [CreateServiceNetwork](#) di Referensi AWS CLI Perintah.

## **create-service**

Contoh kode berikut menunjukkan cara menggunakan `create-service`.

### AWS CLI

Untuk membuat layanan

`create-service` Contoh berikut membuat layanan dengan nama yang ditentukan.

```
aws vpc-lattice create-service \  
  --name my-service
```

```
--name my-lattice-service
```

### Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
  "authType": "NONE",
  "dnsEntry": {
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.1a2b3c4.vpc-lattice-
svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CEXAMPLE"
  },
  "id": "svc-0285b53b2eEXAMPLE",
  "name": "my-lattice-service",
  "status": "CREATE_IN_PROGRESS"
}
```

Untuk informasi selengkapnya, lihat [Layanan di Kisi VPC di Panduan Pengguna Amazon VPC Lattice](#).

- Untuk detail API, lihat [CreateService](#) di Referensi AWS CLI Perintah.

## create-target-group

Contoh kode berikut menunjukkan cara menggunakan `create-target-group`.

### AWS CLI

Contoh 1: Untuk membuat grup target tipe INSTANCE

`create-target-group` Contoh berikut membuat grup target dengan nama, jenis, dan konfigurasi yang ditentukan.

```
aws vpc-lattice create-target-group \
  --name my-lattice-target-group-instance \
  --type INSTANCE \
  --config file://tg-config.json
```

Isi dari `tg-config.json`:

```
{
```

```
"port": 443,  
"protocol": "HTTPS",  
"protocolVersion": "HTTP1",  
"vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
}
```

### Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-0eaa4b9ab4EXAMPLE",  
  "config": {  
    "healthCheck": {  
      "enabled": true,  
      "healthCheckIntervalSeconds": 30,  
      "healthCheckTimeoutSeconds": 5,  
      "healthyThresholdCount": 5,  
      "matcher": {  
        "httpCode": "200"  
      },  
      "path": "/",  
      "protocol": "HTTPS",  
      "protocolVersion": "HTTP1",  
      "unhealthyThresholdCount": 2  
    },  
    "port": 443,  
    "protocol": "HTTPS",  
    "protocolVersion": "HTTP1",  
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
  },  
  "id": "tg-0eaa4b9ab4EXAMPLE",  
  "name": "my-lattice-target-group-instance",  
  "status": "CREATE_IN_PROGRESS",  
  "type": "INSTANCE"  
}
```

### Contoh 2: Untuk membuat grup target tipe IP

`create-target-group` Contoh berikut membuat grup target dengan nama, jenis, dan konfigurasi yang ditentukan.

```
aws vpc-lattice create-target-group \  

```

```
--name my-lattice-target-group-ip \  
--type IP \  
--config file://tg-config.json
```

Isi dari tg-config.json:

```
{  
  "ipAddressType": "IPV4",  
  "port": 443,  
  "protocol": "HTTPS",  
  "protocolVersion": "HTTP1",  
  "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
}
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-0eaa4b9ab4EXAMPLE",  
  "config": {  
    "healthCheck": {  
      "enabled": true,  
      "healthCheckIntervalSeconds": 30,  
      "healthCheckTimeoutSeconds": 5,  
      "healthyThresholdCount": 5,  
      "matcher": {  
        "httpCode": "200"  
      },  
      "path": "/",  
      "protocol": "HTTPS",  
      "protocolVersion": "HTTP1",  
      "unhealthyThresholdCount": 2  
    },  
    "ipAddressType": "IPV4",  
    "port": 443,  
    "protocol": "HTTPS",  
    "protocolVersion": "HTTP1",  
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
  },  
  "id": "tg-0eaa4b9ab4EXAMPLE",  
  "name": "my-lattice-target-group-ip",  
  "status": "CREATE_IN_PROGRESS",  
  "type": "IP"
```



```
}
```

### Contoh 3: Untuk membuat grup target tipe LAMBDA

`create-target-group` Contoh berikut membuat grup target dengan nama, jenis, dan konfigurasi yang ditentukan.

```
aws vpc-lattice create-target-group \  
  --name my-lattice-target-group-lambda \  
  --type LAMBDA
```

### Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-0eaa4b9ab4EXAMPLE",  
  "id": "tg-0eaa4b9ab4EXAMPLE",  
  "name": "my-lattice-target-group-lambda",  
  "status": "CREATE_IN_PROGRESS",  
  "type": "LAMBDA"  
}
```

### Contoh 4: Untuk membuat grup target tipe ALB

`create-target-group` Contoh berikut membuat grup target dengan nama, jenis, dan konfigurasi yang ditentukan.

```
aws vpc-lattice create-target-group \  
  --name my-lattice-target-group-alb \  
  --type ALB \  
  --config file://tg-config.json
```

### Isi dari `tg-config.json`:

```
{  
  "port": 443,  
  "protocol": "HTTPS",  
  "protocolVersion": "HTTP1",  
  "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
}
```

## Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
  "config": {
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
  },
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "name": "my-lattice-target-group-alb",
  "status": "CREATE_IN_PROGRESS",
  "type": "ALB"
}
```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [CreateTargetGroup](#) di Referensi AWS CLI Perintah.

## delete-auth-policy

Contoh kode berikut menunjukkan cara menggunakan `delete-auth-policy`.

### AWS CLI

Untuk menghapus kebijakan autentikasi

`delete-auth-policy` Contoh berikut menghapus kebijakan autentikasi untuk layanan yang ditentukan.

```
aws vpc-lattice delete-auth-policy \
  --resource-identifier svc-0285b53b2eEXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Kebijakan autentikasi](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeleteAuthPolicy](#) di Referensi AWS CLI Perintah.

## delete-listener

Contoh kode berikut menunjukkan cara menggunakan `delete-listener`.

### AWS CLI

Untuk menghapus pendengar

`delete-listener` Contoh berikut menghapus pendengar yang ditentukan.

```
aws vpc-lattice delete-listener \  
  --listener-identifier listener-07cc7fb0abEXAMPLE \  
  --service-identifier svc-0285b53b2eEXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Pendengar](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeleteListener](#) di Referensi AWS CLI Perintah.

## delete-service-network-service-association

Contoh kode berikut menunjukkan cara menggunakan `delete-service-network-service-association`.

### AWS CLI

Untuk menghapus asosiasi layanan

`delete-service-network-service-association` Contoh berikut memisahkan asosiasi layanan yang ditentukan.

```
aws vpc-lattice delete-service-network-service-association \  
  --service-network-service-association-identifier sns-a-031fabb4d8EXAMPLE
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-  
east-2:123456789012:servicenetworkserviceassociation/sns-a-031fabb4d8EXAMPLE",  
  "id": "sns-a-031fabb4d8EXAMPLE",
```

```
"status": "DELETE_IN_PROGRESS"
}
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeleteServiceNetworkServiceAssociation](#) di Referensi AWS CLI Perintah.

## **delete-service-network-vpc-association**

Contoh kode berikut menunjukkan cara menggunakan `delete-service-network-vpc-association`.

### AWS CLI

Untuk menghapus asosiasi VPC

`delete-service-network-vpc-association` Contoh berikut memisahkan asosiasi VPC yang ditentukan.

```
aws vpc-lattice delete-service-network-vpc-association \
  --service-network-vpc-association-identifier snva-0821fc8631EXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
snva-0821fc8631EXAMPLE",
  "id": "snva-0821fc8631EXAMPLE",
  "status": "DELETE_IN_PROGRESS"
}
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi VPC](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeleteServiceNetworkVpcAssociation](#) di Referensi AWS CLI Perintah.

## **delete-service-network**

Contoh kode berikut menunjukkan cara menggunakan `delete-service-network`.

## AWS CLI

Untuk menghapus jaringan layanan

`delete-service-network` Contoh berikut menghapus jaringan layanan yang ditentukan.

```
aws vpc-lattice delete-service-network \  
  --service-network-identifier sn-080ec7dc93EXAMPLE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Jaringan layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeleteServiceNetwork](#) di Referensi AWS CLI Perintah.

## `delete-service`

Contoh kode berikut menunjukkan cara menggunakan `delete-service`.

## AWS CLI

Untuk menghapus layanan

`delete-service` Contoh berikut menghapus layanan yang ditentukan.

```
aws vpc-lattice delete-service \  
  --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-west-2:123456789012:service/  
svc-0285b53b2eEXAMPLE",  
  "id": "svc-0285b53b2eEXAMPLE",  
  "name": "my-lattice-service",  
  "status": "DELETE_IN_PROGRESS"  
}
```

Untuk informasi selengkapnya, lihat [Layanan di Kisi VPC di Panduan Pengguna Amazon VPC Lattice](#).

- Untuk detail API, lihat [DeleteService](#) di Referensi AWS CLI Perintah.

## **delete-target-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-target-group`.

### AWS CLI

Untuk menghapus grup target

`delete-target-group` Contoh berikut menghapus kelompok target yang ditentukan.

```
aws vpc-lattice delete-target-group \  
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-0eaa4b9ab4EXAMPLE",  
  "id": "tg-0eaa4b9ab4EXAMPLE",  
  "status": "DELETE_IN_PROGRESS"  
}
```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeleteTargetGroup](#) di Referensi AWS CLI Perintah.

## **deregister-targets**

Contoh kode berikut menunjukkan cara menggunakan `deregister-targets`.

### AWS CLI

Untuk membatalkan pendaftaran target

`deregister-targets` Contoh berikut membatalkan pendaftaran target yang ditentukan dari kelompok target yang ditentukan.

```
aws vpc-lattice deregister-targets \  
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

```
--targets i-07dd579bc5EXAMPLE \  
--target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{  
  "successful": [  
    {  
      "id": "i-07dd579bc5EXAMPLE",  
      "port": 443  
    }  
  ],  
  "unsuccessful": []  
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan target](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [DeregisterTargets](#) di Referensi AWS CLI Perintah.

## get-auth-policy

Contoh kode berikut menunjukkan cara menggunakan `get-auth-policy`.

### AWS CLI

Untuk mendapatkan informasi tentang kebijakan autentikasi

`get-auth-policy` Contoh berikut mendapatkan informasi tentang kebijakan autentikasi untuk layanan yang ditentukan.

```
aws vpc-lattice get-auth-policy \  
--resource-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{  
  "createdAt": "2023-06-07T03:51:20.266Z",  
  "lastUpdatedAt": "2023-06-07T04:39:27.082Z",  
  "policy": "{\n\"Version\": \"2012-10-17\",  
\"Statement\": [\n{\n\"Effect\": \"Allow\",  
\"Principal\": {\n\"AWS\": \"arn:aws:iam::123456789012:role/my-clients\"},
```

```
\ "Action": "\ "vpc-lattice-svcs:Invoke", "\ "Resource": "\ "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE"} ] ]",
  "state": "Active"
}
```

Untuk informasi selengkapnya, lihat [Kebijakan autentikasi](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetAuthPolicy](#) di Referensi AWS CLI Perintah.

## get-listener

Contoh kode berikut menunjukkan cara menggunakan `get-listener`.

### AWS CLI

Untuk mendapatkan informasi tentang pendengar layanan

`get-listener` Contoh berikut mendapatkan informasi tentang listener tertentu untuk layanan tertentu.

```
aws vpc-lattice get-listener \
  --listener-identifier listener-0ccf55918cEXAMPLE \
  --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE/listener/listener-0ccf55918cEXAMPLE",
  "createdAt": "2023-05-07T05:08:45.192Z",
  "defaultAction": {
    "forward": {
      "targetGroups": [
        {
          "targetGroupIdentifier": "tg-0ff213abb6EXAMPLE",
          "weight": 1
        }
      ]
    }
  },
  "id": "listener-0ccf55918cEXAMPLE",
```



```
"lastUpdatedAt": "2023-05-07T05:08:45.192Z",
"name": "http-80",
"port": 80,
"protocol": "HTTP",
"serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
"serviceId": "svc-0285b53b2eEXAMPLE"
}
```

Untuk informasi selengkapnya, lihat [Mendefinisikan perutean](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetListener](#) di Referensi AWS CLI Perintah.

## get-service-network-service-association

Contoh kode berikut menunjukkan cara menggunakan `get-service-network-service-association`.

### AWS CLI

Untuk mendapatkan informasi tentang asosiasi layanan

`get-service-network-service-association` Contoh berikut mendapat informasi tentang asosiasi layanan yang ditentukan.

```
aws vpc-lattice get-service-network-service-association \
--service-network-service-association-identifier sns-a-031fabb4d8EXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkserviceassociation/sns-a-031fabb4d8EXAMPLE",
  "createdAt": "2023-05-05T21:48:16.076Z",
  "createdBy": "123456789012",
  "dnsEntry": {
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CEXAMPLE"
  },
}
```

```
    "id": "sna-031fabb4d8EXAMPLE",
    "serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "serviceId": "svc-0285b53b2eEXAMPLE",
    "serviceName": "my-lattice-service",
    "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
    "serviceNetworkId": "sn-080ec7dc93EXAMPLE",
    "serviceNetworkName": "my-service-network",
    "status": "ACTIVE"
}
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetServiceNetworkServiceAssociation](#) di Referensi AWS CLI Perintah.

## get-service-network-vpc-association

Contoh kode berikut menunjukkan cara menggunakan `get-service-network-vpc-association`.

### AWS CLI

Untuk mendapatkan informasi tentang asosiasi VPC

`get-service-network-vpc-association` Contoh berikut mendapat informasi tentang asosiasi VPC tertentu.

```
aws vpc-lattice get-service-network-vpc-association \
  --service-network-vpc-association-identifier sna-0821fc8631EXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
sna-0821fc8631EXAMPLE",
  "createdAt": "2023-06-06T23:41:08.421Z",
  "createdBy": "123456789012",
  "id": "sna-0c5dcb60d6EXAMPLE",
  "lastUpdatedAt": "2023-06-06T23:41:08.421Z",
```

```

    "securityGroupIds": [
      "sg-0aee16bc6cEXAMPLE"
    ],
    "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
    "serviceNetworkId": "sn-080ec7dc93EXAMPLE",
    "serviceNetworkName": "my-service-network",
    "status": "ACTIVE",
    "vpcId": "vpc-0a1b2c3d4eEXAMPLE"
  }

```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi VPC](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetServiceNetworkVpcAssociation](#) di Referensi AWS CLI Perintah.

## get-service-network

Contoh kode berikut menunjukkan cara menggunakan `get-service-network`.

### AWS CLI

Untuk mendapatkan informasi tentang jaringan layanan

`get-service-network` Contoh berikut mendapat informasi tentang jaringan layanan yang ditentukan.

```

aws vpc-lattice get-service-network \
  --service-network-identifier sn-080ec7dc93EXAMPLE

```

Output:

```

{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
  "authType": "AWS_IAM",
  "createdAt": "2023-05-05T15:26:08.417Z",
  "id": "sn-080ec7dc93EXAMPLE",
  "lastUpdatedAt": "2023-05-05T15:26:08.417Z",
  "name": "my-service-network",
  "numberOfAssociatedServices": 2,
  "numberOfAssociatedVPCs": 3
}

```

```
}
```

Untuk informasi selengkapnya, lihat [Jaringan layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetServiceNetwork](#) di Referensi AWS CLI Perintah.

## get-service

Contoh kode berikut menunjukkan cara menggunakan `get-service`.

### AWS CLI

Untuk mendapatkan informasi tentang layanan

`get-service` Contoh berikut mendapat informasi tentang layanan yang ditentukan.

```
aws vpc-lattice get-service \
  --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
  "authType": "AWS_IAM",
  "createdAt": "2023-05-05T21:35:29.339Z",
  "dnsEntry": {
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-
svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CFUOHIZH"
  },
  "id": "svc-0285b53b2eEXAMPLE",
  "lastUpdatedAt": "2023-05-05T21:35:29.339Z",
  "name": "my-lattice-service",
  "status": "ACTIVE"
}
```

Untuk informasi selengkapnya, lihat [Layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetService](#) di Referensi AWS CLI Perintah.

## get-target-group

Contoh kode berikut menunjukkan cara menggunakan `get-target-group`.

### AWS CLI

Untuk mendapatkan informasi tentang kelompok sasaran

`get-target-group` Contoh berikut mendapat informasi tentang kelompok target yang ditentukan, yang memiliki jenis target `INSTANCE`.

```
aws vpc-lattice get-target-group \
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
  "config": {
    "healthCheck": {
      "enabled": true,
      "healthCheckIntervalSeconds": 30,
      "healthCheckTimeoutSeconds": 5,
      "healthyThresholdCount": 5,
      "matcher": {
        "httpCode": "200"
      },
      "path": "/",
      "protocol": "HTTPS",
      "protocolVersion": "HTTP1",
      "unhealthyThresholdCount": 2
    },
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
  },
  "createdAt": "2023-05-06T04:41:04.122Z",
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "lastUpdatedAt": "2023-05-06T04:41:04.122Z",
  "name": "my-target-group",
  "serviceArns": [
```

```
    "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE"  
  ],  
  "status": "ACTIVE",  
  "type": "INSTANCE"  
}
```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [GetTargetGroup](#) di Referensi AWS CLI Perintah.

## list-listeners

Contoh kode berikut menunjukkan cara menggunakan `list-listeners`.

### AWS CLI

Untuk daftar pendengar layanan

`list-listeners` Contoh berikut mencantumkan pendengar untuk layanan yang ditentukan.

```
aws vpc-lattice list-listeners \  
  --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{  
  "items": [  
    {  
      "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
svc-0285b53b2eEXAMPLE/listener/listener-0ccf55918cEXAMPLE",  
      "createdAt": "2023-05-07T05:08:45.192Z",  
      "id": "listener-0ccf55918cEXAMPLE",  
      "lastUpdatedAt": "2023-05-07T05:08:45.192Z",  
      "name": "http-80",  
      "port": 80,  
      "protocol": "HTTP"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mendefinisikan perutean](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListListeners](#) di Referensi AWS CLI Perintah.

## **list-service-network-service-associations**

Contoh kode berikut menunjukkan cara menggunakan `list-service-network-service-associations`.

### AWS CLI

Untuk membuat daftar asosiasi layanan

`list-service-network-service-associations` Contoh berikut mencantumkan asosiasi layanan untuk jaringan layanan tertentu. `--query` Opsi ini mencakup output ke ID asosiasi layanan.

```
aws vpc-lattice list-service-network-service-associations \
  --service-network-identifier sn-080ec7dc93EXAMPLE \
  --query items[*].id
```

Output:

```
[
  "snsa-031fabb4d8EXAMPLE",
  "snsa-0e16955a8cEXAMPLE"
]
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListServiceNetworkServiceAssociations](#) di Referensi AWS CLI Perintah.

## **list-service-network-vpc-associations**

Contoh kode berikut menunjukkan cara menggunakan `list-service-network-vpc-associations`.

### AWS CLI

Untuk membuat daftar asosiasi VPC

`list-service-network-vpc-associations` Contoh berikut mencantumkan asosiasi VPC untuk jaringan layanan yang ditentukan. `--query` Opsi ini mencakup output ke ID asosiasi VPC.

```
aws vpc-lattice list-service-network-vpc-associations \  
  --service-network-identifier sn-080ec7dc93EXAMPLE \  
  --query items[*].id
```

Output:

```
[  
  "snva-0821fc8631EXAMPLE",  
  "snva-0c5dcb60d6EXAMPLE"  
]
```

Untuk informasi selengkapnya, lihat [Mengelola asosiasi VPC](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListServiceNetworkVpcAssociations](#) di Referensi AWS CLI Perintah.

## **list-service-networks**

Contoh kode berikut menunjukkan cara menggunakan `list-service-networks`.

### AWS CLI

Untuk membuat daftar jaringan layanan Anda

`list-service-networks` Contoh berikut mencantumkan jaringan layanan yang dimiliki atau dibagikan dengan akun panggilan. `--query` Opsi ini mencakup hasil ke Nama Sumber Daya Amazon (ARN) dari jaringan layanan.

```
aws vpc-lattice list-service-networks \  
  --query items[*].arn
```

Output:

```
[  
  "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/  
sn-080ec7dc93EXAMPLE",  
  "arn:aws:vpc-lattice:us-east-2:111122223333:servicenetwork/sn-0ec4d436cfEXAMPLE"  
]
```



Untuk informasi selengkapnya, lihat [Jaringan layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListServiceNetworks](#) di Referensi AWS CLI Perintah.

## **list-services**

Contoh kode berikut menunjukkan cara menggunakan `list-services`.

### AWS CLI

Untuk daftar layanan Anda

`list-services` Contoh berikut mencantumkan layanan yang dimiliki atau dibagikan dengan akun panggilan. `--query` Opsi ini mencakup hasil ke Nama Sumber Daya Amazon (ARN) layanan.

```
aws vpc-lattice list-services \  
  --query items[*].arn
```

Output:

```
[  
  "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE",  
  "arn:aws:vpc-lattice:us-east-2:111122223333:service/svc-0b8ac96550EXAMPLE"  
]
```

Untuk informasi selengkapnya, lihat [Layanan](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListServices](#) di Referensi AWS CLI Perintah.

## **list-target-groups**

Contoh kode berikut menunjukkan cara menggunakan `list-target-groups`.

### AWS CLI

Untuk membuat daftar grup target Anda

`list-target-groups` Contoh berikut mencantumkan kelompok target dengan jenis `targetLAMBDA`.

```
aws vpc-lattice list-target-groups \  
  --target-group-type LAMBDA
```

Output:

```
{  
  "items": [  
    {  
      "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-045c1b7d9dEXAMPLE",  
      "createdAt": "2023-05-06T05:22:16.637Z",  
      "id": "tg-045c1b7d9dEXAMPLE",  
      "lastUpdatedAt": "2023-05-06T05:22:16.637Z",  
      "name": "my-target-group-lam",  
      "serviceArns": [  
        "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
svc-0285b53b2eEXAMPLE"  
      ],  
      "status": "ACTIVE",  
      "type": "LAMBDA"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListTargetGroups](#) di Referensi AWS CLI Perintah.

## list-targets

Contoh kode berikut menunjukkan cara menggunakan `list-targets`.

### AWS CLI

Untuk membuat daftar target untuk kelompok sasaran

`list-targets` Contoh berikut mencantumkan target untuk kelompok target yang ditentukan.

```
aws vpc-lattice list-targets \  
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
  "items": [
    {
      "id": "i-07dd579bc5EXAMPLE",
      "port": 443,
      "status": "HEALTHY"
    },
    {
      "id": "i-047b3c9078EXAMPLE",
      "port": 443,
      "reasonCode": "HealthCheckFailed",
      "status": "UNHEALTHY"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Grup sasaran](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [ListTargets](#) di Referensi AWS CLI Perintah.

## put-auth-policy

Contoh kode berikut menunjukkan cara menggunakan `put-auth-policy`.

### AWS CLI

Untuk membuat kebijakan autentikasi untuk layanan

`put-auth-policy` Contoh berikut memberikan akses ke permintaan dari prinsipal yang diautentikasi yang menggunakan peran IAM yang ditentukan. Sumber daya adalah ARN dari layanan yang dilampirkan kebijakan tersebut.

```
aws vpc-lattice put-auth-policy \
  --resource-identifier svc-0285b53b2eEXAMPLE \
  --policy file://auth-policy.json
```

Isi dari `auth-policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```

    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/my-clients"
      },
      "Action": "vpc-lattice-svcs:Invoke",
      "Resource": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE"
    }
  ]
}

```

### Output:

```

{
  "policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",
,\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:role/my-clients\"},
,\"Action\":\"vpc-lattice-svcs:Invoke\",,\"Resource\":\"arn:aws:vpc-lattice:us-
east-2:123456789012:service/svc-0285b53b2eEXAMPLE\"}]}",
  "state": "Active"
}

```

Untuk informasi selengkapnya, lihat [Kebijakan autentikasi](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [PutAuthPolicy](#) di Referensi AWS CLI Perintah.

## register-targets

Contoh kode berikut menunjukkan cara menggunakan `register-targets`.

### AWS CLI

Untuk mendaftarkan target

`register-targets` Contoh berikut mendaftarkan target yang ditentukan dengan kelompok target yang ditentukan.

```

aws vpc-lattice register-targets \
  --targets id=i-047b3c9078EXAMPLE id=i-07dd579bc5EXAMPLE \
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE

```

## Output:

```
{
  "successful": [
    {
      "id": "i-07dd579bc5EXAMPLE",
      "port": 443
    }
  ],
  "unsuccessful": [
    {
      "failureCode": "UnsupportedTarget",
      "failureMessage": "Instance targets must be in the same VPC as their
target group",
      "id": "i-047b3c9078EXAMPLE",
      "port": 443
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mendaftarkan target](#) di Panduan Pengguna Amazon VPC Lattice.

- Untuk detail API, lihat [RegisterTargets](#) di Referensi AWS CLI Perintah.

## AWS WAF Classic contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS WAF Classic.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **put-logging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-logging-configuration`.

#### AWS CLI

Untuk membuat konfigurasi logging untuk web ACL ARN dengan Arn Kinesis Firehose stream yang ditentukan

`put-logging-configuration` Contoh berikut menampilkan konfigurasi logging untuk WAF dengan CloudFront.

```
aws waf put-logging-configuration \
  --logging-configuration ResourceArn=arn:aws:waf::123456789012:webacl/3bffd3ed-
fa2e-445e-869f-a6a7cf153fd3,LogDestinationConfigs=arn:aws:firehose:us-
east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream,RedactedFields=[]
```

#### Output:

```
{
  "LoggingConfiguration": {
    "ResourceArn": "arn:aws:waf::123456789012:webacl/3bffd3ed-fa2e-445e-869f-
a6a7cf153fd3",
    "LogDestinationConfigs": [
      "arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-
firehose-stream"
    ]
  }
}
```

- Untuk detail API, lihat [PutLoggingConfiguration](#) di Referensi AWS CLI Perintah.

### **update-byte-match-set**

Contoh kode berikut menunjukkan cara menggunakan `update-byte-match-set`.

#### AWS CLI

Untuk memperbarui set kecocokan byte

**update-byte-match-set** Perintah berikut menghapus ByteMatchTuple objek (filter) dalam ByteMatchSet:

```
aws waf update-byte-match-set --byte-match-set-id a123fae4-
b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --
updates
Action="DELETE",ByteMatchTuple={FieldToMatch={Type="HEADER",Data="referer"},TargetString="b
```

Untuk informasi selengkapnya, lihat Bekerja dengan Ketentuan Pencocokan String di panduan pengembang AWS WAF.

- Untuk detail API, lihat [UpdateByteMatchSet](#) di Referensi AWS CLI Perintah.

## update-ip-set

Contoh kode berikut menunjukkan cara menggunakan `update-ip-set`.

### AWS CLI

Untuk memperbarui set IP

**update-ip-set** Perintah berikut memperbarui IPSet dengan alamat IPv4 dan menghapus alamat IPv6:

```
aws waf update-ip-set --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="INSERT",IPSetDescriptor={Type="IPV4",Value="12.34.56.78/16"},Action="DELETE",IPSetD
```

Atau Anda dapat menggunakan file JSON untuk menentukan input. Sebagai contoh:

```
aws waf update-ip-set --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-
token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates file://change.json
```

Dimana isi file JSON adalah:

```
[
{
  "Action": "INSERT",
  "IPSetDescriptor":
  {
    "Type": "IPV4",
```

```
"Value": "12.34.56.78/16"
}
},
{
  "Action": "DELETE",
  "IPSetDescriptor":
  {
    "Type": "IPV6",
    "Value": "1111:0000:0000:0000:0000:0000:0000:0111/128"
  }
}
]
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan IP](#) di panduan pengembang AWS WAF.

- Untuk detail API, lihat [UpdateIpSet](#) di Referensi AWS CLI Perintah.

## update-rule

Contoh kode berikut menunjukkan cara menggunakan `update-rule`.

### AWS CLI

Untuk memperbarui aturan

`update-rule` Perintah berikut menghapus objek Predikat dalam aturan:

```
aws waf update-rule --rule-id a123fae4-b567-8e90-1234-5ab67ac8ca90
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="DELETE",Predicate={Negated=false,Type="ByteMatch",DataId="MyByteMatchSetID"}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Aturan](#) di panduan pengembang AWS WAF.

- Untuk detail API, lihat [UpdateRule](#) di Referensi AWS CLI Perintah.

## update-size-constraint-set

Contoh kode berikut menunjukkan cara menggunakan `update-size-constraint-set`.

### AWS CLI

Untuk memperbarui set batasan ukuran



**update-size-constraint-set** Perintah berikut menghapus SizeConstraint objek (filter) dalam set batasan ukuran:

```
aws waf update-size-constraint-set --size-constraint-set-id a123fae4-
b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --
updates
Action="DELETE",SizeConstraint={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NONE"
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Kendala Ukuran](#) di panduan pengembang AWS WAF.

- Untuk detail API, lihat [UpdateSizeConstraintSet](#) di Referensi AWS CLI Perintah.

## update-sql-injection-match-set

Contoh kode berikut menunjukkan cara menggunakan `update-sql-injection-match-set`.

### AWS CLI

Untuk memperbarui SQL Injection Match Set

**update-sql-injection-match-set** Perintah berikut menghapus `SqlInjectionMatchTuple` objek (filter) dalam set kecocokan injeksi SQL:

```
aws waf update-sql-injection-match-set --sql-injection-
match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --
change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="DELETE",SqlInjectionMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NONE"
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan Injeksi SQL](#) di panduan pengembang AWS WAF.

- Untuk detail API, lihat [UpdateSqlInjectionMatchSet](#) di Referensi AWS CLI Perintah.

## update-web-acl

Contoh kode berikut menunjukkan cara menggunakan `update-web-acl`.

### AWS CLI

Untuk memperbarui ACL web

`update-web-acl` Perintah berikut menghapus `ActivatedRule` objek di WebACL.

```
aws waf update-web-acl -- web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-token
12cs345-67cd-890b-1cd2-c3a4567d89f1 --update action="Delete", = {Prioritas=1, ="WAFrule-1-
Contoh", Action= {type="allow "}, Type=="Reguler"} 'ActivatedRuleRuleId
```

Output:

```
{
  "ChangeToken": "12cs345-67cd-890b-1cd2-c3a4567d89f1"
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Web ACL](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [UpdateWebAcl](#) di Referensi AWS CLI Perintah.

## **update-xss-match-set**

Contoh kode berikut menunjukkan cara menggunakan `update-xss-match-set`.

### AWS CLI

Untuk memperbarui XSS MatchSet

`update-xss-match-set` Perintah berikut menghapus `XssMatchTuple` objek (filter) dalam `XssMatchSet`

```
aws waf update-xss-match-set --xss-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="DELETE",XssMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="URL_D
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan Skrip Lintas Situs](#) di panduan pengembang AWS WAF.

- Untuk detail API, lihat [UpdateXssMatchSet](#) di Referensi AWS CLI Perintah.

## AWS WAF Classic Regional contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS WAF Classic Regional.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-web-acl**

Contoh kode berikut menunjukkan cara menggunakan `associate-web-acl`.

AWS CLI

Untuk mengaitkan ACL web dengan sumber daya

`associate-web-acl` Perintah berikut mengaitkan ACL web, ditentukan oleh, dengan sumber daya `web-acl-id`, yang ditentukan oleh sumber daya `arn`. Sumber daya ARN dapat merujuk ke penyeimbang beban aplikasi atau API Gateway:

```
aws waf-regional associate-web-acl \  
  --web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \  
  --resource-arn 12cs345-67cd-890b-1cd2-c3a4567d89f1
```

Untuk informasi selengkapnya, lihat [Bekerja dengan ACL Web](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [AssociateWebAcl](#) di Referensi AWS CLI Perintah.

### **put-logging-configuration**

Contoh kode berikut menunjukkan cara menggunakan `put-logging-configuration`.

## AWS CLI

Untuk membuat konfigurasi logging untuk web ACL ARN dengan Arn Kinesis Firehose stream yang ditentukan

`put-logging-configuration` Contoh berikut menampilkan konfigurasi logging untuk WAF dengan ALB/ApiGateway di Wilayah. `us-east-1`

```
aws waf-regional put-logging-configuration \
  --logging-configuration ResourceArn=arn:aws:waf-
regional:us-east-1:123456789012:webacl/3bffd3ed-fa2e-445e-869f-
a6a7cf153fd3,LogDestinationConfigs=arn:aws:firehose:us-
east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream,RedactedFields=[] \
  --region us-east-1
```

Output:

```
{
  "LoggingConfiguration": {
    "ResourceArn": "arn:aws:waf-regional:us-east-1:123456789012:webacl/3bffd3ed-
fa2e-445e-869f-a6a7cf153fd3",
    "LogDestinationConfigs": [
      "arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-
firehose-stream"
    ]
  }
}
```

- Untuk detail API, lihat [PutLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## update-byte-match-set

Contoh kode berikut menunjukkan cara menggunakan `update-byte-match-set`.

## AWS CLI

Untuk memperbarui set kecocokan byte

`update-byte-match-set` Perintah berikut menghapus `ByteMatchTuple` objek (filter) dalam file. `ByteMatchSet` Karena `updates` nilainya telah menyematkan tanda kutip ganda, Anda harus mengelilingi nilainya dengan tanda kutip tunggal.

```
aws waf-regional update-byte-match-set \
  --byte-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
'Action="DELETE",ByteMatchTuple={FieldToMatch={Type="HEADER",Data="referer"},TargetString="
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan String](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [UpdateByteMatchSet](#) di Referensi AWS CLI Perintah.

## update-ip-set

Contoh kode berikut menunjukkan cara menggunakan `update-ip-set`.

### AWS CLI

Untuk memperbarui set IP

`update-ip-set` Perintah berikut memperbarui IPSet dengan alamat IPv4 dan menghapus alamat IPv6. Dapatkan nilainya `change-token` dengan menjalankan `get-change-token` perintah. Karena nilai untuk pembaruan menyertakan tanda kutip ganda yang disematkan, Anda harus mengelilingi nilainya dengan tanda kutip tunggal.

```
aws waf update-ip-set \
  --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
'Action="INSERT",IPSetDescriptor={Type="IPV4",Value="12.34.56.78/16"},Action="DELETE",IPSet
```

Atau Anda dapat menggunakan file JSON untuk menentukan input. Sebagai contoh:

```
aws waf-regional update-ip-set \
  --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates file://change.json
```

Isi dari `change.json`

```
[
  {
```

```

    "Action": "INSERT",
    "IPSetDescriptor":
    {
        "Type": "IPV4",
        "Value": "12.34.56.78/16"
    }
},
{
    "Action": "DELETE",
    "IPSetDescriptor":
    {
        "Type": "IPV6",
        "Value": "1111:0000:0000:0000:0000:0000:0000:0111/128"
    }
}
]

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan IP](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [UpdateIpSet](#) di Referensi AWS CLI Perintah.

## update-rule

Contoh kode berikut menunjukkan cara menggunakan `update-rule`.

### AWS CLI

Untuk memperbarui aturan

`update-rule` Perintah berikut menghapus Predicate objek dalam aturan. Karena `updates` nilainya telah menyematkan tanda kutip ganda, Anda harus mengelilingi seluruh nilai dengan tanda kutip tunggal.

```

aws waf-regional update-rule \
  --rule-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
  'Action="DELETE",Predicate={Negated=false,Type="ByteMatch",DataId="MyByteMatchSetID"}'

```

Untuk informasi selengkapnya, lihat [Bekerja dengan Aturan](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [UpdateRule](#) di Referensi AWS CLI Perintah.

## update-size-constraint-set

Contoh kode berikut menunjukkan cara menggunakan `update-size-constraint-set`.

### AWS CLI

Untuk memperbarui set batasan ukuran

`update-size-constraint-set` Perintah berikut menghapus objek `SizeConstraint` (filter) dalam set batasan ukuran. Karena `updates` nilainya berisi tanda kutip ganda yang disematkan, Anda harus mengelilingi seluruh nilai dengan tanda kutip tunggal.

```
aws waf-regional update-size-constraint-set \
  --size-constraint-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
  'Action="DELETE",SizeConstraint={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NON'
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Kendala Ukuran di Panduan Pengembang AWS WAF](#).

- Untuk detail API, lihat [UpdateSizeConstraintSet](#) di Referensi AWS CLI Perintah.

## update-sql-injection-match-set

Contoh kode berikut menunjukkan cara menggunakan `update-sql-injection-match-set`.

### AWS CLI

Untuk memperbarui SQL Injection Match Set

`update-sql-injection-match-set` Perintah berikut menghapus `SqlInjectionMatchTuple` objek (filter) dalam set kecocokan injeksi SQL. Karena `updates` nilainya berisi tanda kutip ganda yang disematkan, Anda harus mengelilingi seluruh nilai dalam tanda kutip tunggal. :

```
aws waf-regional update-sql-injection-match -set -- sql-injection-match-set -id
a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-
c3a4567d89f1 --update 'action="Delete", = {= {type="Query_string "}, ="URL_DECODE"}
'SqlInjectionMatchTupleFieldToMatchTextTransformation
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan Injeksi SQL](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [UpdateSqlInjectionMatchSet](#) di Referensi AWS CLI Perintah.

## update-web-acl

Contoh kode berikut menunjukkan cara menggunakan `update-web-acl`.

### AWS CLI

Untuk memperbarui ACL web

`update-web-acl` Perintah berikut menghapus `ActivatedRule` objek di WebACL. Karena `updates` nilainya berisi tanda kutip ganda yang disematkan, Anda harus mengelilingi seluruh nilai dalam tanda kutip tunggal.

```
aws waf-regional update-web-acl \  
  --web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \  
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \  
  --updates Action="DELETE",ActivatedRule='{Priority=1,RuleId="WAFRule-1-Example",Action={Type="ALLOW"},Type="ALLOW"}'
```

Untuk informasi selengkapnya, lihat [Bekerja dengan ACL Web](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [UpdateWebAcl](#) di Referensi AWS CLI Perintah.

## update-xss-match-set

Contoh kode berikut menunjukkan cara menggunakan `update-xss-match-set`.

### AWS CLI

Untuk memperbarui XSS MatchSet

`update-xss-match-set` Perintah berikut menghapus `XssMatchTuple` objek (filter) dalam `fileXssMatchSet`. Karena `updates` nilainya berisi tanda kutip ganda yang disematkan, Anda harus mengelilingi seluruh nilai dengan tanda kutip tunggal.

```
aws waf-regional update-xss-match-set \  
  --xss-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \  
  --updates '{XssMatchTuple={Filter={Type="SQLI"},Type="SQLI"},Type="SQLI"}'
```



```
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \  
--updates  
'Action="DELETE",XssMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="URL_
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Ketentuan Pencocokan Skrip Lintas Situs](#) di Panduan Pengembang AWS WAF.

- Untuk detail API, lihat [UpdateXssMatchSet](#) di Referensi AWS CLI Perintah.

## AWS WAFV2 contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with AWS WAFV2.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **associate-web-acl**

Contoh kode berikut menunjukkan cara menggunakan `associate-web-acl`.

AWS CLI

Untuk mengaitkan ACL web dengan sumber daya regional AWS

`associate-web-acl` Contoh berikut mengaitkan ACL web tertentu dengan Application Load Balancer.

```
aws wafv2 associate-web-acl \  

```

```
--web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
--resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/waf-cli-alb/1ea17125f8b25a2a \
--region us-west-2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan atau Memutus ACL Web dengan AWS Sumber Daya](#) di WAF AWS , Firewall AWS Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [AssociateWebAcl](#) di Referensi AWS CLI Perintah.

## check-capacity

Contoh kode berikut menunjukkan cara menggunakan `check-capacity`.

### AWS CLI

Untuk mendapatkan kapasitas yang digunakan oleh seperangkat aturan

Berikut ini `check-capacity` mengambil persyaratan kapasitas untuk kumpulan aturan yang berisi pernyataan aturan berbasis laju, dan pernyataan aturan AND yang berisi aturan bersarang.

```
aws wafv2 check-capacity \
--scope REGIONAL \
--rules file://waf-rule-list.json \
--region us-west-2
```

Isi file://waf-rule-list.json:

```
[
  {
    "Name":"basic-rule",
    "Priority":0,
    "Statement":{
      "AndStatement":{
        "Statements":[
          {
            "ByteMatchStatement":{
              "SearchString":"example.com",
              "FieldToMatch":{
                "SingleHeader":{
```

```

        "Name": "host"
      }
    },
    "TextTransformations": [
      {
        "Priority": 0,
        "Type": "LOWERCASE"
      }
    ],
    "PositionalConstraint": "EXACTLY"
  }
},
{
  "GeoMatchStatement": {
    "CountryCodes": [
      "US",
      "IN"
    ]
  }
}
]
}
},
"Action": {
  "Allow": {
  }
},
"VisibilityConfig": {
  "SampledRequestsEnabled": true,
  "CloudWatchMetricsEnabled": true,
  "MetricName": "basic-rule"
}
},
{
  "Name": "rate-rule",
  "Priority": 1,
  "Statement": {
    "RateBasedStatement": {
      "Limit": 1000,
      "AggregateKeyType": "IP"
    }
  },
  "Action": {

```

```

        "Block":{
            }
        },
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
            "MetricName":"rate-rule"
        }
    }
]

```

Output:

```

{
    "Capacity":15
}

```

Untuk informasi selengkapnya, lihat [AWS WAF Web ACL Capacity Units \(WCU\)](#) di AWS WAF, Firewall Manager, dan AWS Shield AWS Advanced Developer Guide.

- Untuk detail API, lihat [CheckCapacity](#) di Referensi AWS CLI Perintah.

## create-ip-set

Contoh kode berikut menunjukkan cara menggunakan `create-ip-set`.

### AWS CLI

Untuk membuat set IP untuk digunakan di ACL web dan grup aturan

`create-ip-set` Perintah berikut membuat set IP dengan spesifikasi rentang alamat tunggal.

```

aws wafv2 create-ip-set \
  --name testip \
  --scope REGIONAL \
  --ip-address-version IPV4 \
  --addresses 198.51.100.0/16

```

Output:

```

{

```

```

    "Summary":{
      "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Description":"",
      "Name":"testip",
      "LockToken":"447e55ac-0000-0000-0000-86b67c17f8b5",
      "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [CreatelpSet](#) di Referensi AWS CLI Perintah.

## create-regex-pattern-set

Contoh kode berikut menunjukkan cara menggunakan `create-regex-pattern-set`.

### AWS CLI

Untuk membuat set pola regex untuk digunakan di ACL web dan grup aturan

`create-regex-pattern-set` Perintah berikut menciptakan pola regex yang ditetapkan dengan dua pola regex yang ditentukan.

```

aws wafv2 create-regex-pattern-set \
  --name regexPatterSet01 \
  --scope REGIONAL \
  --description 'Test web-acl' \
  --regular-expression-list '[{"RegexString": "/[0-9]*/"}, {"RegexString": "/[a-z]*/"}]'

```

### Output:

```

{
  "Summary":{
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/
regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Description":"Test web-acl",
    "Name":"regexPatterSet01",
    "LockToken":"0bc01e21-03c9-4b98-9433-6229cbf1ef1c",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

```
}  
}
```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [CreateRegexPatternSet](#) di Referensi AWS CLI Perintah.

## create-rule-group

Contoh kode berikut menunjukkan cara menggunakan `create-rule-group`.

### AWS CLI

Untuk membuat grup aturan khusus untuk digunakan di ACL web

`create-rule-group` Perintah berikut membuat grup aturan kustom untuk penggunaan regional. Pernyataan aturan untuk grup disediakan dalam file berformat JSON.

```
aws wafv2 create-rule-group \  
  --name "TestRuleGroup" \  
  --scope REGIONAL \  
  --capacity 250 \  
  --rules file://waf-rule.json \  
  --visibility-config  
  SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=TestRuleGroupMetrics  
  \  
  --region us-west-2
```

Isi file: `//waf-rule.json`:

```
[  
  {  
    "Name":"basic-rule",  
    "Priority":0,  
    "Statement":{  
      "AndStatement":{  
        "Statements":[  
          {  
            "ByteMatchStatement":{  
              "SearchString":"example.com",  
              "FieldToMatch":{
```

```

        "SingleHeader":{
            "Name":"host"
        }
    },
    "TextTransformations":[
        {
            "Priority":0,
            "Type":"LOWERCASE"
        }
    ],
    "PositionalConstraint":"EXACTLY"
}
},
{
    "GeoMatchStatement":{
        "CountryCodes":[
            "US",
            "IN"
        ]
    }
}
]
}
},
"Action":{
    "Allow":{

    }
},
"VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"basic-rule"
}
}
]

```

### Output:

```

{
    "Summary":{
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/
TestRuleGroup/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",

```

```

    "Description": "",
    "Name": "TestRuleGroup",
    "LockToken": "7b3bcec2-374e-4c5a-b2b9-563bf47249f0",
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

Untuk informasi selengkapnya, lihat [Mengelola Grup Aturan Anda Sendiri](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [CreateRuleGroup](#) di Referensi AWS CLI Perintah.

## create-web-acl

Contoh kode berikut menunjukkan cara menggunakan `create-web-acl`.

### AWS CLI

Untuk membuat web ACL

`create-web-acl` Perintah berikut membuat ACL web untuk penggunaan regional. Pernyataan aturan untuk ACL web disediakan dalam file berformat JSON.

```

aws wafv2 create-web-acl \
  --name TestWebAcl \
  --scope REGIONAL \
  --default-action Allow={} \
  --visibility-config
SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=TestWebAclMetrics
\
  --rules file://waf-rule.json \
  --region us-west-2

```

Isi file: `//waf-rule.json`:

```

[
  {
    "Name": "basic-rule",
    "Priority": 0,
    "Statement": {
      "AndStatement": {
        "Statements": [
          {

```



```
        "ByteMatchStatement":{
            "SearchString":"example.com",
            "FieldToMatch":{
                "SingleHeader":{
                    "Name":"host"
                }
            },
            "TextTransformations":[
                {
                    "Priority":0,
                    "Type":"LOWERCASE"
                }
            ],
            "PositionalConstraint":"EXACTLY"
        },
        {
            "GeoMatchStatement":{
                "CountryCodes":[
                    "US",
                    "IN"
                ]
            }
        }
    ]
},
"Action":{
    "Allow":{
    }
},
"VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"basic-rule"
}
}
```

Output:

```
{
```

```
"Summary":{
  "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/TestWebAcl/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Description":"","
  "Name":"TestWebAcl",
  "LockToken":"2294b3a1-eb60-4aa0-a86f-a3ae04329de9",
  "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Untuk informasi selengkapnya, lihat [Mengelola dan Menggunakan Daftar Kontrol Akses Web \(Web ACL\)](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [CreateWebAcl](#) di Referensi AWS CLI Perintah.

## delete-ip-set

Contoh kode berikut menunjukkan cara menggunakan `delete-ip-set`.

### AWS CLI

Untuk menghapus set IP

Berikut ini `delete-ip-set` menghapus set IP yang ditentukan. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-ip-sets`, dan token kunci, yang dapat Anda peroleh dari panggilan, `list-ip-sets` dan `get-ip-set`.

```
aws wafv2 delete-ip-set \
  --name test1 \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --lock-token 46851772-db6f-459d-9385-49428812e357
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [DeleteIpSet](#) di Referensi AWS CLI Perintah.

## delete-logging-configuration

Contoh kode berikut menunjukkan cara menggunakan `delete-logging-configuration`.

### AWS CLI

Untuk menonaktifkan pencatatan untuk ACL web

Berikut ini `delete-logging-configuration` menghapus konfigurasi logging dari ACL web yang ditentukan.

```
aws wafv2 delete-logging-configuration \  
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mencatat Informasi Lalu Lintas ACL Web](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [DeleteLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## delete-regex-pattern-set

Contoh kode berikut menunjukkan cara menggunakan `delete-regex-pattern-set`.

### AWS CLI

Untuk menghapus set pola regex

Berikut ini `delete-regex-pattern-set` memperbarui pengaturan untuk set pola regex yang ditentukan. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-regex-pattern-sets`, dan token kunci, yang dapat Anda peroleh dari panggilan `list-regex-pattern-sets` atau panggilan `get-regex-pattern-set`.

```
aws wafv2 delete-regex-pattern-set \  
  --name regexPatterSet01 \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token 0bc01e21-03c9-4b98-9433-6229cbf1ef1c
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [DeleteRegexPatternSet](#) di Referensi AWS CLI Perintah.

## **delete-rule-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-rule-group`.

### AWS CLI

Untuk menghapus grup aturan kustom

Berikut ini `delete-rule-group` menghapus grup aturan kustom yang ditentukan. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-rule-groups`, dan token kunci, yang dapat Anda peroleh dari panggilan `list-rule-groups` atau panggilan `get-rule-group`.

```
aws wafv2 delete-rule-group \
  --name TestRuleGroup \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --lock-token 7b3bcec2-0000-0000-0000-563bf47249f0
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Grup Aturan Anda Sendiri](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DeleteRuleGroup](#) di Referensi AWS CLI Perintah.

## **delete-web-acl**

Contoh kode berikut menunjukkan cara menggunakan `delete-web-acl`.

### AWS CLI

Untuk menghapus ACL web

Berikut ini `delete-web-acl` menghapus ACL web yang ditentukan dari akun Anda. ACL web hanya dapat dihapus jika tidak terkait dengan sumber daya apa pun. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-web-acls`, dan token kunci, yang dapat Anda peroleh dari panggilan `list-web-acls` atau panggilan `get-web-acl`.

```
aws wafv2 delete-web-acl \  
  --name test \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token ebab4ed2-155e-4c9a-9efb-e4c45665b1f5
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola dan Menggunakan Daftar Kontrol Akses Web \(Web ACL\)](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DeleteWebAcl](#) di Referensi AWS CLI Perintah.

## **describe-managed-rule-group**

Contoh kode berikut menunjukkan cara menggunakan `describe-managed-rule-group`.

### AWS CLI

Untuk mengambil deskripsi untuk grup aturan terkelola

Berikut ini `describe-managed-rule-group` mengambil deskripsi untuk grup aturan AWS terkelola.

```
aws wafv2 describe-managed-rule-group \  
  --vendor-name AWS \  
  --name AWSManagedRulesCommonRuleSet \  
  --scope REGIONAL
```

Output:

```
{  
  "Capacity": 700,  
  "Rules": [  
    {  
      "Name": "NoUserAgent_HEADER",  
      "Action": {  
        "Block": {}  
      }  
    },  
    {
```

```
    "Name": "UserAgent_BadBots_HEADER",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "SizeRestrictions_QUERYSTRING",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "SizeRestrictions_Cookie_HEADER",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "SizeRestrictions_BODY",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "SizeRestrictions_URI_PATH",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "EC2Meta-data-SSRF_BODY",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "EC2Meta-data-SSRF_COOKIE",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "EC2Meta-data-SSRF_URI_PATH",
    "Action": {
```

```
        "Block": {}
    }
},
{
    "Name": "EC2MetaDataSSRF_QUERYARGUMENTS",
    "Action": {
        "Block": {}
    }
},
{
    "Name": "GenericLFI_QUERYARGUMENTS",
    "Action": {
        "Block": {}
    }
},
{
    "Name": "GenericLFI_URIPATH",
    "Action": {
        "Block": {}
    }
},
{
    "Name": "GenericLFI_BODY",
    "Action": {
        "Block": {}
    }
},
{
    "Name": "RestrictedExtensions_URIPATH",
    "Action": {
        "Block": {}
    }
},
{
    "Name": "RestrictedExtensions_QUERYARGUMENTS",
    "Action": {
        "Block": {}
    }
},
{
    "Name": "GenericRFI_QUERYARGUMENTS",
    "Action": {
        "Block": {}
    }
}
```

```
    }
  },
  {
    "Name": "GenericRFI_BODY",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "GenericRFI_URIPATH",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "CrossSiteScripting_COOKIE",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "CrossSiteScripting_QUERYARGUMENTS",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "CrossSiteScripting_BODY",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "CrossSiteScripting_URIPATH",
    "Action": {
      "Block": {}
    }
  }
]
}
```

Untuk informasi selengkapnya, lihat [Grup Aturan Terkelola](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.



- Untuk detail API, lihat [DescribeManagedRuleGroup](#) di Referensi AWS CLI Perintah.

## disassociate-web-acl

Contoh kode berikut menunjukkan cara menggunakan `disassociate-web-acl`.

### AWS CLI

Untuk memisahkan ACL web dari sumber daya regional AWS

`disassociate-web-acl` Contoh berikut menghapus asosiasi ACL web yang ada dari Application Load Balancer yang ditentukan.

```
aws wafv2 disassociate-web-acl \
  --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/waf-cli-alb/1ea17125f8b25a2a \
  --region us-west-2
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengaitkan atau Memutus ACL Web dengan AWS Sumber Daya](#) di WAF AWS , Firewall AWS Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [DisassociateWebAcl](#) di Referensi AWS CLI Perintah.

## get-ip-set

Contoh kode berikut menunjukkan cara menggunakan `get-ip-set`.

### AWS CLI

Untuk mengambil set IP tertentu

Berikut ini `get-ip-set` mengambil IP set dengan nama yang ditentukan, ruang lingkup, dan ID. Anda bisa mendapatkan ID untuk set IP dari perintah `create-ip-set` dan `list-ip-sets`.

```
aws wafv2 get-ip-set \
  --name testip \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
  "IPSet":{
    "Description":"",
    "Name":"testip",
    "IPAddressVersion":"IPv4",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE1111",
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE1111",
    "Addresses":[
      "192.0.2.0/16"
    ]
  },
  "LockToken":"447e55ac-2396-4c6d-b9f9-86b67c17f8b5"
}
```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetIpSet](#) di Referensi AWS CLI Perintah.

## get-logging-configuration

Contoh kode berikut menunjukkan cara menggunakan `get-logging-configuration`.

### AWS CLI

Untuk mengambil konfigurasi logging untuk ACL web

Berikut ini `get-logging-configuration` mengambil konfigurasi logging untuk ACL web tertentu.

```
aws wafv2 get-logging-configuration \
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 \
  --region us-west-2
```

### Output:

```
{
  "LoggingConfiguration":{
    "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
```

```

    "RedactedFields":[
      {
        "Method":{
          }
        }
      ],
      "LogDestinationConfigs":[
        "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-
        custom-transformation"
      ]
    ]
  }
}

```

Untuk informasi selengkapnya, lihat [Mencatat Informasi Lalu Lintas ACL Web](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## get-rate-based-statement-managed-keys

Contoh kode berikut menunjukkan cara menggunakan `get-rate-based-statement-managed-keys`.

### AWS CLI

Untuk mengambil daftar alamat IP yang diblokir oleh aturan berbasis tarif

Berikut ini `get-rate-based-statement-managed-keys` mengambil alamat IP yang saat ini diblokir oleh aturan berbasis tarif yang digunakan untuk aplikasi regional.

```

aws wafv2 get-rate-based-statement-managed-keys \
  --scope REGIONAL \
  --web-acl-name testwebacl2 \
  --web-acl-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --rule-name ratebasedtest

```

### Output:

```

{
  "ManagedKeysIPv4":{
    "IPAddressVersion":"IPV4",
    "Addresses":[

```

```

        "198.51.100.0/32"
      ]
    },
    "ManagedKeysIPV6":{
      "IPAddressVersion":"IPV6",
      "Addresses":[
        ]
      }
    }
  }
}

```

Untuk informasi selengkapnya, lihat [Pernyataan Aturan Berbasis Tarif](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetRateBasedStatementManagedKeys](#) di Referensi AWS CLI Perintah.

## get-regex-pattern-set

Contoh kode berikut menunjukkan cara menggunakan `get-regex-pattern-set`.

### AWS CLI

Untuk mengambil set pola regex tertentu

Berikut ini `get-regex-pattern-set` mengambil pola regex yang ditetapkan dengan nama, cakupan, wilayah, dan ID yang ditentukan. Anda bisa mendapatkan ID untuk pola regex yang ditetapkan dari perintah `create-regex-pattern-set` dan `list-regex-pattern-sets`

```

aws wafv2 get-regex-pattern-set \
  --name regexPatterSet01 \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --region us-west-2

```

Output:

```

{
  "RegexPatternSet":{
    "Description":"Test web-acl",
    "RegularExpressionList":[
      {
        "RegexString":"/[0-9]*/"
      },
    ],
  },
}

```

```

        {
            "RegexString":"/[a-z]*/"
        }
    ],
    "Name":"regexPatterSet01",
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/
regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
},
"LockToken":"c8abf33f-b6fc-46ae-846e-42f994d57b29"
}

```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetRegexPatternSet](#) di Referensi AWS CLI Perintah.

## get-rule-group

Contoh kode berikut menunjukkan cara menggunakan `get-rule-group`.

### AWS CLI

Untuk mengambil grup aturan khusus tertentu

Berikut ini `get-rule-group` mengambil grup aturan kustom dengan nama, cakupan, dan ID yang ditentukan. Anda bisa mendapatkan ID untuk grup aturan dari perintah `create-rule-group` dan `list-rule-groups`.

```

aws wafv2 get-rule-group \
  --name ff \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Output:

```

{
  "RuleGroup":{
    "Capacity":1,
    "Description":"",
    "Rules":[
      {
        "Priority":0,

```

```
    "Action":{
      "Block":{

      }
    },
    "VisibilityConfig":{
      "SampledRequestsEnabled":true,
      "CloudWatchMetricsEnabled":true,
      "MetricName":"jj"
    },
    "Name":"jj",
    "Statement":{
      "SizeConstraintStatement":{
        "ComparisonOperator":"LE",
        "TextTransformations":[
          {
            "Priority":0,
            "Type":"NONE"
          }
        ],
        "FieldToMatch":{
          "UriPath":{

          }
        },
        "Size":7
      }
    }
  ],
  "VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"ff"
  },
  "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/ff/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Name":"ff"
},
"LockToken":"485458c9-1830-4234-af31-ec4d52ced1b3"
}
```

Untuk informasi selengkapnya, lihat [Mengelola Grup Aturan Anda Sendiri](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [GetRuleGroup](#) di Referensi AWS CLI Perintah.

## get-sampled-requests

Contoh kode berikut menunjukkan cara menggunakan `get-sampled-requests`.

### AWS CLI

Untuk mengambil sampel permintaan web untuk ACL web

Berikut ini `get-sampled-requests` mengambil sampel permintaan web untuk ACL web tertentu, metrik aturan, dan kerangka waktu.

```
aws wafv2 get-sampled-requests \
  --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --rule-metric-name AWS-AWSManagedRulesSQLiRuleSet \
  --scope=REGIONAL \
  --time-window StartTime=2020-02-12T20:00Z,EndTime=2020-02-12T21:10Z \
  --max-items 100
```

### Output:

```
{
  "TimeWindow": {
    "EndTime": 1581541800.0,
    "StartTime": 1581537600.0
  },
  "SampledRequests": [
    {
      "Action": "BLOCK",
      "Timestamp": 1581541799.564,
      "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
      "Request": {
        "Country": "US",
        "URI": "/",
        "Headers": [
          {
            "Name": "Host",
            "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
```

```

        },
        {
            "Name": "Content-Length",
            "Value": "7456"
        },
        {
            "Name": "User-Agent",
            "Value": "curl/7.53.1"
        },
        {
            "Name": "Accept",
            "Value": "/"
        },
        {
            "Name": "Content-Type",
            "Value": "application/x-www-form-urlencoded"
        }
    ],
    "ClientIP": "198.51.100.08",
    "Method": "POST",
    "HTTPVersion": "HTTP/1.1"
},
"Weight": 1
},
{
    "Action": "BLOCK",
    "Timestamp": 1581541799.988,
    "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
    "Request": {
        "Country": "US",
        "URI": "/",
        "Headers": [
            {
                "Name": "Host",
                "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
            },
            {
                "Name": "Content-Length",
                "Value": "7456"
            },
            {
                "Name": "User-Agent",
                "Value": "curl/7.53.1"
            }
        ]
    }
},

```



```
        {
            "Name": "Accept",
            "Value": "/"
        },
        {
            "Name": "Content-Type",
            "Value": "application/x-www-form-urlencoded"
        }
    ],
    "ClientIP": "198.51.100.08",
    "Method": "POST",
    "HTTPVersion": "HTTP/1.1"
},
"Weight": 3
},
{
    "Action": "BLOCK",
    "Timestamp": 1581541799.846,
    "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
    "Request": {
        "Country": "US",
        "URI": "/",
        "Headers": [
            {
                "Name": "Host",
                "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
            },
            {
                "Name": "Content-Length",
                "Value": "7456"
            },
            {
                "Name": "User-Agent",
                "Value": "curl/7.53.1"
            },
            {
                "Name": "Accept",
                "Value": "/"
            },
            {
                "Name": "Content-Type",
                "Value": "application/x-www-form-urlencoded"
            }
        ]
    }
},
],
```

```
        "ClientIP": "198.51.100.08",
        "Method": "POST",
        "HTTPVersion": "HTTP/1.1"
    },
    "Weight": 1
},
{
    "Action": "BLOCK",
    "Timestamp": 1581541799.4,
    "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
    "Request": {
        "Country": "US",
        "URI": "/",
        "Headers": [
            {
                "Name": "Host",
                "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
            },
            {
                "Name": "Content-Length",
                "Value": "7456"
            },
            {
                "Name": "User-Agent",
                "Value": "curl/7.53.1"
            },
            {
                "Name": "Accept",
                "Value": "/"
            },
            {
                "Name": "Content-Type",
                "Value": "application/x-www-form-urlencoded"
            }
        ],
        "ClientIP": "198.51.100.08",
        "Method": "POST",
        "HTTPVersion": "HTTP/1.1"
    },
    "Weight": 1
}
],
"PopulationSize": 4
```

```
}
```

Untuk informasi selengkapnya, lihat [Melihat Contoh Permintaan Web](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [GetSampledRequests](#) di Referensi AWS CLI Perintah.

## get-web-acl-for-resource

Contoh kode berikut menunjukkan cara menggunakan `get-web-acl-for-resource`.

### AWS CLI

Untuk mengambil ACL web yang terkait dengan sumber daya AWS

Berikut ini `get-web-acl-for-resource` mengambil JSON untuk ACL web yang terkait dengan sumber daya yang ditentukan.

```
aws wafv2 get-web-acl-for-resource \
  --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
  app/waf-cli-alb/1ea17125f8b25a2a
```

Output:

```
{
  "WebACL":{
    "Capacity":3,
    "Description":"",
    "Rules":[
      {
        "Priority":1,
        "Action":{
          "Block":{

          }
        },
        "VisibilityConfig":{
          "SampledRequestsEnabled":true,
          "CloudWatchMetricsEnabled":true,
          "MetricName":"testrule01"
        },
        "Name":"testrule01",
        "Statement":{
```

```

    "AndStatement":{
      "Statements":[
        {
          "ByteMatchStatement":{
            "PositionalConstraint":"EXACTLY",
            "TextTransformations":[
              {
                "Priority":0,
                "Type":"NONE"
              }
            ],
            "SearchString":"dGVzdHN0cm1uZw==",
            "FieldToMatch":{
              "UriPath":{

              }
            }
          }
        },
        {
          "SizeConstraintStatement":{
            "ComparisonOperator":"EQ",
            "TextTransformations":[
              {
                "Priority":0,
                "Type":"NONE"
              }
            ],
            "FieldToMatch":{
              "QueryString":{

              }
            },
            "Size":0
          }
        }
      ]
    }
  ],
  "VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,

```

```

        "MetricName":"test01"
    },
    "DefaultAction":{
        "Allow":{

        }
    },
    "Id":"9a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  ",
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test01/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  ",
    "Name":"test01"
}
}

```

Untuk informasi selengkapnya, lihat [Mengaitkan atau Memutus ACL Web dengan AWS Sumber Daya](#) di WAF AWS , Firewall AWS Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [GetWebAclForResource](#) di Referensi AWS CLI Perintah.

## get-web-acl

Contoh kode berikut menunjukkan cara menggunakan `get-web-acl`.

### AWS CLI

Untuk mengambil ACL web

Berikut ini `get-web-acl` mengambil ACL web dengan nama, ruang lingkup, dan ID yang ditentukan. Anda bisa mendapatkan ID untuk ACL web dari perintah `create-web-acl` dan `list-web-acls`.

```

aws wafv2 get-web-acl \
  --name test01 \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Output:

```

{
  "WebACL":{
    "Capacity":3,
    "Description":"",
    "Rules":[

```

```
{
  "Priority":1,
  "Action":{
    "Block":{

    }
  },
  "VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"testrule01"
  },
  "Name":"testrule01",
  "Statement":{
    "AndStatement":{
      "Statements":[
        {
          "ByteMatchStatement":{
            "PositionalConstraint":"EXACTLY",
            "TextTransformations":[
              {
                "Priority":0,
                "Type":"NONE"
              }
            ],
            "SearchString":"dGVzdHN0cmVuZw==",
            "FieldToMatch":{
              "UriPath":{

              }
            }
          }
        },
        {
          "SizeConstraintStatement":{
            "ComparisonOperator":"EQ",
            "TextTransformations":[
              {
                "Priority":0,
                "Type":"NONE"
              }
            ],
            "FieldToMatch":{
              "QueryString":{

              }
            }
          }
        }
      ]
    }
  }
}
```

```

    }
  },
  "Size":0
}
]
}
}
},
],
"VisibilityConfig":{
  "SampledRequestsEnabled":true,
  "CloudWatchMetricsEnabled":true,
  "MetricName":"test01"
},
"DefaultAction":{
  "Allow":{

  }
},
"Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test01/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"Name":"test01"
},
"LockToken":"e3db7e2c-d58b-4ee6-8346-6aec5511c6fb"
}

```

Untuk informasi selengkapnya, lihat [Mengelola dan Menggunakan Daftar Kontrol Akses Web \(Web ACL\)](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [GetWebAcl](#) di Referensi AWS CLI Perintah.

## list-available-managed-rule-groups

Contoh kode berikut menunjukkan cara menggunakan `list-available-managed-rule-groups`.

### AWS CLI

Untuk mengambil grup aturan terkelola

Berikut ini `list-available-managed-rule-groups` menampilkan daftar semua grup aturan terkelola yang saat ini tersedia untuk digunakan di ACL web Anda.

```
aws wafv2 list-available-managed-rule-groups \  
--scope REGIONAL
```

Output:

```
{  
  "ManagedRuleGroups": [  
    {  
      "VendorName": "AWS",  
      "Name": "AWSManagedRulesCommonRuleSet",  
      "Description": "Contains rules that are generally applicable to web  
applications. This provides protection against exploitation of a wide range of  
vulnerabilities, including those described in OWASP publications and common Common  
Vulnerabilities and Exposures (CVE)."  
    },  
    {  
      "VendorName": "AWS",  
      "Name": "AWSManagedRulesAdminProtectionRuleSet",  
      "Description": "Contains rules that allow you to block external access  
to exposed admin pages. This may be useful if you are running third-party software  
or would like to reduce the risk of a malicious actor gaining administrative access  
to your application."  
    },  
    {  
      "VendorName": "AWS",  
      "Name": "AWSManagedRulesKnownBadInputsRuleSet",  
      "Description": "Contains rules that allow you to block request patterns  
that are known to be invalid and are associated with exploitation or discovery of  
vulnerabilities. This can help reduce the risk of a malicious actor discovering a  
vulnerable application."  
    },  
    {  
      "VendorName": "AWS",  
      "Name": "AWSManagedRulesSQLiRuleSet",  
      "Description": "Contains rules that allow you to block request patterns  
associated with exploitation of SQL databases, like SQL injection attacks. This can  
help prevent remote injection of unauthorized queries."  
    },  
    {  
      "VendorName": "AWS",
```



```
    "Name": "AWSManagedRulesLinuxRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploitation of vulnerabilities specific to Linux, including LFI attacks. This
can help prevent attacks that expose file contents or execute code for which the
attacker should not have had access."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesUnixRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to POSIX/POSIX-like OS, including LFI
attacks. This can help prevent attacks that expose file contents or execute code
for which access should not been allowed."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesWindowsRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to Windows, (e.g., PowerShell commands).
This can help prevent exploits that allow attacker to run unauthorized commands or
execute malicious code."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesPHPRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to the use of the PHP, including injection
of unsafe PHP functions. This can help prevent exploits that allow an attacker to
remotely execute code or commands."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesWordPressRuleSet",
    "Description": "The WordPress Applications group contains rules that
block request patterns associated with the exploitation of vulnerabilities specific
to WordPress sites."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesAmazonIpReputationList",
    "Description": "This group contains rules that are based on Amazon
threat intelligence. This is useful if you would like to block sources associated
with bots or other threats."
  }
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Grup Aturan Terkelola](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [ListAvailableManagedRuleGroups](#) di Referensi AWS CLI Perintah.

## list-ip-sets

Contoh kode berikut menunjukkan cara menggunakan `list-ip-sets`.

### AWS CLI

Untuk mengambil daftar set IP

Berikut ini `list-ip-sets` mengambil semua set IP untuk akun yang memiliki cakupan regional.

```
aws wafv2 list-ip-sets \
  --scope REGIONAL
```

Output:

```
{
  "IPSets": [
    {
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Description": "",
      "Name": "testip",
      "LockToken": "0674c84b-0304-47fe-8728-c6bff46af8fc",
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
  ],
  "NextMarker": "testip"
}
```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [ListIpSets](#) di Referensi AWS CLI Perintah.

## list-logging-configurations

Contoh kode berikut menunjukkan cara menggunakan `list-logging-configurations`.

### AWS CLI

Untuk mengambil daftar semua konfigurasi logging untuk suatu wilayah

Berikut ini `list-logging-configurations` mengambil semua konfigurasi logging untuk ACL web yang dicakup untuk penggunaan regional di wilayah tersebut. `us-west-2`

```
aws wafv2 list-logging-configurations \
  --scope REGIONAL \
  --region us-west-2
```

### Output:

```
{
  "LoggingConfigurations": [
    {
      "ResourceArn": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
test-2/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "RedactedFields": [
        {
          "QueryString": {
          }
        }
      ],
      "LogDestinationConfigs": [
        "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-
logs-test"
      ]
    },
    {
      "ResourceArn": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
test/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "RedactedFields": [
        {
          "Method": {
          }
        }
      ]
    }
  ]
}
```

```

    ],
    "LogDestinationConfigs":[
        "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-custom-transformation"
    ]
}
]
}

```

Untuk informasi selengkapnya, lihat [Mencatat Informasi Lalu Lintas ACL Web](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [ListLoggingConfigurations](#) di Referensi AWS CLI Perintah.

## list-regex-pattern-sets

Contoh kode berikut menunjukkan cara menggunakan `list-regex-pattern-sets`.

### AWS CLI

Untuk mengambil daftar set pola regex

Berikut ini `list-regex-pattern-sets` mengambil semua set pola regex untuk akun yang didefinisikan di wilayah. `us-west-2`

```

aws wafv2 list-regex-pattern-sets \
--scope REGIONAL \
--region us-west-2

```

Output:

```

{
  "NextMarker":"regexPatterSet01",
  "RegexPatternSets":[
    {
      "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Description":"Test web-acl",
      "Name":"regexPatterSet01",
      "LockToken":"f17743f7-0000-0000-0000-19a8b93bfb01",
      "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
  ]
}

```

```
]
}
```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [ListRegexPatternSets](#) di Referensi AWS CLI Perintah.

## **list-resources-for-web-acl**

Contoh kode berikut menunjukkan cara menggunakan `list-resources-for-web-acl`.

### AWS CLI

Untuk mengambil sumber daya yang terkait dengan ACL web

Berikut ini `list-resources-for-web-acl` mengambil sumber daya API API Gateway REST API yang saat ini dikaitkan dengan ACL web tertentu di wilayah tersebut. `us-west-2`

```
aws wafv2 list-resources-for-web-acl \
  --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/TestWebAcl/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --resource-type API_GATEWAY \
  --region us-west-2
```

Output:

```
{
  "ResourceArns": [
    "arn:aws:apigateway:us-west-2::/restapis/EXAMPLE111/stages/testing"
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengaitkan atau Memutus ACL Web dengan AWS Sumber Daya](#) di WAF AWS, Firewall AWS Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [ListResourcesForWebAcl](#) di Referensi AWS CLI Perintah.

## **list-rule-groups**

Contoh kode berikut menunjukkan cara menggunakan `list-rule-groups`.

## AWS CLI

Untuk mengambil daftar grup aturan kustom

Berikut ini `list-rule-groups` mengambil semua grup aturan kustom yang ditentukan untuk akun untuk lingkup dan lokasi wilayah yang ditentukan.

```
aws wafv2 list-rule-groups \  
  --scope REGIONAL \  
  --region us-west-2
```

Output:

```
{  
  "RuleGroups":[  
    {  
      "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/  
TestRuleGroup/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Description":"","  
      "Name":"TestRuleGroup",  
      "LockToken":"1eb5ec48-0000-0000-0000-ee9b906c541e",  
      "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
    },  
    {  
      "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/test/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Description":"","  
      "Name":"test",  
      "LockToken":"b0f4583e-998b-4880-9069-3fbe45738b43",  
      "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"  
    }  
  ],  
  "NextMarker":"test"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Grup Aturan Anda Sendiri](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [ListRuleGroups](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

## AWS CLI

Untuk mengambil semua tag untuk sumber daya AWS WAF

Berikut ini `list-tags-for-resource` mengambil daftar semua kunci tag, pasangan nilai untuk ACL web tertentu.

```
aws wafv2 list-tags-for-resource \
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/testwebacl/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
  "NextMarker": "",
  "TagInfoForResource": {
    "ResourceARN": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
testwebacl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "TagList": [
      ]
    }
  }
}
```

Untuk informasi selengkapnya, lihat [Memulai AWS WAF di AWS WAF](#), AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.

## **list-web-acls**

Contoh kode berikut menunjukkan cara menggunakan `list-web-acls`.

## AWS CLI

Untuk mengambil ACL web untuk cakupan

Berikut ini `list-web-acls` mengambil semua ACL web yang didefinisikan untuk akun untuk cakupan yang ditentukan.

```
aws wafv2 list-web-acls \
```

```
--scope REGIONAL
```

### Output:

```
{
  "NextMarker": "Testt",
  "WebACLs": [
    {
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/Testt/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Description": "sssss",
      "Name": "Testt",
      "LockToken": "7f36cb30-74ef-4cff-8cd4-a77e1aba1746",
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola dan Menggunakan Daftar Kontrol Akses Web \(Web ACL\)](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [ListWebAcls](#) di Referensi AWS CLI Perintah.

## put-logging-configuration

Contoh kode berikut menunjukkan cara menggunakan `put-logging-configuration`.

### AWS CLI

Untuk menambahkan konfigurasi logging ke ACL web

Berikut ini `put-logging-configuration` menambahkan konfigurasi `aws-waf-logs-custom-transformation` logging Amazon Kinesis Data Firehose ke ACL web yang ditentukan, tanpa bidang yang disunting dari log.

```
aws wafv2 put-logging-configuration \
  --logging-configuration ResourceArn=arn:aws:wafv2:us-
west-2:123456789012:regional/webacl/test-cli/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111,LogDestinationConfigs=arn:aws:firehose:us-
west-2:123456789012:deliverystream/aws-waf-logs-custom-transformation \
```



```
--region us-west-2
```

### Output:

```
{
  "LoggingConfiguration":{
    "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-
cli/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "LogDestinationConfigs":[
      "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-
custom-transformation"
    ]
  }
}
```

Untuk informasi selengkapnya, lihat [Mencatat Informasi Lalu Lintas ACL Web](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [PutLoggingConfiguration](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menambahkan tag ke sumber daya AWS WAF

`tag-resource` Contoh berikut menambahkan tag dengan kunci Name dan nilai diatur AWSWAF ke ACL web tertentu.

```
aws wafv2 tag-resource \
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
apiGatewayWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --tags Key=Name,Value=AWSWAF
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai AWS WAF di AWS WAF](#), AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## untag-resource

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag dari sumber daya AWS WAF

`untag-resource` Contoh berikut menghapus tag dengan kunci `KeyName` dari ACL web yang ditentukan.

```
aws wafv2 untag-resource \  
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/  
  apiGatewayWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --tag-keys "KeyName"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulai AWS WAF di AWS WAF](#), AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## update-ip-set

Contoh kode berikut menunjukkan cara menggunakan `update-ip-set`.

### AWS CLI

Untuk memodifikasi pengaturan untuk set IP yang ada

Berikut ini `update-ip-set` memperbarui pengaturan untuk set IP yang ditentukan. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-ip-sets`, dan token kunci yang dapat Anda peroleh dari panggilan, `list-ip-sets` dan `get-ip-set`. Panggilan ini juga mengembalikan token kunci yang dapat Anda gunakan untuk pembaruan berikutnya.

```
aws wafv2 update-ip-set \  
  --name testip \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --addresses 198.51.100.0/16 \  
  --lock-token 447e55ac-2396-4c6d-b9f9-86b67c17f8b5
```

**Output:**

```
{
  "NextLockToken": "0674c84b-0304-47fe-8728-c6bff46af8fc"
}
```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [UpdateIpSet](#) di Referensi AWS CLI Perintah.

**update-regex-pattern-set**

Contoh kode berikut menunjukkan cara menggunakan `update-regex-pattern-set`.

**AWS CLI**

Untuk mengubah pengaturan untuk set pola regex yang ada

Berikut ini `update-regex-pattern-set` memperbarui pengaturan untuk set pola regex yang ditentukan. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-regex-pattern-sets`, dan token kunci yang dapat Anda peroleh dari panggilan, `list-regex-pattern-sets` dan `get-regex-pattern-set`. Panggilan ini juga mengembalikan token kunci yang dapat Anda gunakan untuk pembaruan berikutnya.

```
aws wafv2 update-regex-pattern-set \
  --name ExampleRegex \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --regular-expression-list RegexString="^.+ $" \
  --lock-token ed207e9c-82e9-4a77-aadd-81e6173ab7eb
```

**Output:**

```
{
  "NextLockToken": "12ebc73e-fa68-417d-a9b8-2bdd761a4fa5"
}
```

Untuk informasi selengkapnya, lihat [Set IP dan Set Pola Regex](#) di AWS WAF, AWS Firewall Manager, dan AWS Shield Advanced Developer Guide.

- Untuk detail API, lihat [UpdateRegexPatternSet](#) di Referensi AWS CLI Perintah.

## update-rule-group

Contoh kode berikut menunjukkan cara menggunakan `update-rule-group`.

### AWS CLI

Untuk memperbarui grup aturan kustom

Berikut ini `update-rule-group` mengubah konfigurasi visibilitas untuk grup aturan kustom yang ada. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-rule-groups`, dan token kunci yang dapat Anda peroleh dari panggilan, `list-rule-groups` dan `get-rule-group`. Panggilan ini juga mengembalikan token kunci yang dapat Anda gunakan untuk pembaruan berikutnya.

```
aws wafv2 update-rule-group \  
  --name TestRuleGroup \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token 7b3bcec2-0000-0000-0000-563bf47249f0 \  
  --visibility-config  
  SampledRequestsEnabled=false,CloudWatchMetricsEnabled=false,MetricName=TestMetricsForRuleGr  
  \  
  --region us-west-2
```

Output:

```
{  
  "NextLockToken": "1eb5ec48-0000-0000-0000-ee9b906c541e"  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Grup Aturan Anda Sendiri](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [UpdateRuleGroup](#) di Referensi AWS CLI Perintah.

## update-web-acl

Contoh kode berikut menunjukkan cara menggunakan `update-web-acl`.

### AWS CLI

Untuk memperbarui ACL web

Berikut ini `update-web-acl` mengubah pengaturan untuk ACL web yang ada. Panggilan ini memerlukan ID, yang dapat Anda peroleh dari panggilan `list-web-acls`, dan token kunci dan pengaturan lainnya, yang dapat Anda peroleh dari panggilan tersebut `get-web-acl`. Panggilan ini juga mengembalikan token kunci yang dapat Anda gunakan untuk pembaruan berikutnya.

```
aws wafv2 update-web-acl \
  --name TestWebAcl \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --lock-token 2294b3a1-0000-0000-0000-a3ae04329de9 \
  --default-action Block={} \
  --visibility-config
SampledRequestsEnabled=false,CloudWatchMetricsEnabled=false,MetricName=NewMetricTestWebAcl
\
  --rules file://waf-rule.json \
  --region us-west-2
```

Output:

```
{
  "NextLockToken": "714a0cfb-0000-0000-0000-2959c8b9a684"
}
```

Untuk informasi selengkapnya, lihat [Mengelola dan Menggunakan Daftar Kontrol Akses Web \(Web ACL\)](#) di AWS WAF, AWS Firewall Manager, dan Panduan Pengembang Lanjutan AWS Shield.

- Untuk detail API, lihat [UpdateWebAcl](#) di Referensi AWS CLI Perintah.

## WorkDocs Contoh Amazon menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with Amazon WorkDocs.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **abort-document-version-upload**

Contoh kode berikut menunjukkan cara menggunakan `abort-document-version-upload`.

AWS CLI

Untuk menghentikan unggahan versi dokumen

Contoh ini menghentikan unggahan versi dokumen yang dimulai sebelumnya.

Perintah:

```
aws workdocs abort-document-version-upload --document-id
feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2 --version-id
1536773972914-ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dccccd417da9313
```

Output:

```
None
```

- Untuk detail API, lihat [AbortDocumentVersionUpload](#) di Referensi AWS CLI Perintah.

### **activate-user**

Contoh kode berikut menunjukkan cara menggunakan `activate-user`.

AWS CLI

Untuk mengaktifkan pengguna

Contoh ini mengaktifkan pengguna yang tidak aktif.

Perintah:

```
aws workdocs activate-user --user-id
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Output:

```
{
  "User": {
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Username": "exampleUser",
    "EmailAddress": "exampleUser@site.awsapps.com",
    "GivenName": "Example",
    "Surname": "User",
    "OrganizationId": "d-926726012c",
    "RootFolderId":
"75f67c183aa1217409ac87576a45c03a5df5e6d8c51c35c01669970538e86cd0",
    "RecycleBinFolderId":
"642b7dd3e60b14204534f3df7b1959e01b5d170f8c2707f410e40a8149120a57",
    "Status": "ACTIVE",
    "Type": "MINIMALUSER",
    "CreatedTimestamp": 1521226107.747,
    "ModifiedTimestamp": 1525297406.462,
    "Storage": {
      "StorageUtilizedInBytes": 0,
      "StorageRule": {
        "StorageAllocatedInBytes": 0,
        "StorageType": "QUOTA"
      }
    }
  }
}
```

- Untuk detail API, lihat [ActivateUser](#) di Referensi AWS CLI Perintah.

## add-resource-permissions

Contoh kode berikut menunjukkan cara menggunakan `add-resource-permissions`.

### AWS CLI

Untuk menambahkan izin untuk sumber daya

Contoh ini menambahkan izin ke sumber daya untuk prinsipal tertentu.

**Perintah:**

```
aws workdocs add-resource-permissions --resource-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --principals
Id=anonymous,Type=ANONYMOUS,Role=VIEWER
```

**Output:**

```
{
  "ShareResults": [
    {
      "PrincipalId": "anonymous",
      "Role": "VIEWER",
      "Status": "SUCCESS",
      "ShareId":
"d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
      "StatusMessage": ""
    }
  ]
}
```

- Untuk detail API, lihat [AddResourcePermissions](#) di Referensi AWS CLI Perintah.

**create-comment**

Contoh kode berikut menunjukkan cara menggunakan `create-comment`.

**AWS CLI**

Untuk menambahkan komentar baru

Contoh ini menambahkan komentar baru ke versi dokumen yang ditentukan.

**Perintah:**

```
aws workdocs create-comment --document-id
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-id
1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
text "This is a comment."
```

**Output:**



```
{
  "Comment": {
    "CommentId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
    "ThreadId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
    "Text": "This is a comment.",
    "Contributor": {
      "Id": "arn:aws:iam::123456789123:user/exampleUser",
      "Username": "exampleUser",
      "GivenName": "Example",
      "Surname": "User",
      "Status": "ACTIVE"
    },
    "CreatedTimestamp": 1534799058.197,
    "Status": "PUBLISHED",
    "Visibility": "PUBLIC"
  }
}
```

- Untuk detail API, lihat [CreateComment](#) di Referensi AWS CLI Perintah.

## create-custom-metadata

Contoh kode berikut menunjukkan cara menggunakan `create-custom-metadata`.

### AWS CLI

Untuk membuat metadata kustom

Contoh ini membuat metadata kustom untuk dokumen yang ditentukan.

Perintah:

```
aws workdocs create-custom-metadata --resource-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --custom-metadata
KeyName1=example,KeyName2=example2
```

Output:

```
None
```

- Untuk detail API, lihat [CreateCustomMetadata](#) di Referensi AWS CLI Perintah.

## create-folder

Contoh kode berikut menunjukkan cara menggunakan `create-folder`.

### AWS CLI

Untuk membuat folder

Contoh ini membuat folder.

Perintah:

```
aws workdocs create-folder --name documents --parent-folder-id
1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Output:

```
{
  "Metadata": {
    "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
    "Name": "documents",
    "CreatorId": "S-1-1-11-111111111-222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1534450467.622,
    "ModifiedTimestamp": 1534450467.622,
    "ResourceState": "ACTIVE",
    "Signature": "",
    "Size": 0,
    "LatestVersionSize": 0
  }
}
```

- Untuk detail API, lihat [CreateFolder](#) di Referensi AWS CLI Perintah.

## create-labels

Contoh kode berikut menunjukkan cara menggunakan `create-labels`.

## AWS CLI

Untuk membuat label

Contoh ini membuat serangkaian label untuk dokumen.

Perintah:

```
aws workdocs create-labels --resource-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --labels
"documents" "examples" "my_documents"
```

Output:

```
None
```

- Untuk detail API, lihat [CreateLabels](#) di Referensi AWS CLI Perintah.

## create-notification-subscription

Contoh kode berikut menunjukkan cara menggunakan `create-notification-subscription`.

## AWS CLI

Untuk membuat langganan notifikasi

`create-notification-subscription` Contoh berikut mengonfigurasi langganan notifikasi untuk WorkDocs organisasi Amazon yang ditentukan.

```
aws workdocs create-notification-subscription \
  --organization-id d-123456789c \
  --protocol HTTPS \
  --subscription-type ALL \
  --notification-endpoint "https://example.com/example"
```

Output:

```
{
  "Subscription": {
    "SubscriptionId": "123ab4c5-678d-901e-f23g-45h6789j0123",
    "EndPoint": "https://example.com/example",
```

```
    "Protocol": "HTTPS"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan](#) di Panduan WorkDocs Pengembang Amazon.

- Untuk detail API, lihat [CreateNotificationSubscription](#) di Referensi AWS CLI Perintah.

## create-user

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

### AWS CLI

Untuk membuat pengguna baru

Contoh ini membuat pengguna baru di direktori Simple AD atau Microsoft AD.

Perintah:

```
aws workdocs create-user --organization-id d-926726012c --username exampleUser2  
--email-address exampleUser2@site.awsapps.com --given-name example2Name --surname  
example2Surname --password examplePa$$w0rd
```

Output:

```
{  
  "User": {  
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",  
    "Username": "exampleUser2",  
    "EmailAddress": "exampleUser2@site.awsapps.com",  
    "GivenName": "example2Name",  
    "Surname": "example2Surname",  
    "OrganizationId": "d-926726012c",  
    "RootFolderId":  
"35b886cb17198cbd547655e58b025dff0cf34aaed638be52009567e23dc67390",  
    "RecycleBinFolderId":  
"9858c3e9ed4c2460dde9aadb4c69fde998070dd46e5e985bd08ec6169ea249ff",  
    "Status": "ACTIVE",  
    "Type": "MINIMALUSER",  
    "CreatedTimestamp": 1535478836.584,  
  }  
}
```

```
"ModifiedTimestamp": 1535478836.584,  
"Storage": {  
  "StorageUtilizedInBytes": 0,  
  "StorageRule": {  
    "StorageAllocatedInBytes": 0,  
    "StorageType": "QUOTA"  
  }  
}  
}  
}
```

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## deactivate-user

Contoh kode berikut menunjukkan cara menggunakan `deactivate-user`.

### AWS CLI

Untuk menonaktifkan pengguna

Contoh ini menonaktifkan pengguna aktif.

Perintah:

```
aws workdocs deactivate-user --user-id  
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Output:

```
None
```

- Untuk detail API, lihat [DeactivateUser](#) di Referensi AWS CLI Perintah.

## delete-comment

Contoh kode berikut menunjukkan cara menggunakan `delete-comment`.

### AWS CLI

Untuk menghapus komentar tertentu dari versi dokumen

Contoh ini menghapus komentar yang ditentukan dari versi dokumen yang ditentukan.

Perintah:

```
aws workdocs delete-comment --document-id
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-id
1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
comment-id 1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteComment](#) di Referensi AWS CLI Perintah.

## **delete-custom-metadata**

Contoh kode berikut menunjukkan cara menggunakan `delete-custom-metadata`.

AWS CLI

Untuk menghapus metadata kustom dari sumber daya

Contoh ini menghapus semua metadata kustom dari sumber daya yang ditentukan.

Perintah:

```
aws workdocs delete-custom-metadata --resource-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --delete-all
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteCustomMetadata](#) di Referensi AWS CLI Perintah.

## **delete-document**

Contoh kode berikut menunjukkan cara menggunakan `delete-document`.

## AWS CLI

Untuk menghapus dokumen

Contoh ini menghapus dokumen yang ditentukan.

Perintah:

```
aws workdocs delete-document --document-id
b83ed5e5b167b65ef69de9d597627ff1a0d4f07a45e67f1fab7d26b54427de0a
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteDocument](#) di Referensi AWS CLI Perintah.

## delete-folder-contents

Contoh kode berikut menunjukkan cara menggunakan `delete-folder-contents`.

## AWS CLI

Untuk menghapus isi folder

Contoh ini menghapus isi folder yang ditentukan.

Perintah:

```
aws workdocs delete-folder-contents --folder-id
26fa8aa4ba2071447c194f7b150b07149dbdb9e1c8a301872dcd93a4735ce65d
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteFolderContents](#) di Referensi AWS CLI Perintah.

## delete-folder

Contoh kode berikut menunjukkan cara menggunakan `delete-folder`.

## AWS CLI

Untuk menghapus folder

Contoh ini menghapus folder yang ditentukan.

Perintah:

```
aws workdocs delete-folder --folder-id
26fa8aa4ba2071447c194f7b150b07149dbdb9e1c8a301872dcd93a4735ce65d
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteFolder](#) di Referensi AWS CLI Perintah.

## **delete-labels**

Contoh kode berikut menunjukkan cara menggunakan `delete-labels`.

## AWS CLI

Untuk menghapus label

Contoh ini menghapus label yang ditentukan dari dokumen.

Perintah:

```
aws workdocs delete-labels --resource-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --labels
"documents" "examples"
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteLabels](#) di Referensi AWS CLI Perintah.

## **delete-notification-subscription**

Contoh kode berikut menunjukkan cara menggunakan `delete-notification-subscription`.



## AWS CLI

Untuk menghapus langganan notifikasi

`delete-notification-subscription` Contoh berikut menghapus langganan notifikasi yang ditentukan.

```
aws workdocs delete-notification-subscription \  
  --subscription-id 123ab4c5-678d-901e-f23g-45h6789j0123 \  
  --organization-id d-123456789c
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan](#) di Panduan WorkDocs Pengembang Amazon.

- Untuk detail API, lihat [DeleteNotificationSubscription](#) di Referensi AWS CLI Perintah.

## `delete-user`

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

## AWS CLI

Untuk menghapus pengguna

Contoh ini menghapus pengguna.

Perintah:

```
aws workdocs delete-user --user-id  
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Output:

```
None
```

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## `describe-activities`

Contoh kode berikut menunjukkan cara menggunakan `describe-activities`.

## AWS CLI

Untuk mendapatkan daftar aktivitas pengguna

Contoh ini menampilkan daftar aktivitas pengguna terbaru untuk organisasi tertentu, dengan batas yang ditetapkan untuk dua aktivitas terbaru.

Perintah:

```
aws workdocs describe-activities --organization-id d-926726012c --limit 2
```

Output:

```
{
  "UserActivities": [
    {
      "Type": "DOCUMENT_VERSION_DOWNLOADED",
      "TimeStamp": 1534800122.17,
      "Initiator": {
        "Id": "arn:aws:iam::123456789123:user/exampleUser"
      },
      "ResourceMetadata": {
        "Type": "document",
        "Name": "updatedDoc",
        "Id":
"15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3",
        "Owner": {
          "Id":
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "GivenName": "exampleName",
          "Surname": "exampleSurname"
        }
      }
    },
    {
      "Type": "DOCUMENT_VERSION_VIEWED",
      "TimeStamp": 1534799079.207,
      "Initiator": {
        "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
        "GivenName": "exampleName",
        "Surname": "exampleSurname"
      },
      "ResourceMetadata": {
```

```

        "Type": "document",
        "Name": "updatedDoc",
        "Id":
"15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3",
        "Owner": {
            "Id":
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
            "GivenName": "exampleName",
            "Surname": "exampleSurname"
        }
    }
],
"Marker":
"DnF1ZXJ5VGhlbkZldGNoAgAAAAAAS7Fm1TaU10d1FTU1h1UU00VVFibD1RWHcAAAAAAAJTRY3bWh5eUgzaVF1ZX
}

```

- Untuk detail API, lihat [DescribeActivities](#) di Referensi AWS CLI Perintah.

## describe-comments

Contoh kode berikut menunjukkan cara menggunakan `describe-comments`.

### AWS CLI

Untuk daftar semua komentar untuk versi dokumen tertentu

Contoh ini mencantumkan semua komentar untuk versi dokumen yang ditentukan.

Perintah:

```

aws workdocs describe-comments --document-id
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-id
1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920

```

Output:

```

{
  "Comments": [
    {
      "CommentId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",

```

```

    "ThreadId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
    "Text": "This is a comment.",
    "Contributor": {
      "Username": "arn:aws:iam::123456789123:user/exampleUser",
      "Type": "USER"
    },
    "CreatedTimestamp": 1534799058.197,
    "Status": "PUBLISHED",
    "Visibility": "PUBLIC"
  }
]
}

```

- Untuk detail API, lihat [DescribeComments](#) di Referensi AWS CLI Perintah.

## describe-document-versions

Contoh kode berikut menunjukkan cara menggunakan `describe-document-versions`.

### AWS CLI

Untuk mengambil versi dokumen

Contoh ini mengambil versi dokumen untuk dokumen yang ditentukan, termasuk versi yang diinisialisasi dan URL untuk dokumen sumber.

Perintah:

```
aws workdocs describe-document-versions --document-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --fields SOURCE
```

Output:

```

{
  "DocumentVersions": [
    {
      "Id":
"1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
      "Name": "exampleDoc.docx",
      "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
      "Size": 13922,

```

```

    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Status": "ACTIVE",
    "CreatedTimestamp": 1534452029.587,
    "ModifiedTimestamp": 1534452029.849,
    "CreatorId":
    "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Source": {
        "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65/1534452029587-15e129dfc1875
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc29.docx&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180816T204149Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180816%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k23456781901234mno56pqr78EXAMPLE"
    }
  },
  {
    "Id": "1529005196082-
bb75fa19abc287699cb07147f75816dce43a53a10f28dc001bf61ef2fab01c59",
    "Name": "exampleDoc.pdf",
    "ContentType": "application/pdf",
    "Size": 425916,
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Status": "ACTIVE",
    "CreatedTimestamp": 1529005196.082,
    "ModifiedTimestamp": 1529005196.796,
    "CreatorId":
    "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Source": {
        "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65/1529005196082-
bb75fa19abc287699cb07147f75816dce43a53a10f28dc001bf61ef2fab01c59?
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc29.pdf&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180816T204149Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180816%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k23456781901234mno56pqr78EXAMPLE"
    }
  }
]
}

```

- Untuk detail API, lihat [DescribeDocumentVersions](#) di Referensi AWS CLI Perintah.

## describe-folder-contents

Contoh kode berikut menunjukkan cara menggunakan `describe-folder-contents`.

### AWS CLI

Untuk mendeskripsikan isi folder

Contoh ini menjelaskan semua konten aktif dari folder yang ditentukan, termasuk dokumen dan subfoldernya, diurutkan berdasarkan tanggal dalam urutan menaik.

Perintah:

```
aws workdocs describe-folder-contents --folder-id
1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678 --sort DATE --
order ASCENDING --type ALL
```

Output:

```
{
  "Folders": [
    {
      "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
      "Name": "testing",
      "CreatorId":
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
      "CreatedTimestamp": 1534450467.622,
      "ModifiedTimestamp": 1534451113.504,
      "ResourceState": "ACTIVE",
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Size": 23019,
      "LatestVersionSize": 11537
    }
  ],
  "Documents": [
    {
      "Id": "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
      "CreatorId":
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
```

```

    "ParentFolderId":
    "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1529005196.082,
    "ModifiedTimestamp": 1534452483.01,
    "LatestVersionMetadata": {
      "Id":
    "1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
      "Name": "exampleDoc.docx",
      "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
      "Size": 13922,
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Status": "ACTIVE",
      "CreatedTimestamp": 1534452029.587,
      "ModifiedTimestamp": 1534452029.587,
      "CreatorId":
    "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
      },
      "ResourceState": "ACTIVE"
    }
  ]
}

```

- Untuk detail API, lihat [DescribeFolderContents](#) di Referensi AWS CLI Perintah.

## describe-groups

Contoh kode berikut menunjukkan cara menggunakan `describe-groups`.

### AWS CLI

Untuk mengambil daftar grup

`describe-groups` Contoh berikut mencantumkan grup yang terkait dengan WorkDocs organisasi Amazon yang ditentukan.

```

aws workdocs describe-groups \
  --search-query "e" \
  --organization-id d-123456789c

```

Output:

```
{
  "Groups": [
    {
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444&d-123456789c",
      "Name": "Example Group 1"
    },
    {
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-5555&d-123456789c",
      "Name": "Example Group 2"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Memulai Amazon WorkDocs](#) di Panduan WorkDocs Administrasi Amazon.

- Untuk detail API, lihat [DescribeGroups](#) di Referensi AWS CLI Perintah.

## describe-notification-subscriptions

Contoh kode berikut menunjukkan cara menggunakan `describe-notification-subscriptions`.

### AWS CLI

Untuk mengambil daftar langganan notifikasi

`describe-notification-subscriptions` Contoh berikut mengambil langganan notifikasi untuk organisasi Amazon WorkDocs yang ditentukan.

```
aws workdocs describe-notification-subscriptions \
  --organization-id d-123456789c
```

Output:

```
{
  "Subscriptions": [
    {
      "SubscriptionId": "123ab4c5-678d-901e-f23g-45h6789j0123",
      "EndPoint": "https://example.com/example",
      "Protocol": "HTTPS"
    }
  ]
}
```



```
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Berlangganan Pemberitahuan](#) di Panduan WorkDocs Pengembang Amazon.

- Untuk detail API, lihat [DescribeNotificationSubscriptions](#) di Referensi AWS CLI Perintah.

## describe-resource-permissions

Contoh kode berikut menunjukkan cara menggunakan `describe-resource-permissions`.

### AWS CLI

Untuk mendapatkan daftar izin untuk sumber daya

Contoh ini mengembalikan daftar izin untuk sumber daya tertentu (dokumen atau folder).

Perintah:

```
aws workdocs describe-resource-permissions --resource-id  
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3
```

Output:

```
{  
  "Principals": [  
    {  
      "Id": "anonymous",  
      "Type": "ANONYMOUS",  
      "Roles": [  
        {  
          "Role": "VIEWER",  
          "Type": "DIRECT"  
        }  
      ]  
    },  
    {  
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",  
      "Type": "USER",  
      "Roles": [  

```

```

        {
            "Role": "OWNER",
            "Type": "DIRECT"
        }
    ],
},
{
    "Id": "d-926726012c",
    "Type": "ORGANIZATION",
    "Roles": [
        {
            "Role": "VIEWER",
            "Type": "INHERITED"
        }
    ]
}
]
}

```

- Untuk detail API, lihat [DescribeResourcePermissions](#) di Referensi AWS CLI Perintah.

## describe-users

Contoh kode berikut menunjukkan cara menggunakan `describe-users`.

### AWS CLI

Untuk mengambil detail untuk pengguna tertentu

Contoh ini mengambil rincian untuk semua pengguna dalam organisasi tertentu.

Perintah:

```
aws workdocs describe-users --organization-id d-926726012c
```

Output:

```

{
  "Users": [
    {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "Username": "example1User",

```

```

    "OrganizationId": "d-926726012c",
    "RootFolderId":
"3c0e3f849dd20a9771d937b9bbcc97e18796150ae56c26d64a4fa0320a2dedc9",
    "RecycleBinFolderId":
"c277f4c4d647be1f5147b3184ffa96e1e2bf708278b696cacba68ba13b91f4fe",
    "Status": "INACTIVE",
    "Type": "USER",
    "CreatedTimestamp": 1535478999.452,
    "ModifiedTimestamp": 1535478999.452
  },
  {
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-4444&d-926726012c",
    "Username": "example2User",
    "EmailAddress": "example2User@site.awsapps.com",
    "GivenName": "example2Name",
    "Surname": "example2Surname",
    "OrganizationId": "d-926726012c",
    "RootFolderId":
"35b886cb17198cbd547655e58b025dff0cf34aaed638be52009567e23dc67390",
    "RecycleBinFolderId":
"9858c3e9ed4c2460dde9aadb4c69fde998070dd46e5e985bd08ec6169ea249ff",
    "Status": "ACTIVE",
    "Type": "MINIMALUSER",
    "CreatedTimestamp": 1535478836.584,
    "ModifiedTimestamp": 1535478836.584
  }
]
}

```

- Untuk detail API, lihat [DescribeUsers](#) di Referensi AWS CLI Perintah.

## get-document-path

Contoh kode berikut menunjukkan cara menggunakan `get-document-path`.

### AWS CLI

Untuk mengambil informasi jalur dokumen

Contoh ini mengambil informasi jalur (hierarki dari folder root) untuk dokumen yang ditentukan, dan menyertakan nama folder induk.

Perintah:

```
aws workdocs get-document-path --document-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --fields NAME
```

### Output:

```
{
  "Path": {
    "Components": [
      {
        "Id":
"a43d29cbb8e7c4d25cfee8b803a504b0dc63e760b55ad0c611c6b87691eb6ff3",
        "Name": "/"
      },
      {
        "Id":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
        "Name": "Top Level Folder"
      },
      {
        "Id":
"d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
        "Name": "exampleDoc.docx"
      }
    ]
  }
}
```

- Untuk detail API, lihat [GetDocumentPath](#) di Referensi AWS CLI Perintah.

## get-document-version

Contoh kode berikut menunjukkan cara menggunakan `get-document-version`.

### AWS CLI

Untuk mengambil metadata versi untuk dokumen tertentu

Contoh ini mengambil metadata versi untuk dokumen yang ditentukan, termasuk URL sumber dan metadata kustom.

Perintah:

```
aws workdocs get-document-version --document-id
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-id
1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
fields SOURCE --include-custom-metadata
```

## Output:

```
{
  "Metadata": {
    "Id":
"1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920",
    "Name": "exampleDoc",
    "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
    "Size": 11537,
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Status": "ACTIVE",
    "CreatedTimestamp": 1521672507.741,
    "ModifiedTimestamp": 1534451113.504,
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Source": {
      "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3/152167
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180820T212202Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180820%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k2345678l901234mno56pqr78EXAMPLE"
    }
  }
}
```

- Untuk detail API, lihat [GetDocumentVersion](#) di Referensi AWS CLI Perintah.

## get-document

Contoh kode berikut menunjukkan cara menggunakan `get-document`.

### AWS CLI

Untuk mengambil detail dokumen

Contoh ini mengambil rincian dokumen yang ditentukan.

Perintah:

```
aws workdocs get-document --document-id
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65
```

Output:

```
{
  "Metadata": {
    "Id": "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1529005196.082,
    "ModifiedTimestamp": 1534452483.01,
    "LatestVersionMetadata": {
      "Id":
"1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
      "Name": "exampleDoc.docx",
      "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
      "Size": 13922,
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Status": "ACTIVE",
      "CreatedTimestamp": 1534452029.587,
      "ModifiedTimestamp": 1534452029.587,
      "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
    },
    "ResourceState": "ACTIVE"
  }
}
```

- Untuk detail API, lihat [GetDocument](#) di Referensi AWS CLI Perintah.

## get-folder-path

Contoh kode berikut menunjukkan cara menggunakan `get-folder-path`.

## AWS CLI

Untuk mengambil informasi jalur untuk folder

Contoh ini mengambil informasi jalur (hierarki dari folder root) untuk folder yang ditentukan, dan menyertakan nama folder induk.

Perintah:

```
aws workdocs get-folder-path --folder-id
50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08 --fields NAME
```

Output:

```
{
  "Path": {
    "Components": [
      {
        "Id":
"a43d29cbb8e7c4d25cfee8b803a504b0dc63e760b55ad0c611c6b87691eb6ff3",
        "Name": "/"
      },
      {
        "Id":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
        "Name": "Top Level Folder"
      },
      {
        "Id":
"50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
        "Name": "Sublevel Folder"
      }
    ]
  }
}
```

- Untuk detail API, lihat [GetFolderPath](#) di Referensi AWS CLI Perintah.

### get-folder

Contoh kode berikut menunjukkan cara menggunakan `get-folder`.

## AWS CLI

Untuk mengambil metadata untuk folder

Contoh ini mengambil metadata untuk folder tertentu.

Perintah:

```
aws workdocs get-folder --folder-id
50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08
```

Output:

```
{
  "Metadata": {
    "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
    "Name": "exampleFolder",
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1534450467.622,
    "ModifiedTimestamp": 1534451113.504,
    "ResourceState": "ACTIVE",
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Size": 23019,
    "LatestVersionSize": 11537
  }
}
```

- Untuk detail API, lihat [GetFolder](#) di Referensi AWS CLI Perintah.

## get-resources

Contoh kode berikut menunjukkan cara menggunakan `get-resources`.

## AWS CLI

Untuk mengambil sumber daya bersama

`get-resources` Contoh berikut mengambil sumber daya yang dibagikan dengan WorkDocs pengguna Amazon yang ditentukan.



```
aws workdocs get-resources \  
  --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333" \  
  --collection-type SHARED_WITH_ME
```

Output:

```
{  
  "Folders": [],  
  "Documents": []  
}
```

Untuk informasi selengkapnya, lihat [Berbagi File dan Folder](#) di Panduan WorkDocs Pengguna Amazon.

- Untuk detail API, lihat [GetResources](#) di Referensi AWS CLI Perintah.

## initiate-document-version-upload

Contoh kode berikut menunjukkan cara menggunakan `initiate-document-version-upload`.

AWS CLI

Untuk memulai upload versi dokumen

`initiate-document-upload` Contoh berikut membuat objek dokumen baru dan objek versi.

```
aws workdocs initiate-document-version-upload \  
  --name exampledocname \  
  --parent-folder-id  
  eacd546d952531c633452ed67cac23161aa0d5df2e8061223a59e8f67e7b6189
```

Output:

```
{  
  "Metadata": {  
    "Id": "feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2",  
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",  
    "ParentFolderId":  
    "eacd546d952531c633452ed67cac23161aa0d5df2e8061223a59e8f67e7b6189",  
    "CreatedTimestamp": 1536773972.914,  
  }  
}
```

```

    "ModifiedTimestamp": 1536773972.914,
    "LatestVersionMetadata": {
      "Id": "1536773972914-
ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dccccd417da9313",
      "Name": "exampledocname",
      "ContentType": "application/octet-stream",
      "Size": 0,
      "Status": "INITIALIZED",
      "CreatedTimestamp": 1536773972.914,
      "ModifiedTimestamp": 1536773972.914,
      "CreatorId": "arn:aws:iam::123456789123:user/EXAMPLE"
    },
    "ResourceState": "ACTIVE"
  },
  "UploadMetadata": {
    "UploadUrl": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2/1536773972914-
ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dccccd417da9313?X-Amz-
Algorithm=AWS1-ABCD-EFG234&X-Amz-Date=20180912T173932Z&X-Amz-SignedHeaders=content-
type%3Bhost%3Bx-amz-server-side-encryption&X-Amz-Expires=899&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180912%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k23456781901234mno56pqr78EXAMPLE",
    "SignedHeaders": {
      "Content-Type": "application/octet-stream",
      "x-amz-server-side-encryption": "ABC123"
    }
  }
}

```

- Untuk detail API, lihat [InitiateDocumentVersionUpload](#) di Referensi AWS CLI Perintah.

## remove-all-resource-permissions

Contoh kode berikut menunjukkan cara menggunakan `remove-all-resource-permissions`.

### AWS CLI

Untuk menghapus semua izin dari sumber daya tertentu

Contoh ini menghapus semua izin dari sumber daya yang ditentukan.

Perintah:

```
aws workdocs remove-all-resource-permissions --resource-id
1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Output:

```
None
```

- Untuk detail API, lihat [RemoveAllResourcePermissions](#) di Referensi AWS CLI Perintah.

## **remove-resource-permission**

Contoh kode berikut menunjukkan cara menggunakan `remove-resource-permission`.

### AWS CLI

Untuk menghapus izin dari sumber daya

Contoh ini menghapus izin dari sumber daya untuk prinsipal yang ditentukan.

Perintah:

```
aws workdocs remove-resource-permission --resource-id
1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678 --principal-id
anonymous
```

Output:

```
None
```

- Untuk detail API, lihat [RemoveResourcePermission](#) di Referensi AWS CLI Perintah.

## **update-document-version**

Contoh kode berikut menunjukkan cara menggunakan `update-document-version`.

### AWS CLI

Untuk mengubah status versi dokumen menjadi Aktif

Contoh ini mengubah status versi dokumen menjadi Aktif.

Perintah:

```
aws workdocs update-document-version --document-id
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-id
1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
version-status ACTIVE
```

Output:

None

- Untuk detail API, lihat [UpdateDocumentVersion](#) di Referensi AWS CLI Perintah.

## update-document

Contoh kode berikut menunjukkan cara menggunakan `update-document`.

AWS CLI

Untuk memperbarui dokumen

Contoh ini memperbarui nama dokumen dan folder induk.

Perintah:

```
aws workdocs update-document --document-id
15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --name updatedDoc
--parent-folder-id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08
```

Output:

None

- Untuk detail API, lihat [UpdateDocument](#) di Referensi AWS CLI Perintah.

## update-folder

Contoh kode berikut menunjukkan cara menggunakan `update-folder`.

## AWS CLI

Untuk memperbarui folder

Contoh ini memperbarui nama folder dan folder induk.

Perintah:

```
aws workdocs update-folder --folder-id
50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08 --name
exampleFolder1 --parent-folder-id
1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Output:

```
None
```

- Untuk detail API, lihat [UpdateFolder](#) di Referensi AWS CLI Perintah.

## update-user

Contoh kode berikut menunjukkan cara menggunakan `update-user`.

## AWS CLI

Untuk memperbarui pengguna

Contoh ini memperbarui zona waktu untuk pengguna yang ditentukan.

Perintah:

```
aws workdocs update-user --user-id
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c" --time-zone-id
"America/Los_Angeles"
```

Output:

```
{
  "User": {
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
```

```
"Username": "exampleUser",
"EmailAddress": "exampleUser@site.awsapps.com",
"GivenName": "Example",
"Surname": "User",
"OrganizationId": "d-926726012c",
"RootFolderId":
"c5eceb5e1a2d1d460c9d1af8330ae117fc8d39bb1d3ed6acd0992d5ff192d986",
"RecycleBinFolderId":
"6ca20102926ad15f04b1d248d6d6e44f2449944eda5c758f9a1e9df6a6b7fa66",
"Status": "ACTIVE",
"Type": "USER",
"TimeZoneId": "America/Los_Angeles",
"Storage": {
  "StorageUtilizedInBytes": 0,
  "StorageRule": {
    "StorageAllocatedInBytes": 53687091200,
    "StorageType": "QUOTA"
  }
}
}
```

- Untuk detail API, lihat [UpdateUser](#) di Referensi AWS CLI Perintah.

## WorkMail Contoh Amazon menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface With Amazon WorkMail.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

### Topik

- [Tindakan](#)

## Tindakan

### **associate-delegate-to-resource**

Contoh kode berikut menunjukkan cara menggunakan `associate-delegate-to-resource`.

#### AWS CLI

Untuk menambahkan delegasi ke sumber daya

`associate-delegate-to-resource` Perintah berikut menambahkan delegasi ke sumber daya.

```
aws workmail associate-delegate-to-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-68bf2d3b1c0244aab7264c24b9217443 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AssociateDelegateToResource](#) di Referensi AWS CLI Perintah.

### **associate-member-to-group**

Contoh kode berikut menunjukkan cara menggunakan `associate-member-to-group`.

#### AWS CLI

Untuk menambahkan anggota ke grup

`associate-member-to-group` Perintah berikut menambahkan anggota yang ditentukan ke grup.

```
aws workmail associate-member-to-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --member-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [AssociateMemberToGroup](#) di Referensi AWS CLI Perintah.

## create-alias

Contoh kode berikut menunjukkan cara menggunakan `create-alias`.

### AWS CLI

Untuk membuat alias

`create-alias` Perintah berikut membuat alias untuk entitas tertentu (pengguna atau grup).

```
aws workmail create-alias \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --alias exampleAlias@site.awsapps.com
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [CreateAlias](#) di Referensi AWS CLI Perintah.

## create-group

Contoh kode berikut menunjukkan cara menggunakan `create-group`.

### AWS CLI

Untuk membuat grup baru

`create-group` Perintah berikut membuat grup baru untuk organisasi tertentu.

```
aws workmail create-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --name exampleGroup1
```

Output:

```
{  
  "GroupId": "S-1-1-11-1122222222-2222233333-3333334444-4444"  
}
```

- Untuk detail API, lihat [CreateGroup](#) di Referensi AWS CLI Perintah.



## create-resource

Contoh kode berikut menunjukkan cara menggunakan `create-resource`.

### AWS CLI

Untuk membuat sumber daya baru

`create-resource` Perintah berikut membuat sumber daya baru (ruang rapat) untuk organisasi tertentu.

```
aws workmail create-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --name exampleRoom1 \  
  --type ROOM
```

Output:

```
{  
  "ResourceId": "r-7afe0efbade843a58cdc10251fce992c"  
}
```

- Untuk detail API, lihat [CreateResource](#) di Referensi AWS CLI Perintah.

## create-user

Contoh kode berikut menunjukkan cara menggunakan `create-user`.

### AWS CLI

Untuk membuat pengguna baru

`create-user` Perintah berikut menciptakan pengguna baru.

```
aws workmail create-user \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --name exampleName \  
  --display-name exampleDisplayName \  
  --password examplePa$$w0rd
```

Output:

```
{
  "UserId": "S-1-1-11-1111111111-2222222222-3333333333-3333"
}
```

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## delete-access-control-rule

Contoh kode berikut menunjukkan cara menggunakan `delete-access-control-rule`.

### AWS CLI

Untuk menghapus aturan kontrol akses

`delete-access-control-rule` Contoh berikut menghapus aturan kontrol akses yang ditentukan dari WorkMail organisasi Amazon yang ditentukan.

```
aws workmail delete-access-control-rule \
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \
  --name "myRule"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Aturan Kontrol Akses](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [DeleteAccessControlRule](#) di Referensi AWS CLI Perintah.

## delete-alias

Contoh kode berikut menunjukkan cara menggunakan `delete-alias`.

### AWS CLI

Untuk menghapus alias

`delete-alias` Perintah berikut menghapus alias untuk entitas tertentu (pengguna atau grup).

```
aws workmail delete-alias \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
```

```
--alias exampleAlias@site.awsapps.com
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteAlias](#) di Referensi AWS CLI Perintah.

## **delete-group**

Contoh kode berikut menunjukkan cara menggunakan `delete-group`.

### AWS CLI

Untuk menghapus grup yang ada

`delete-group` Perintah berikut menghapus grup yang ada dari Amazon WorkMail.

```
aws workmail delete-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteGroup](#) di Referensi AWS CLI Perintah.

## **delete-mailbox-permissions**

Contoh kode berikut menunjukkan cara menggunakan `delete-mailbox-permissions`.

### AWS CLI

Untuk menghapus izin kotak pesan

`delete-mailbox-permissions` Perintah berikut menghapus izin kotak pesan yang sebelumnya diberikan kepada pengguna atau grup. Entitas mewakili pengguna yang memiliki kotak pesan, dan penerima hibah mewakili pengguna atau grup yang akan menghapus izin.

```
aws workmail delete-mailbox-permissions \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --grantee-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteMailboxPermissions](#) di Referensi AWS CLI Perintah.

## **delete-resource**

Contoh kode berikut menunjukkan cara menggunakan `delete-resource`.

### AWS CLI

Untuk menghapus sumber daya yang ada

`delete-resource` Perintah berikut menghapus sumber daya yang ada dari Amazon WorkMail.

```
aws workmail delete-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-7afe0efbade843a58cdc10251fce992c
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteResource](#) di Referensi AWS CLI Perintah.

## **delete-user**

Contoh kode berikut menunjukkan cara menggunakan `delete-user`.

### AWS CLI

Untuk menghapus pengguna

`delete-user` Perintah berikut menghapus pengguna yang ditentukan dari Amazon WorkMail dan semua sistem berikutnya.

```
aws workmail delete-user \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## **deregister-from-work-mail**

Contoh kode berikut menunjukkan cara menggunakan `deregister-from-work-mail`.

## AWS CLI

Untuk menonaktifkan entitas yang ada

`deregister-from-work-mail` Perintah berikut menonaktifkan entitas yang ada (pengguna, grup, atau sumber daya) dari menggunakan Amazon WorkMail.

```
aws workmail deregister-from-work-mail \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DeregisterFromWorkMail](#) di Referensi AWS CLI Perintah.

## `describe-group`

Contoh kode berikut menunjukkan cara menggunakan `describe-group`.

## AWS CLI

Untuk mengambil informasi untuk grup

`describe-group` Perintah berikut mengambil informasi tentang grup yang ditentukan.

```
aws workmail describe-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Output:

```
{  
  "GroupId": "S-1-1-11-1122222222-2222233333-3333334444-4444",  
  "Name": "exampleGroup1",  
  "State": "ENABLED"  
}
```

- Untuk detail API, lihat [DescribeGroup](#) di Referensi AWS CLI Perintah.

## `describe-organization`

Contoh kode berikut menunjukkan cara menggunakan `describe-organization`.

## AWS CLI

Untuk mengambil informasi untuk organisasi

`describe-organization` Perintah berikut mengambil informasi untuk WorkMail organisasi Amazon yang ditentukan.

```
aws workmail describe-organization \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{  
  "OrganizationId": "m-d281d0a2fd824be5b6cd3d3ce909fd27",  
  "Alias": "alias",  
  "State": "Active",  
  "DirectoryId": "d-926726012c",  
  "DirectoryType": "VpcDirectory",  
  "DefaultMailDomain": "site.awsapps.com",  
  "CompletedDate": 1522693605.468,  
  "ARN": "arn:aws:workmail:us-west-2:111122223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza"  
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Organizations](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [DescribeOrganization](#) di Referensi AWS CLI Perintah.

## **describe-resource**

Contoh kode berikut menunjukkan cara menggunakan `describe-resource`.

### AWS CLI

Untuk mengambil informasi untuk sumber daya

`describe-resource` Perintah berikut mengambil informasi tentang sumber daya yang ditentukan.

```
aws workmail describe-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-926726012c
```

```
--resource-id r-7afe0efbade843a58cdc10251fce992c
```

### Output:

```
{
  "ResourceId": "r-7afe0efbade843a58cdc10251fce992c",
  "Name": "exampleRoom1",
  "Type": "ROOM",
  "BookingOptions": {
    "AutoAcceptRequests": true,
    "AutoDeclineRecurringRequests": false,
    "AutoDeclineConflictingRequests": true
  },
  "State": "ENABLED"
}
```

- Untuk detail API, lihat [DescribeResource](#) di Referensi AWS CLI Perintah.

## describe-user

Contoh kode berikut menunjukkan cara menggunakan `describe-user`.

### AWS CLI

Untuk mengambil informasi pengguna

`describe-user` Perintah berikut mengambil informasi tentang pengguna yang ditentukan.

```
aws workmail describe-user \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

### Output:

```
{
  "UserId": "S-1-1-11-1111111111-2222222222-3333333333-3333",
  "Name": "exampleUser1",
  "Email": "exampleUser1@site.awsapps.com",
  "DisplayName": "",
  "State": "ENABLED",
  "UserRole": "USER",
  "EnabledDate": 1532459261.827
}
```

```
}
```

- Untuk detail API, lihat [DescribeUser](#) di Referensi AWS CLI Perintah.

## **disassociate-delegate-from-resource**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-delegate-from-resource`.

### AWS CLI

Untuk menghapus anggota dari sumber daya

`disassociate-delegate-from-resource` Perintah berikut menghapus anggota yang ditentukan dari sumber daya.

```
aws workmail disassociate-delegate-from-resource \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --resource-id r-68bf2d3b1c0244aab7264c24b9217443 \
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [DisassociateDelegateFromResource](#) di Referensi AWS CLI Perintah.

## **disassociate-member-from-group**

Contoh kode berikut menunjukkan cara menggunakan `disassociate-member-from-group`.

### AWS CLI

Untuk menghapus anggota dari grup

`disassociate-member-from-group` Perintah berikut menghapus anggota yang ditentukan dari grup.

```
aws workmail disassociate-member-from-group \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
  --member-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Perintah ini tidak menghasilkan output.



- Untuk detail API, lihat [DisassociateMemberFromGroup](#) di Referensi AWS CLI Perintah.

## get-access-control-effect

Contoh kode berikut menunjukkan cara menggunakan `get-access-control-effect`.

### AWS CLI

Untuk mendapatkan efek aturan kontrol akses

`get-access-control-effect` Contoh berikut mengambil efek dari aturan kontrol akses WorkMail organisasi Amazon yang ditentukan untuk alamat IP yang ditentukan, tindakan protokol akses, dan ID pengguna.

```
aws workmail get-access-control-effect \
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \
  --ip-address "192.0.2.0" \
  --action "WindowsOutlook" \
  --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333"
```

Output:

```
{
  "Effect": "DENY",
  "MatchedRules": [
    "myRule"
  ]
}
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Aturan Kontrol Akses](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [GetAccessControlEffect](#) di Referensi AWS CLI Perintah.

## get-mailbox-details

Contoh kode berikut menunjukkan cara menggunakan `get-mailbox-details`.

### AWS CLI

Untuk mendapatkan detail kotak pesan pengguna

`get-mailbox-details` Perintah berikut mengambil rincian tentang kotak pesan pengguna yang ditentukan.

```
aws workmail get-mailbox-details \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{  
  "MailboxQuota": 51200,  
  "MailboxSize": 0.03890800476074219  
}
```

Untuk informasi selengkapnya, lihat [Mengelola Akun Pengguna](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [GetMailboxDetails](#) di Referensi AWS CLI Perintah.

## **list-access-control-rules**

Contoh kode berikut menunjukkan cara menggunakan `list-access-control-rules`.

### AWS CLI

Untuk membuat daftar aturan kontrol akses

`list-access-control-rules` Contoh berikut mencantumkan aturan kontrol akses untuk WorkMail organisasi Amazon yang ditentukan.

```
aws workmail list-access-control-rules \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza
```

Output:

```
{  
  "Rules": [  
    {  
      "Name": "default",  
      "Effect": "ALLOW",  
      "Description": "Default WorkMail Rule",
```

```
    "DateCreated": 0.0,  
    "DateModified": 0.0  
  },  
  {  
    "Name": "myRule",  
    "Effect": "DENY",  
    "Description": "my rule",  
    "UserIds": [  
      "S-1-1-11-1111111111-2222222222-3333333333-3333"  
    ],  
    "DateCreated": 1581635628.0,  
    "DateModified": 1581635628.0  
  }  
]
```

Untuk informasi selengkapnya, lihat [Bekerja dengan Aturan Kontrol Akses](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [ListAccessControlRules](#) di Referensi AWS CLI Perintah.

## list-aliases

Contoh kode berikut menunjukkan cara menggunakan `list-aliases`.

### AWS CLI

Untuk membuat daftar alias untuk anggota

`list-aliases` Perintah berikut mencantumkan alias untuk anggota tertentu (pengguna atau grup).

```
aws workmail list-aliases \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{  
  "Aliases": [  
    "exampleAlias@site.awsapps.com",  
    "exampleAlias1@site.awsapps.com"  ]  
}
```

```
]
}
```

- Untuk detail API, lihat [ListAliases](#) di Referensi AWS CLI Perintah.

## **list-group-members**

Contoh kode berikut menunjukkan cara menggunakan `list-group-members`.

### AWS CLI

Untuk daftar anggota grup

`list-group-members` Perintah berikut mencantumkan anggota grup yang ditentukan.

```
aws workmail list-group-members \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Output:

```
{
  "Members": [
    {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",
      "Name": "exampleUser1",
      "Type": "USER",
      "State": "ENABLED",
      "EnabledDate": 1532459261.827
    }
  ]
}
```

- Untuk detail API, lihat [ListGroupMembers](#) di Referensi AWS CLI Perintah.

## **list-groups**

Contoh kode berikut menunjukkan cara menggunakan `list-groups`.

### AWS CLI

Untuk mengambil daftar grup

**list-groups** Perintah berikut mengambil ringkasan grup dalam organisasi yang ditentukan.

```
aws workmail list-groups \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{  
  "Groups": [  
    {  
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444",  
      "Name": "exampleGroup1",  
      "State": "DISABLED"  
    },  
    {  
      "Id": "S-4-4-44-1122222222-2222233333-3333334444-4444",  
      "Name": "exampleGroup2",  
      "State": "ENABLED"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListGroups](#) di Referensi AWS CLI Perintah.

## **list-mailbox-permissions**

Contoh kode berikut menunjukkan cara menggunakan `list-mailbox-permissions`.

AWS CLI

Untuk mengambil izin kotak pesan

**list-mailbox-permissions** Perintah berikut mengambil izin kotak pesan yang terkait dengan kotak pesan entitas tertentu.

```
aws workmail list-mailbox-permissions \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{
```

```
"Permissions": [
  {
    "GranteeId": "S-1-1-11-1122222222-2222233333-3333334444-4444",
    "GranteeType": "USER",
    "PermissionValues": [
      "FULL_ACCESS"
    ]
  }
]
```

- Untuk detail API, lihat [ListMailboxPermissions](#) di Referensi AWS CLI Perintah.

## list-organizations

Contoh kode berikut menunjukkan cara menggunakan `list-organizations`.

### AWS CLI

Untuk mengambil daftar organisasi

`list-organizations` Perintah berikut mengambil ringkasan organisasi yang tidak dihapus.

```
aws workmail list-organizations
```

Output:

```
{
  "OrganizationSummaries": [
    {
      "OrganizationId": "m-d281d0a2fd824be5b6cd3d3ce909fd27",
      "Alias": "exampleAlias",
      "State": "Active"
    }
  ]
}
```

- Untuk detail API, lihat [ListOrganizations](#) di Referensi AWS CLI Perintah.

## list-resource-delegates

Contoh kode berikut menunjukkan cara menggunakan `list-resource-delegates`.

## AWS CLI

Untuk membuat daftar delegasi untuk sumber daya

`list-resource-delegates` Perintah berikut mengambil delegasi yang terkait dengan sumber daya yang ditentukan.

```
aws workmail list-resource-delegates \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-68bf2d3b1c0244aab7264c24b9217443
```

Output:

```
{  
  "Delegates": [  
    {  
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",  
      "Type": "USER"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListResourceDelegates](#) di Referensi AWS CLI Perintah.

## `list-resources`

Contoh kode berikut menunjukkan cara menggunakan `list-resources`.

## AWS CLI

Untuk mengambil daftar sumber daya

`list-resources` Perintah berikut mengambil ringkasan sumber daya untuk organisasi yang ditentukan.

```
aws workmail list-resources \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{
```

```
"Resources": [  
  {  
    "Id": "r-7afe0efbade843a58cdc10251fce992c",  
    "Name": "exampleRoom1",  
    "Type": "ROOM",  
    "State": "ENABLED"  
  }  
]
```

- Untuk detail API, lihat [ListResources](#) di Referensi AWS CLI Perintah.

## list-tags-for-resource

Contoh kode berikut menunjukkan cara menggunakan `list-tags-for-resource`.

### AWS CLI

Untuk membuat daftar tag untuk sumber daya

`list-tags-for-resource` Contoh berikut mencantumkan tag untuk WorkMail organisasi Amazon yang ditentukan.

```
aws workmail list-tags-for-resource \  
  --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza
```

Output:

```
{  
  "Tags": [  
    {  
      "Key": "priority",  
      "Value": "1"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai Organisasi](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [ListTagsForResource](#) di Referensi AWS CLI Perintah.



## list-users

Contoh kode berikut menunjukkan cara menggunakan `list-users`.

### AWS CLI

Untuk mengambil daftar pengguna

`list-users` Perintah berikut mengambil ringkasan dari pengguna dalam organisasi tertentu.

```
aws workmail list-users \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{  
  "Users": [  
    {  
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",  
      "Email": "exampleUser1@site.awsapps.com",  
      "Name": "exampleUser1",  
      "State": "ENABLED",  
      "UserRole": "USER",  
      "EnabledDate": 1532459261.827  
    },  
    {  
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444",  
      "Name": "exampleGuestUser",  
      "State": "DISABLED",  
      "UserRole": "SYSTEM_USER"  
    }  
  ]  
}
```

- Untuk detail API, lihat [ListUsers](#) di Referensi AWS CLI Perintah.

## put-access-control-rule

Contoh kode berikut menunjukkan cara menggunakan `put-access-control-rule`.

### AWS CLI

Untuk menempatkan aturan kontrol akses baru

`put-access-control-rule` Contoh berikut menyangkal akses pengguna yang ditentukan ke WorkMail organisasi Amazon yang ditentukan.

```
aws workmail put-access-control-rule \  
  --name "myRule" \  
  --effect "DENY" \  
  --description "my rule" \  
  --user-ids "S-1-1-11-1111111111-2222222222-3333333333-3333" \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Bekerja dengan Aturan Kontrol Akses](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [PutAccessControlRule](#) di Referensi AWS CLI Perintah.

## **put-mailbox-permissions**

Contoh kode berikut menunjukkan cara menggunakan `put-mailbox-permissions`.

### AWS CLI

Untuk menyetel izin kotak pesan

`put-mailbox-permissions` Perintah berikut menetapkan izin akses penuh untuk penerima hibah tertentu (pengguna atau grup). Entitas mewakili pemilik kotak pesan.

```
aws workmail put-mailbox-permissions \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
  --grantee-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --permission-values FULL_ACCESS
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [PutMailboxPermissions](#) di Referensi AWS CLI Perintah.

## **register-to-work-mail**

Contoh kode berikut menunjukkan cara menggunakan `register-to-work-mail`.

## AWS CLI

Untuk mendaftarkan entitas yang ada atau dinonaktifkan

`register-to-work-mail` Perintah berikut memungkinkan entitas yang sudah ada (pengguna, grup, atau sumber daya) yang ditentukan untuk menggunakan Amazon WorkMail.

```
aws workmail register-to-work-mail \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --email exampleGroup1@site.awsapps.com
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [RegisterToWorkMail](#) di Referensi AWS CLI Perintah.

## reset-password

Contoh kode berikut menunjukkan cara menggunakan `reset-password`.

### AWS CLI

Untuk mengatur ulang kata sandi pengguna

`reset-password` Perintah berikut mengatur ulang kata sandi untuk pengguna yang ditentukan.

```
aws workmail reset-password \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
  --password examplePa$$w0rd
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [ResetPassword](#) di Referensi AWS CLI Perintah.

## tag-resource

Contoh kode berikut menunjukkan cara menggunakan `tag-resource`.

### AWS CLI

Untuk menerapkan tag ke sumber daya

`tag-resource` Contoh berikut menerapkan tag dengan kunci “prioritas” dan nilai “1” ke WorkMail organisasi Amazon yang ditentukan.

```
aws workmail tag-resource \  
  --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza \  
  --tags "Key=priority,Value=1"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Organisasi](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [TagResource](#) di Referensi AWS CLI Perintah.

## **untag-resource**

Contoh kode berikut menunjukkan cara menggunakan `untag-resource`.

### AWS CLI

Untuk menghapus tag sumber daya

`untag-resource` Contoh berikut menghapus tag yang ditentukan dari WorkMail organisasi Amazon yang ditentukan.

```
aws workmail untag-resource \  
  --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza \  
  --tag-keys "priority"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai Organisasi](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [UntagResource](#) di Referensi AWS CLI Perintah.

## **update-mailbox-quota**

Contoh kode berikut menunjukkan cara menggunakan `update-mailbox-quota`.

## AWS CLI

Untuk memperbarui kuota kotak pesan pengguna

`update-mailbox-quota` Perintah berikut mengubah kuota kotak pesan pengguna yang ditentukan.

```
aws workmail update-mailbox-quota \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
  --mailbox-quota 40000
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengelola Akun Pengguna](#) di Panduan WorkMail Administrator Amazon.

- Untuk detail API, lihat [UpdateMailboxQuota](#) di Referensi AWS CLI Perintah.

## `update-primary-email-address`

Contoh kode berikut menunjukkan cara menggunakan `update-primary-email-address`.

## AWS CLI

Untuk memperbarui alamat email utama

`update-primary-email-address` Perintah berikut memperbarui alamat email utama dari entitas tertentu (pengguna, grup, atau sumber daya).

```
aws workmail update-primary-email-address \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
  --email exampleUser2@site.awsapps.com
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdatePrimaryEmailAddress](#) di Referensi AWS CLI Perintah.

## `update-resource`

Contoh kode berikut menunjukkan cara menggunakan `update-resource`.

## AWS CLI

Untuk memperbarui sumber daya

`update-resource` Perintah berikut memperbarui nama sumber daya yang ditentukan.

```
aws workmail update-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-7afe0efbade843a58cdc10251fce992c \  
  --name exampleRoom2
```

Perintah ini tidak menghasilkan output.

- Untuk detail API, lihat [UpdateResource](#) di Referensi AWS CLI Perintah.

## Contoh Amazon WorkMail Message Flow menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda cara melakukan tindakan dan mengimplementasikan skenario umum AWS Command Line Interface dengan menggunakan Alur WorkMail Pesan Amazon.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **get-raw-message-content**

Contoh kode berikut menunjukkan cara menggunakan `get-raw-message-content`.

AWS CLI

Untuk mendapatkan konten mentah dari pesan email

`get-raw-message-content` Contoh berikut mendapatkan konten mentah dari pesan email dalam transit dan mengirimkannya ke file teks bernama `test`.

```
aws workmailmessageflow get-raw-message-content \  
  --message-id a1b2cd34-ef5g-6h7j-kl8m-npq9012345rs \  
  test
```

Isi file `test` setelah perintah berjalan:

```
Subject: Hello World  
From: =?UTF-8?Q?marymajor_marymajor?= <marymajor@example.com>  
To: =?UTF-8?Q?mateojackson=40example=2Enet?= <mateojackson@example.net>  
Date: Thu, 7 Nov 2019 19:22:46 +0000  
Mime-Version: 1.0  
Content-Type: multipart/alternative;  
  boundary="=_EXAMPLE+"  
References: <mail.1ab23c45.5de6.7f890g123hj45678@storage.wm.amazon.com>  
X-Priority: 3 (Normal)  
X-Mailer: Amazon WorkMail  
Thread-Index: EXAMPLE  
Thread-Topic: Hello World  
Message-Id: <mail.1ab23c45.5de6.7f890g123hj45678@storage.wm.amazon.com>  
  
This is a multi-part message in MIME format. Your mail reader does not  
understand MIME message format.  
--=_EXAMPLE+  
Content-Type: text/plain; charset=UTF-8  
Content-Transfer-Encoding: 7bit  
  
hello world  
  
--=_EXAMPLE+  
Content-Type: text/html; charset=utf-8  
Content-Transfer-Encoding: quoted-printable  
  
<!DOCTYPE HTML><html>  
<head>  
<meta name=3D"Generator" content=3D"Amazon WorkMail v3.0-4510">  
<meta http-equiv=3D"Content-Type" content=3D"text/html; charset=3Dutf-8">=  
  
<title>testing</title>  
</head>
```

```
<body>
<p style="margin: 0px; font-family: Arial, Tahoma, Helvetica, sans-serif; font-size: small;">hello world</p>
</body>
</html>
--=_EXAMPLE+--
```

Untuk informasi selengkapnya, lihat [Mengambil Konten Pesan dengan AWS Lambda](#) di Panduan Administrator WorkMail Amazon.

- Untuk detail API, lihat [GetRawMessageContent](#) di Referensi AWS CLI Perintah.

## WorkSpaces contoh menggunakan AWS CLI

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan AWS Command Line Interface with WorkSpaces.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **create-tags**

Contoh kode berikut menunjukkan cara menggunakan `create-tags`.

AWS CLI

Untuk menambahkan tag ke Workspace

`create-tags` Contoh berikut menambahkan tag yang ditentukan ke yang ditentukan Workspace.



```
aws workspaces create-tags \  
  --resource-id ws-dk1xzc417 \  
  --tags Key=Department,Value=Finance
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai WorkSpaces sumber daya](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [CreateTags](#) di Referensi AWS CLI Perintah.

## create-workspaces

Contoh kode berikut menunjukkan cara menggunakan `create-workspaces`.

### AWS CLI

Contoh 1: Untuk membuat AlwaysOn WorkSpace

`create-workspaces` Contoh berikut menciptakan AlwaysOn WorkSpace untuk pengguna tertentu, menggunakan direktori dan bundel yang ditentukan.

```
aws workspaces create-workspaces \  
  --workspaces DirectoryId=d-926722edaf,UserName=Mateo,BundleId=wsb-0zsvgp8fc
```

Output:

```
{  
  "FailedRequests": [],  
  "PendingRequests": [  
    {  
      "WorkspaceId": "ws-kcqms853t",  
      "DirectoryId": "d-926722edaf",  
      "UserName": "Mateo",  
      "State": "PENDING",  
      "BundleId": "wsb-0zsvgp8fc"  
    }  
  ]  
}
```

Contoh 2: Untuk membuat AutoStop WorkSpace

`create-workspaces` Contoh berikut menciptakan AutoStop WorkSpace untuk pengguna tertentu, menggunakan direktori dan bundel yang ditentukan.

```
aws workspaces create-workspaces \
  --workspaces
  DirectoryId=d-926722edaf,UserName=Mary,BundleId=wsb-0zsvgp8fc,WorkspaceProperties={RunningM
```

Output:

```
{
  "FailedRequests": [],
  "PendingRequests": [
    {
      "WorkspaceId": "ws-dk1x zr417",
      "DirectoryId": "d-926722edaf",
      "UserName": "Mary",
      "State": "PENDING",
      "BundleId": "wsb-0zsvgp8fc"
    }
  ]
}
```

Contoh 3: Untuk membuat user-decoupled WorkSpace

`create-workspaces` Contoh berikut membuat user-decoupled WorkSpace dengan menyetel nama pengguna ke[UNDEFINED], dan menentukan nama, ID direktori, dan WorkSpace ID bundel.

```
aws workspaces create-workspaces \
  --workspaces
  DirectoryId=d-926722edaf,UserName=' "[UNDEFINED]" ',WorkspaceName=MaryWorkspace1,BundleId=wsb
```

Output:

```
{
  "FailedRequests": [],
  "PendingRequests": [
    {
      "WorkspaceId": "ws-abcd1234",
      "DirectoryId": "d-926722edaf",
```

```
    "UserName": "[UNDEFINED]",
    "State": "PENDING",
    "BundleId": "wsb-0zsvgp8fc",
    "WorkspaceName": "MaryWorkspace1"
  }
]
```

Untuk informasi selengkapnya, lihat [Meluncurkan desktop virtual](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [CreateWorkspaces](#) di Referensi AWS CLI Perintah.

## delete-tags

Contoh kode berikut menunjukkan cara menggunakan `delete-tags`.

### AWS CLI

Untuk menghapus tag dari WorkSpace

`delete-tags` Contoh berikut menghapus tag yang ditentukan dari yang ditentukan WorkSpace.

```
aws workspaces delete-tags \
  --resource-id ws-dk1x zr417 \
  --tag-keys Department
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Menandai WorkSpaces sumber daya](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DeleteTags](#) di Referensi AWS CLI Perintah.

## deregister-workspace-directory

Contoh kode berikut menunjukkan cara menggunakan `deregister-workspace-directory`.

### AWS CLI

Untuk membatalkan pendaftaran direktori

`deregister-workspace-directory` Contoh berikut membatalkan pendaftaran direktori yang ditentukan.

```
aws workspaces deregister-workspace-directory \  
  --directory-id d-926722edaf
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendaftarkan direktori dengan WorkSpaces](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DeregisterWorkspaceDirectory](#) di Referensi AWS CLI Perintah.

## describe-tags

Contoh kode berikut menunjukkan cara menggunakan `describe-tags`.

### AWS CLI

Untuk mendeskripsikan tag untuk a Workspace

`describe-tags` Contoh berikut menjelaskan tag untuk yang ditentukan Workspace.

```
aws workspaces describe-tags \  
  --resource-id ws-dk1x zr417
```

Output:

```
{  
  "TagList": [  
    {  
      "Key": "Department",  
      "Value": "Finance"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Menandai WorkSpaces sumber daya](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DescribeTags](#) di Referensi AWS CLI Perintah.

## describe-workspace-bundles

Contoh kode berikut menunjukkan cara menggunakan `describe-workspace-bundles`.

### AWS CLI

Untuk membuat daftar bundel yang disediakan oleh Amazon

`describe-workspace-bundles` Contoh berikut mencantumkan nama dan ID bundel yang disediakan oleh Amazon, dalam format tabel dan diurutkan berdasarkan nama.

```
aws workspaces describe-workspace-bundles \
  --owner AMAZON \
  --query "Bundles[*].[Name, BundleId]"
```

Output:

```
[
  [
    "Standard with Amazon Linux 2",
    "wsb-clj85qzj1"
  ],
  [
    "Performance with Windows 10 (Server 2016 based)",
    "wsb-gm4d5tx2v"
  ],
  [
    "PowerPro with Windows 7",
    "wsb-1pzkp0bx4"
  ],
  [
    "Power with Amazon Linux 2",
    "wsb-2bs6k5lgn"
  ],
  [
    "Graphics with Windows 10 (Server 2019 based)",
    "wsb-03gyjnfyy"
  ],
  ...
]
```

Untuk informasi selengkapnya, lihat [WorkSpaces bundel dan gambar](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DescribeWorkspaceBundles](#) di Referensi AWS CLI Perintah.

## describe-workspace-directories

Contoh kode berikut menunjukkan cara menggunakan `describe-workspace-directories`.

### AWS CLI

Untuk menggambarkan direktori terdaftar

`describe-workspace-directories` Contoh berikut menjelaskan direktori terdaftar yang ditentukan.

```
aws workspaces describe-workspace-directories \  
  --directory-ids d-926722edaf
```

Output:

```
{  
  "Directories": [  
    {  
      "DirectoryId": "d-926722edaf",  
      "Alias": "d-926722edaf",  
      "DirectoryName": "example.com",  
      "RegistrationCode": "WSpdx+9RJ8JT",  
      "SubnetIds": [  
        "subnet-9d19c4c6",  
        "subnet-500d5819"  
      ],  
      "DnsIpAddresses": [  
        "172.16.1.140",  
        "172.16.0.30"  
      ],  
      "CustomerUserName": "Administrator",  
      "IamRoleId": "arn:aws:iam::123456789012:role/workspaces_DefaultRole",  
      "DirectoryType": "SIMPLE_AD",  
      "WorkspaceSecurityGroupId": "sg-0d89e927e5645d7c5",  
      "State": "REGISTERED",  
      "WorkspaceCreationProperties": {  
        "EnableWorkDocs": false,  
        "EnableInternetAccess": false,  
        "UserEnabledAsLocalAdministrator": true,  
      }  
    }  
  ]  
}
```

```

        "EnableMaintenanceMode": true
    },
    "WorkspaceAccessProperties": {
        "DeviceTypeWindows": "ALLOW",
        "DeviceTypeOsx": "ALLOW",
        "DeviceTypeWeb": "DENY",
        "DeviceTypeIos": "ALLOW",
        "DeviceTypeAndroid": "ALLOW",
        "DeviceTypeChromeOs": "ALLOW",
        "DeviceTypeZeroClient": "ALLOW",
        "DeviceTypeLinux": "DENY"
    },
    "Tenancy": "SHARED",
    "SelfservicePermissions": {
        "RestartWorkspace": "ENABLED",
        "IncreaseVolumeSize": "DISABLED",
        "ChangeComputeType": "DISABLED",
        "SwitchRunningMode": "DISABLED",
        "RebuildWorkspace": "DISABLED"
    }
}
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola direktori WorkSpaces](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DescribeWorkspaceDirectories](#) di Referensi AWS CLI Perintah.

## describe-workspaces-connection-status

Contoh kode berikut menunjukkan cara menggunakan `describe-workspaces-connection-status`.

### AWS CLI

Untuk menggambarkan status koneksi dari sebuah `WorkSpace`

`describe-workspaces-connection-status` Contoh berikut menjelaskan status koneksi yang ditentukan `WorkSpace`.

```
aws workspaces describe-workspaces-connection-status \
```

```
--workspace-ids ws-dk1x zr417
```

Output:

```
{
  "WorkspacesConnectionStatus": [
    {
      "WorkspaceId": "ws-dk1x zr417",
      "ConnectionState": "CONNECTED",
      "ConnectionStateCheckTimestamp": 1662526214.744
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengelola Anda WorkSpaces](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DescribeWorkspacesConnectionStatus](#) di Referensi AWS CLI Perintah.

## describe-workspaces

Contoh kode berikut menunjukkan cara menggunakan `describe-workspaces`.

AWS CLI

Untuk menggambarkan a WorkSpace

`describe-workspaces` Contoh berikut menjelaskan yang ditentukan WorkSpace.

```
aws workspaces describe-workspaces \
  --workspace-ids ws-dk1x zr417
```

Output:

```
{
  "Workspaces": [
    {
      "WorkspaceId": "ws-dk1x zr417",
      "DirectoryId": "d-926722edaf",
      "UserName": "Mary",
      "IpAddress": "172.16.0.175",
    }
  ]
}
```



```

    "State": "STOPPED",
    "BundleId": "wsb-0zsvgp8fc",
    "SubnetId": "subnet-500d5819",
    "ComputerName": "WSAMZN-RBSLTDD9",
    "WorkspaceProperties": {
      "RunningMode": "AUTO_STOP",
      "RunningModeAutoStopTimeoutInMinutes": 60,
      "RootVolumeSizeGib": 80,
      "UserVolumeSizeGib": 10,
      "ComputeTypeName": "VALUE"
    },
    "ModificationStates": []
  }
]
}

```

Untuk informasi selengkapnya, lihat [Mengelola Anda WorkSpaces](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [DescribeWorkspaces](#) di Referensi AWS CLI Perintah.

## migrate-workspace

Contoh kode berikut menunjukkan cara menggunakan `migrate-workspace`.

### AWS CLI

Untuk memigrasikan a Workspace

`migrate-workspace` Contoh berikut memigrasikan yang ditentukan Workspace ke bundel yang ditentukan.

```

aws workspaces migrate-workspace \
  --source-workspace-id ws-dk1x zr417 \
  --bundle-id wsb-j4d ky1gs4

```

Output:

```

{
  "SourceWorkspaceId": "ws-dk1x zr417",
  "TargetWorkspaceId": "ws-x5h1l bkp5"
}

```

Untuk informasi selengkapnya, lihat [Memigrasi a WorkSpace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [MigrateWorkspace](#) di Referensi AWS CLI Perintah.

## **modify-workspace-creation-properties**

Contoh kode berikut menunjukkan cara menggunakan `modify-workspace-creation-properties`.

### AWS CLI

Untuk memodifikasi properti WorkSpace penciptaan direktori

`modify-workspace-creation-properties` Contoh berikut memungkinkan `EnableInternetAccess` properti untuk direktori tertentu. Ini memungkinkan penetapan otomatis alamat IP publik untuk yang WorkSpaces dibuat untuk direktori.

```
aws workspaces modify-workspace-creation-properties \  
  --resource-id d-926722edaf \  
  --workspace-creation-properties EnableInternetAccess=true
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Perbarui detail direktori untuk Anda WorkSpaces](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [ModifyWorkspaceCreationProperties](#) di Referensi AWS CLI Perintah.

## **modify-workspace-properties**

Contoh kode berikut menunjukkan cara menggunakan `modify-workspace-properties`.

### AWS CLI

Untuk memodifikasi mode berjalan dari a WorkSpace

`modify-workspace-properties` Contoh berikut menetapkan mode berjalan dari yang ditentukan WorkSpace untuk `AUTO_STOP`.

```
aws workspaces modify-workspace-properties \  
  --workspace-id ws-dk1xzr417 \  
  --mode-running-mode AUTO_STOP
```

```
--workspace-properties RunningMode=AUTO_STOP
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memodifikasi Workspace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [ModifyWorkspaceProperties](#) di Referensi AWS CLI Perintah.

## **modify-workspace-state**

Contoh kode berikut menunjukkan cara menggunakan `modify-workspace-state`.

### AWS CLI

Untuk memodifikasi keadaan a Workspace

`modify-workspace-state` Contoh berikut menetapkan keadaan yang ditentukan Workspace untuk `ADMIN_MAINTENANCE`.

```
aws workspaces modify-workspace-state \  
  --workspace-id ws-dk1x zr417 \  
  --workspace-state ADMIN_MAINTENANCE
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Workspace pemeliharaan](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [ModifyWorkspaceState](#) di Referensi AWS CLI Perintah.

## **reboot-workspaces**

Contoh kode berikut menunjukkan cara menggunakan `reboot-workspaces`.

### AWS CLI

Untuk me-reboot a Workspace

`reboot-workspaces` Contoh berikut reboot yang ditentukan Workspace.

```
aws workspaces reboot-workspaces \  
  --workspace-id ws-dk1x zr417
```

```
--reboot-workspace-requests ws-dk1xzr417
```

Output:

```
{
  "FailedRequests": []
}
```

Untuk informasi selengkapnya, lihat [Reboot a WorkSpace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [RebootWorkspaces](#) di Referensi AWS CLI Perintah.

## rebuild-workspaces

Contoh kode berikut menunjukkan cara menggunakan `rebuild-workspaces`.

AWS CLI

Untuk membangun kembali a WorkSpace

`rebuild-workspaces` Contoh berikut membangun kembali yang ditentukan. WorkSpace

```
aws workspaces rebuild-workspaces \
  --rebuild-workspace-requests ws-dk1xzr417
```

Output:

```
{
  "FailedRequests": []
}
```

Untuk informasi selengkapnya, lihat [Membangun kembali WorkSpace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [RebuildWorkspaces](#) di Referensi AWS CLI Perintah.

## register-workspace-directory

Contoh kode berikut menunjukkan cara menggunakan `register-workspace-directory`.

## AWS CLI

Untuk mendaftarkan direktori

`register-workspace-directory` Contoh berikut mendaftarkan direktori yang ditentukan untuk digunakan dengan Amazon WorkSpaces.

```
aws workspaces register-workspace-directory \  
  --directory-id d-926722edaf \  
  --no-enable-work-docs
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mendaftarkan direktori dengan WorkSpaces](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [RegisterWorkspaceDirectory](#) di Referensi AWS CLI Perintah.

## restore-workspace

Contoh kode berikut menunjukkan cara menggunakan `restore-workspace`.

### AWS CLI

Untuk mengembalikan a Workspace

`restore-workspace` Contoh berikut mengembalikan yang ditentukan Workspace.

```
aws workspaces restore-workspace \  
  --workspace-id ws-dk1xzi417
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Memulihkan a Workspace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [RestoreWorkspace](#) di Referensi AWS CLI Perintah.

## start-workspaces

Contoh kode berikut menunjukkan cara menggunakan `start-workspaces`.

## AWS CLI

Untuk memulai sebuah AutoStop WorkSpace

`start-workspaces` Contoh berikut memulai yang ditentukan WorkSpace. WorkSpace Harus memiliki mode berjalan `AutoStop`.

```
aws workspaces start-workspaces \  
  --start-workspace-requests WorkspaceId=ws-dk1xzr417
```

Output:

```
{  
  "FailedRequests": []  
}
```

Untuk informasi selengkapnya, lihat [Berhenti dan memulai AutoStop WorkSpace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [StartWorkspaces](#) di Referensi AWS CLI Perintah.

## **stop-workspaces**

Contoh kode berikut menunjukkan cara menggunakan `stop-workspaces`.

## AWS CLI

Untuk menghentikan sebuah AutoStop WorkSpace

`stop-workspaces` Contoh berikut menghentikan yang ditentukan WorkSpace. WorkSpace Harus memiliki mode berjalan `AutoStop`.

```
aws workspaces stop-workspaces \  
  --stop-workspace-requests WorkspaceId=ws-dk1xzr417
```

Output:

```
{  
  "FailedRequests": []  
}
```

Untuk informasi selengkapnya, lihat [Berhenti dan memulai AutoStop WorkSpace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [StopWorkspaces](#) di Referensi AWS CLI Perintah.

## **terminate-workspaces**

Contoh kode berikut menunjukkan cara menggunakan `terminate-workspaces`.

### AWS CLI

Untuk mengakhiri a WorkSpace

`terminate-workspaces` Contoh berikut mengakhiri ruang kerja yang ditentukan.

```
aws workspaces terminate-workspaces \  
  --terminate-workspace-requests ws-dk1x zr417
```

Output:

```
{  
  "FailedRequests": []  
}
```

Untuk informasi selengkapnya, lihat [Menghapus a WorkSpace](#) di Panduan WorkSpaces Administrasi Amazon.

- Untuk detail API, lihat [TerminateWorkspaces](#) di Referensi AWS CLI Perintah.

## Contoh X-Ray menggunakan AWS CLI

Contoh kode berikut menunjukkan kepada Anda bagaimana melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan X-Ray AWS Command Line Interface dengan.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)

## Tindakan

### **batch-traces-get**

Contoh kode berikut menunjukkan cara menggunakan `batch-traces-get`.

## AWS CLI

Untuk mendapatkan daftar jejak

`batch-get-traces` Contoh berikut mengambil daftar jejak yang ditentukan oleh ID. Pelacakan lengkap mencakup dokumen untuk setiap segmen, dikompilasi dari semua dokumen segmen yang diterima dengan ID pelacakan yang sama.

```
aws xray batch-get-traces \
  --trace-ids 1-5d82881a-0a9126e92a73e971eed891b9
```

## Output:

```
{
  "Traces": [
    {
      "Id": "1-5d82881a-0a9126e92a73e971eed891b9",
      "Duration": 0.232,
      "Segments": [
        {
          "Id": "54aff5735b12dd28",
          "Document": "{\"id\":\"54aff5735b12dd28\",\"name\":
\\\"Scorekeep\\\",\\\"start_time\\\":1.568835610432E9,\\\"end_time\\\":1.568835610664E9,
\\\"http\\\":{\\\"request\\\":{\\\"url\\\":\\\"http://scorekeep-env-1.m4fg2pfzpv.us-
east-2.elasticbeanstalk.com/api/user\\\",\\\"method\\\":\\\"POST\\\",\\\"user_agent\\\":
\\\"curl/7.59.0\\\",\\\"client_ip\\\":\\\"52.95.4.28\\\",\\\"x_forwarded_for\\\":true},
\\\"response\\\":{\\\"status\\\":200}},\\\"aws\\\":{\\\"elastic_beanstalk\\\":{\\\"version_label
\\\":\\\"Sample Application-1\\\",\\\"deployment_id\\\":3,\\\"environment_name\\\":\\\"Scorekeep-
env-1\\\"},\\\"ec2\\\":{\\\"availability_zone\\\":\\\"us-east-2b\\\",\\\"instance_id\\\":
```



```

\ "i-0e3cf4d2de0f3f37a\" }, \ "xray\" : { \ "sdk_version\" : \ "1.1.0\" , \ "sdk\" : \ "X-Ray for
Java\" } } , \ "service\" : { \ "runtime\" : \ "OpenJDK 64-Bit Server VM\" , \ "runtime_version
\" : \ "1.8.0_222\" } , \ "trace_id\" : \ "1-5d82881a-0a9126e92a73e971eed891b9\" ,
\ "origin\" : \ "AWS::ElasticBeanstalk::Environment\" , \ "subsegments\" : [ { \ "id\" :
\ "2d6900034ccfe558\" , \ "name\" : \ "DynamoDB\" , \ "start_time\" : 1.568835610658E9 ,
\ "end_time\" : 1.568835610664E9 , \ "http\" : { \ "response\" : { \ "status\" : 200 ,
\ "content_length\" : 61 } } } , \ "aws\" : { \ "table_name\" : \ "scorekeep-user\" , \ "operation\" :
\ "UpdateItem\" , \ "request_id\" : \ "TPEIDNDUROMLP0V17U4A79555NVV4KQNS05AEMVJF66Q9ASUAAJG
\" , \ "resource_names\" : [ \ "scorekeep-user\" ] } , \ "namespace\" : \ "aws\" } } ]
    } ,
    {
        "Id": "0f278b6334c34e6b",
        "Document": "{ \ "id\" : \ "0f278b6334c34e6b\" , \ "name\" :
\ "DynamoDB\" , \ "start_time\" : 1.568835610658E9 , \ "end_time\" : 1.568835610664E9 ,
\ "parent_id\" : \ "2d6900034ccfe558\" , \ "inferred\" : true , \ "http\" : { \ "response
\" : { \ "status\" : 200 , \ "content_length\" : 61 } } } , \ "aws\" : { \ "table_name
\" : \ "scorekeep-user\" , \ "operation\" : \ "UpdateItem\" , \ "request_id\" :
\ "TPEIDNDUROMLP0V17U4A79555NVV4KQNS05AEMVJF66Q9ASUAAJG\" , \ "resource_names\" :
[ \ "scorekeep-user\" ] } , \ "trace_id\" : \ "1-5d82881a-0a9126e92a73e971eed891b9\" , \ "origin
\" : \ "AWS::DynamoDB::Table\" } }
    }
  ]
}
  ],
  "UnprocessedTraceIds": []
}

```

Untuk informasi selengkapnya, lihat [Menggunakan AWS X-Ray API dengan AWS CLI](#) di Panduan Pengembang AWS X-Ray.

- Untuk detail API, lihat [BatchTracesGet](#) di Referensi AWS CLI Perintah.

## create-group

Contoh kode berikut menunjukkan cara menggunakan `create-group`.

### AWS CLI

Untuk membuat grup

`create-group` Contoh berikut membuat sumber daya grup bernama `AdminGroup`. Grup mendapatkan ekspresi filter yang mendefinisikan kriteria grup sebagai segmen yang terkait dengan layanan tertentu yang menyebabkan kesalahan atau kesalahan.

```
aws xray create-group \  
  --group-name "AdminGroup" \  
  --filter-expression "service(\"mydomain.com\") {fault OR error}"
```

Output:

```
{  
  "GroupName": "AdminGroup",  
  "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789",  
  "FilterExpression": "service(\"mydomain.com\") {fault OR error}"  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan](#) Pengembang AWS X-Ray.

- Untuk detail API, lihat [CreateGroup](#) di Referensi AWS CLI Perintah.

## create-sampling-rule

Contoh kode berikut menunjukkan cara menggunakan `create-sampling-rule`.

### AWS CLI

Untuk membuat aturan pengambilan sampel

`create-sampling-rule` Contoh berikut membuat aturan untuk mengontrol perilaku pengambilan sampel untuk aplikasi berinstrumen. Aturan disediakan oleh file JSON. Mayoritas bidang aturan pengambilan sampel diperlukan untuk membuat aturan.

```
aws xray create-sampling-rule \  
  --cli-input-json file://9000-base-scorekeep.json
```

Isi dari `9000-base-scorekeep.json`:

```
{  
  "SamplingRule": {  
    "RuleName": "base-scorekeep",  
    "ResourceARN": "*",  
    "Priority": 9000,  
    "FixedRate": 0.1,  
  }  
}
```

```
    "ReservoirSize": 5,
    "ServiceName": "Scorekeep",
    "ServiceType": "*",
    "Host": "*",
    "HTTPMethod": "*",
    "URLPath": "*",
    "Version": 1
  }
}
```

Output:

```
{
  "SamplingRuleRecord": {
    "SamplingRule": {
      "RuleName": "base-scorekeep",
      "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/base-scorekeep",
      "ResourceARN": "*",
      "Priority": 9000,
      "FixedRate": 0.1,
      "ReservoirSize": 5,
      "ServiceName": "Scorekeep",
      "ServiceType": "*",
      "Host": "*",
      "HTTPMethod": "*",
      "URLPath": "*",
      "Version": 1,
      "Attributes": {}
    },
    "CreatedAt": 1530574410.0,
    "ModifiedAt": 1530574410.0
  }
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan Pengembang AWS X-Ray](#).

- Untuk detail API, lihat [CreateSamplingRule](#) di Referensi AWS CLI Perintah.

## delete-group

Contoh kode berikut menunjukkan cara menggunakan `delete-group`.

## AWS CLI

Untuk menghapus grup

`delete-group` Contoh berikut menghapus sumber daya grup yang ditentukan.

```
aws xray delete-group \  
  --group-name "AdminGroup" \  
  --group-arn "arn:aws:xray:us-east-2:123456789012:group/AdminGroup/123456789"
```

Perintah ini tidak menghasilkan output.

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan Pengembang AWS X-Ray](#).

- Untuk detail API, lihat [DeleteGroup](#) di Referensi AWS CLI Perintah.

## `delete-sampling-rule`

Contoh kode berikut menunjukkan cara menggunakan `delete-sampling-rule`.

## AWS CLI

Untuk menghapus aturan pengambilan sampel

`delete-sampling-rule` Contoh berikut menghapus aturan sampling yang ditentukan. Anda dapat menentukan grup dengan menggunakan nama grup atau grup ARN.

```
aws xray delete-sampling-rule \  
  --rule-name polling-scorekeep
```

Output:

```
{  
  "SamplingRuleRecord": {  
    "SamplingRule": {  
      "RuleName": "polling-scorekeep",  
      "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/polling-  
scorekeep",  
      "ResourceARN": "*",  
      "Priority": 5000,  
    }  
  }  
}
```

```
    "FixedRate": 0.003,  
    "ReservoirSize": 0,  
    "ServiceName": "Scorekeep",  
    "ServiceType": "*",  
    "Host": "*",  
    "HTTPMethod": "GET",  
    "URLPath": "/api/state/*",  
    "Version": 1,  
    "Attributes": {}  
  },  
  "CreatedAt": 1530574399.0,  
  "ModifiedAt": 1530574399.0  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan](#) Pengembang AWS X-Ray.

- Untuk detail API, lihat [DeleteSamplingRule](#) di Referensi AWS CLI Perintah.

## get-encryption-config

Contoh kode berikut menunjukkan cara menggunakan `get-encryption-config`.

### AWS CLI

Untuk mengambil konfigurasi enkripsi

`get-encryption-config` Contoh berikut mengambil konfigurasi enkripsi saat ini untuk data AWS X-Ray Anda.

```
aws xray get-encryption-config
```

Output:

```
{  
  "EncryptionConfig": {  
    "KeyId": "ae4aa6d49-a4d8-9df9-a475-4ff6d7898456",  
    "Status": "ACTIVE",  
    "Type": "NONE"  
  }  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan](#) Pengembang AWS X-Ray.

- Untuk detail API, lihat [GetEncryptionConfig](#) di Referensi AWS CLI Perintah.

## get-group

Contoh kode berikut menunjukkan cara menggunakan `get-group`.

### AWS CLI

Untuk mengambil grup

`get-group` Contoh berikut menampilkan rincian untuk sumber daya grup tertentu. Rinciannya mencakup nama grup, ARN grup, dan ekspresi filter yang menentukan kriteria untuk grup tersebut. Grup juga dapat diambil oleh ARN.

```
aws xray get-group \  
  --group-name "AdminGroup"
```

Output:

```
{  
  "Group": [  
    {  
      "GroupName": "AdminGroup",  
      "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/  
AdminGroup/123456789",  
      "FilterExpression": "service(\"mydomain.com\") {fault OR error}"  
    }  
  ]  
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan](#) Pengembang AWS X-Ray.

- Untuk detail API, lihat [GetGroup](#) di Referensi AWS CLI Perintah.

## get-groups

Contoh kode berikut menunjukkan cara menggunakan `get-groups`.

## AWS CLI

Untuk mengambil semua grup

Contoh berikut menampilkan rincian untuk semua grup aktif.

```
aws xray get-groups
```

Output:

```
{
  "Groups": [
    {
      "GroupName": "AdminGroup",
      "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789",
      "FilterExpression": "service(\"example.com\") {fault OR error}"
    },
    {
      "GroupName": "SDETGroup",
      "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/SDETGroup/987654321",
      "FilterExpression": "responsetime > 2"
    }
  ]
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan Pengembang AWS X-Ray](#).

- Untuk detail API, lihat [GetGroups](#) di Referensi AWS CLI Perintah.

## get-sampling-rules

Contoh kode berikut menunjukkan cara menggunakan `get-sampling-rules`.

### AWS CLI

Untuk mengambil semua aturan pengambilan sampel

`get-sampling-rules` Contoh berikut menampilkan rincian untuk semua aturan sampling yang tersedia. :

```
aws xray get-sampling-rules
```

**Output:**

```
{
  "SamplingRuleRecords": [
    {
      "SamplingRule": {
        "RuleName": "Default",
        "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/Default",
        "ResourceARN": "*",
        "Priority": 10000,
        "FixedRate": 0.01,
        "ReservoirSize": 0,
        "ServiceName": "*",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "*",
        "URLPath": "*",
        "Version": 1,
        "Attributes": {}
      },
      "CreatedAt": 0.0,
      "ModifiedAt": 1530558121.0
    },
    {
      "SamplingRule": {
        "RuleName": "base-scorekeep",
        "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/base-scorekeep",
        "ResourceARN": "*",
        "Priority": 9000,
        "FixedRate": 0.1,
        "ReservoirSize": 2,
        "ServiceName": "Scorekeep",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "*",
        "URLPath": "*",
        "Version": 1,
        "Attributes": {}
      },
      "CreatedAt": 1530573954.0,
      "ModifiedAt": 1530920505.0
    }
  ]
}
```



```

    },
    {
      "SamplingRule": {
        "RuleName": "polling-scorekeep",
        "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/polling-
scorekeep",
        "ResourceARN": "*",
        "Priority": 5000,
        "FixedRate": 0.003,
        "ReservoirSize": 0,
        "ServiceName": "Scorekeep",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "GET",
        "URLPath": "/api/state/*",
        "Version": 1,
        "Attributes": {}
      },
      "CreatedAt": 1530918163.0,
      "ModifiedAt": 1530918163.0
    }
  ]
}

```

Untuk informasi selengkapnya, lihat [Menggunakan Aturan Pengambilan Sampel dengan X-Ray API](#) di Panduan Pengembang AWS X-Ray.

- Untuk detail API, lihat [GetSamplingRules](#) di Referensi AWS CLI Perintah.

## get-sampling-targets

Contoh kode berikut menunjukkan cara menggunakan `get-sampling-targets`.

### AWS CLI

Untuk meminta kuota sampling

`get-sampling-targets` Contoh berikut meminta kuota sampling untuk aturan yang digunakan layanan untuk sampel permintaan. Tanggapan dari AWS X-Ray termasuk kuota yang dapat digunakan sebagai pengganti meminjam dari reservoir.

```
aws xray get-sampling-targets \
```

```
--sampling-statistics-documents '[ { "RuleName": "base-scorekeep", "ClientID":
"ABCDEF1234567890ABCDEF10", "Timestamp": "2018-07-07T00:20:06", "RequestCount": 110,
"SampledCount": 20, "BorrowCount": 10 }, { "RuleName": "polling-scorekeep", 31,
"BorrowCount": 0 } ]'
```

## Output:

```
{
  "SamplingTargetDocuments": [
    {
      "RuleName": "base-scorekeep",
      "FixedRate": 0.1,
      "ReservoirQuota": 2,
      "ReservoirQuotaTTL": 1530923107.0,
      "Interval": 10
    },
    {
      "RuleName": "polling-scorekeep",
      "FixedRate": 0.003,
      "ReservoirQuota": 0,
      "ReservoirQuotaTTL": 1530923107.0,
      "Interval": 10
    }
  ],
  "LastRuleModification": 1530920505.0,
  "UnprocessedStatistics": []
}
```

Untuk informasi selengkapnya, lihat [Menggunakan Aturan Pengambilan Sampel dengan X-Ray API](#) di Panduan Pengembang AWS X-Ray.

- Untuk detail API, lihat [GetSamplingTargets](#) di Referensi AWS CLI Perintah.

## get-service-graph

Contoh kode berikut menunjukkan cara menggunakan `get-service-graph`.

### AWS CLI

Untuk mendapatkan grafik layanan

Contoh berikut menampilkan dokumen dalam jangka waktu tertentu yang menjelaskan layanan yang memproses permintaan masuk, dan layanan hilir yang mereka panggil sebagai hasilnya. :

```
aws xray get-service-graph \  
  --start-time 1568835392.0  
  --end-time 1568835446.0
```

**Output:**

```
{  
  "Services": [  
    {  
      "ReferenceId": 0,  
      "Name": "Scorekeep",  
      "Names": [  
        "Scorekeep"  
      ],  
      "Root": true,  
      "Type": "AWS::ElasticBeanstalk::Environment",  
      "State": "active",  
      "StartTime": 1568835392.0,  
      "EndTime": 1568835446.0,  
      "Edges": [  
        {  
          "ReferenceId": 1,  
          "StartTime": 1568835392.0,  
          "EndTime": 1568835446.0,  
          "SummaryStatistics": {  
            "OkCount": 14,  
            "ErrorStatistics": {  
              "ThrottleCount": 0,  
              "OtherCount": 0,  
              "TotalCount": 0  
            },  
            "FaultStatistics": {  
              "OtherCount": 0,  
              "TotalCount": 0  
            },  
            "TotalCount": 14,  
            "TotalResponseTime": 0.13  
          },  
          "ResponseTimeHistogram": [  
            {  
              "Value": 0.008,  
              "Count": 1  
            }  
          ],  
        }  
      ],  
    }  
  ]  
}
```

```
        {
            "Value": 0.005,
            "Count": 7
        },
        {
            "Value": 0.009,
            "Count": 1
        },
        {
            "Value": 0.021,
            "Count": 1
        },
        {
            "Value": 0.038,
            "Count": 1
        },
        {
            "Value": 0.007,
            "Count": 1
        },
        {
            "Value": 0.006,
            "Count": 2
        }
    ],
    "Aliases": []
},
... TRUNCATED FOR BREVITY ...

]
}
],
"StartTime": 1568835392.0,
"EndTime": 1568835446.0,
"ContainsOldGroupVersions": false
}
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS X-Ray API dengan AWS CLI](#) di Panduan Pengembang AWS X-Ray.

- Untuk detail API, lihat [GetServiceGraph](#) di Referensi AWS CLI Perintah.

## get-trace-summaries

Contoh kode berikut menunjukkan cara menggunakan `get-trace-summaries`.

### AWS CLI

Untuk mendapatkan ringkasan jejak

`get-trace-summaries` Contoh berikut mengambil ID dan metadata untuk jejak yang tersedia dalam jangka waktu tertentu.

```
aws xray get-trace-summaries \  
  --start-time 1568835392.0 \  
  --end-time 1568835446.0
```

Output:

```
[  
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/move/  
VSAE93HF/GSSD2NTB/DP0PCC09",  
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/move/  
GCQ2B35P/FREELDFT/4LRE643M",  
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/game/  
VSAE93HF/GSSD2NTB/starttime/1568835513",  
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/  
move/4MQNA5NN/L99KK2RF/null"  
]
```

Untuk informasi selengkapnya, lihat [Menggunakan AWS X-Ray API dengan AWS CLI](#) di Panduan Pengembang AWS X-Ray.

- Untuk detail API, lihat [GetTraceSummaries](#) di Referensi AWS CLI Perintah.

## put-encryption-config

Contoh kode berikut menunjukkan cara menggunakan `put-encryption-config`.

### AWS CLI

Untuk memperbarui konfigurasi enkripsi

Berikut input-encryption-config``example updates the encryption configuration for AWS X-Ray data to use the default AWS managed KMS key ``aws/xray.

```
aws xray put-encryption-config \
  --type KMS \
  --key-id alias/aws/xray
```

Output:

```
{
  "EncryptionConfig": {
    "KeyId": "arn:aws:kms:us-west-2:123456789012:key/c234g4e8-39e9-4gb0-84e2-
b0ea215cbba5",
    "Status": "UPDATING",
    "Type": "KMS"
  }
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan](#) Pengembang AWS X-Ray.

- Untuk detail API, lihat [PutEncryptionConfig](#) di Referensi AWS CLI Perintah.

## put-trace-segments

Contoh kode berikut menunjukkan cara menggunakanput-trace-segments.

### AWS CLI

Untuk mengunggah segmen

put-trace-segmentsContoh berikut mengunggah dokumen segmen ke AWS X-Ray. Dokumen segmen dikonsumsi sebagai daftar dokumen segmen JSON.

```
aws xray put-trace-segments \
  --trace-segment-documents '{"id":"20312a0e2b8809f4","name
":"DynamoDB","trace_id":"1-5832862d-a43aafded3334a971fe312db",
"start_time":1.479706157195E9,"end_time":1.479706157202E9,"parent_id":
"79736b962fe3239e","http":{"response":{"content_length":60,"status
":200}},"inferred":true,"aws":{"consistent_read":false,"table_name
":"scorekeep-session-xray"},"operation":"GetItem","request_id":
```

```
\ "SCAU230M6M8F038UASGC7785ARVV4KQNS05AEMVJF66Q9ASUAAJG\", \"resource_names\":
[\"scorekeep-session-xray\"]}, \"origin\": \"AWS::DynamoDB::Table\"}"
```

Output:

```
{
  "UnprocessedTraceSegments": []
}
```

Untuk informasi selengkapnya, lihat [Mengirim Data Jejak ke AWS X-Ray](#) di Panduan Pengembang AWS X-Ray.

- Untuk detail API, lihat [PutTraceSegments](#) di Referensi AWS CLI Perintah.

## update-group

Contoh kode berikut menunjukkan cara menggunakan `update-group`.

### AWS CLI

Untuk memperbarui grup

`update-group` Contoh berikut memperbarui kriteria yang digunakan untuk menerima jejak ke dalam grup bernama `AdminGroup`. Anda dapat menentukan grup yang diinginkan dengan menggunakan nama grup atau grup ARN.

```
aws xray update-group \
  --group-name "AdminGroup" \
  --group-arn "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789" \
  --filter-expression "service(\"mydomain.com\") {fault}"
```

Output:

```
{
  "GroupName": "AdminGroup",
  "GroupARN": "arn:aws:xray:us-east-2:123456789012:group/AdminGroup/123456789",
  "FilterExpression": "service(\"mydomain.com\") {fault}"
}
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan Pengembang AWS X-Ray](#).

- Untuk detail API, lihat [UpdateGroup](#) di Referensi AWS CLI Perintah.

## update-sampling-rule

Contoh kode berikut menunjukkan cara menggunakan `update-sampling-rule`.

### AWS CLI

Untuk memperbarui aturan pengambilan sampel

`update-sampling-rule` Contoh berikut memodifikasi konfigurasi aturan sampling ini. Aturan dikonsumsi dari file JSON. Hanya bidang yang diperbarui yang diperlukan.

```
aws xray update-sampling-rule \  
  --cli-input-json file://1000-default.json
```

Isi dari `1000-default.json`:

```
{  
  "SamplingRuleUpdate": {  
    "RuleName": "Default",  
    "FixedRate": 0.01,  
    "ReservoirSize": 0  
  }  
}
```

Output:

```
{  
  "SamplingRuleRecords": [  
    {  
      "SamplingRule": {  
        "RuleName": "Default",  
        "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/  
Default",  
        "ResourceARN": "*",  
        "Priority": 10000,  
        "FixedRate": 0.01,  
        "ReservoirSize": 0,  
        "ServiceName": "*",  
        "ServiceType": "*",  

```



```
        "Host": "*",
        "HTTPMethod": "*",
        "URLPath": "*",
        "Version": 1,
        "Attributes": {}
    },
    "CreatedAt": 0.0,
    "ModifiedAt": 1529959993.0
}
]
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi Pengaturan Pengambilan Sampel, Grup, dan Enkripsi dengan AWS X-Ray API di Panduan](#) Pengembang AWS X-Ray.

- Untuk detail API, lihat [UpdateSamplingRule](#) di Referensi AWS CLI Perintah.

## AWS CLI dengan contoh kode skrip Bash

Contoh kode dalam topik ini menunjukkan cara menggunakan skrip AWS Command Line Interface with Bash dengan AWS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Contoh lintas layanan adalah contoh aplikasi yang bekerja di beberapa Layanan AWS.

Contoh

- [Tindakan dan skenario menggunakan AWS CLI skrip Bash](#)

## Tindakan dan skenario menggunakan AWS CLI skrip Bash

Contoh kode berikut menunjukkan bagaimana melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan skrip AWS Command Line Interface with Bash with Layanan AWS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

## Layanan

- [Contoh AWS CLI DynamoDB menggunakan dengan skrip Bash](#)
- [HealthImaging contoh menggunakan AWS CLI dengan skrip Bash](#)
- [Contoh IAM menggunakan skrip AWS CLI Bash](#)
- [Contoh Amazon S3 menggunakan skrip AWS CLI Bash](#)
- [AWS STS contoh menggunakan AWS CLI dengan skrip Bash](#)

## Contoh AWS CLI DynamoDB menggunakan dengan skrip Bash

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan skrip AWS Command Line Interface with Bash dengan DynamoDB.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

## Topik

- [Tindakan](#)
- [Skenario](#)

## Tindakan

### **BatchGetItem**

Contoh kode berikut menunjukkan cara menggunakan `BatchGetItem`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```
#####
# function dynamodb_batch_get_item
#
# This function gets a batch of items from a DynamoDB table.
#
# Parameters:
#     -i item -- Path to json file containing the keys of the items to get.
#
# Returns:
#     The items as json output.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_batch_get_item() {
    local item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_batch_get_item"
        echo "Get a batch of items from a DynamoDB table."
        echo " -i item -- Path to json file containing the keys of the items to get."
        echo ""
    }

    while getopt "i:h" option; do
        case "${option}" in
            i) item="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

response=$(aws dynamodb batch-get-item \
    --request-items file://"${item}")
local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-get-item operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```
#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Untuk detail API, lihat [BatchGetItem](#) di Referensi AWS CLI Perintah.

## BatchWriteItem

Contoh kode berikut menunjukkan cara menggunakan `BatchWriteItem`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```
#####
# function dynamodb_batch_write_item
#
# This function writes a batch of items into a DynamoDB table.
#
# Parameters:
#     -i item -- Path to json file containing the items to write.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_batch_write_item() {
    local item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_batch_write_item"
        echo "Write a batch of items into a DynamoDB table."
        echo " -i item -- Path to json file containing the items to write."
        echo ""
    }
    while getopt "i:h" option; do
        case "${option}" in
            i) item="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
```

```

        usage
        return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho "    item:        $item"
iecho ""

response=$(aws dynamodb batch-write-item \
    --request-items file://"${item}")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
fi

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then

```

```

    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho " One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
  elif [ "$err_code" == 255 ]; then

```



```

    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Untuk detail API, lihat [BatchWriteItem](#) di Referensi AWS CLI Perintah.

## CreateTable

Contoh kode berikut menunjukkan cara menggunakan CreateTable.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_create_table
#
# This function creates an Amazon DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table to create.
#     -a attribute_definitions -- JSON file path of a list of attributes and their
types.
#     -k key_schema -- JSON file path of a list of attributes and their key types.
#     -p provisioned_throughput -- Provisioned throughput settings for the table.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_create_table() {
    local table_name attribute_definitions key_schema provisioned_throughput response
    local option OPTARG # Required to use getopt command in a function.

#####

```

```
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_create_table"
    echo "Creates an Amazon DynamoDB table."
    echo " -n table_name -- The name of the table to create."
    echo " -a attribute_definitions -- JSON file path of a list of attributes and
their types."
    echo " -k key_schema -- JSON file path of a list of attributes and their key
types."
    echo " -p provisioned_throughput -- Provisioned throughput settings for the
table."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:a:k:p:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        a) attribute_definitions="${OPTARG}" ;;
        k) key_schema="${OPTARG}" ;;
        p) provisioned_throughput="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$attribute_definitions" ]]; then
    errecho "ERROR: You must provide an attribute definitions json file path the -a
parameter."
fi
```

```
usage
return 1
fi

if [[ -z "$key_schema" ]]; then
    errecho "ERROR: You must provide a key schema json file path the -k parameter."
    usage
    return 1
fi

if [[ -z "$provisioned_throughput" ]]; then
    errecho "ERROR: You must provide a provisioned throughput json file path the -p
parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  attribute_definitions:  $attribute_definitions"
iecho "  key_schema:  $key_schema"
iecho "  provisioned_throughput:  $provisioned_throughput"
iecho ""

response=$(aws dynamodb create-table \
  --table-name "$table_name" \
  --attribute-definitions file://"$attribute_definitions" \
  --key-schema file://"$key_schema" \
  --provisioned-throughput "$provisioned_throughput")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-table operation failed.$response"
    return 1
fi

return 0
}
```

Fungsi utilitas yang digunakan dalam contoh ini.

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    fi
}
```

```

elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Untuk detail API, lihat [CreateTable](#) di Referensi AWS CLI Perintah.

## DeleteItem

Contoh kode berikut menunjukkan cara menggunakan `DeleteItem`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_delete_item
#
# This function deletes an item from a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table.
#     -k keys        -- Path to json file containing the keys that identify the item to
#                       delete.
#
# Returns:
#     0 - If successful.

```

```

#      1 - If it fails.
#####
function dynamodb_delete_item() {
    local table_name keys response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_delete_item"
        echo "Delete an item from a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -k keys -- Path to json file containing the keys that identify the item
to delete."
        echo ""
    }
    while getopt "n:k:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi

    if [[ -z "$keys" ]]; then
        errecho "ERROR: You must provide a keys json file path the -k parameter."
        usage
        return 1
    fi
}

```

```

fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:    $keys"
iecho ""

response=$(aws dynamodb delete-item \
  --table-name "$table_name" \
  --key file://"$keys")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-item operation failed.$response"
  return 1
fi

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####

```

```

function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Untuk detail API, lihat [Deleteltem](#) di Referensi AWS CLI Perintah.



## DeleteTable

Contoh kode berikut menunjukkan cara menggunakan DeleteTable.

### AWS CLI dengan skrip Bash

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```
#####
# function dynamodb_delete_table
#
# This function deletes a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_delete_table() {
    local table_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function dynamodb_delete_table"
        echo "Deletes an Amazon DynamoDB table."
        echo " -n table_name  -- The name of the table to delete."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho ""

response=$(aws dynamodb delete-table \
    --table-name "$table_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-table operation failed.$response"
    return 1
fi

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####

```

```

function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho " One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then

```

```

    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Untuk detail API, lihat [DeleteTable](#) di Referensi AWS CLI Perintah.

## DescribeTable

Contoh kode berikut menunjukkan cara menggunakan `DescribeTable`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_describe_table
#
# This function returns the status of a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#
# Response:
#     - TableStatus:
#     And:
#     0 - Table is active.
#     1 - If it fails.
#####
function dynamodb_describe_table {
    local table_name
    local option OPTARG # Required to use getopt command in a function.

#####

```

```
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_describe_table"
    echo "Describe the status of a DynamoDB table."
    echo "  -n table_name  -- The name of the table."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

local table_status
table_status=$(
    aws dynamodb describe-table \
        --table-name "$table_name" \
        --output text \
        --query 'Table.TableStatus'
)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log "$error_code"
```

```

    errecho "ERROR: AWS reports describe-table operation failed.$table_status"
    return 1
fi

echo "$table_status"

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    fi
}

```

```

elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Untuk detail API, lihat [DescribeTable](#) di Referensi AWS CLI Perintah.

## GetItem

Contoh kode berikut menunjukkan cara menggunakan `GetItem`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_get_item
#
# This function gets an item from a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k keys -- Path to json file containing the keys that identify the item to
get.
#     [-q query] -- Optional JMESPath query expression.
#
# Returns:

```

```

#       The item as text output.
# And:
#       0 - If successful.
#       1 - If it fails.
#####
function dynamodb_get_item() {
    local table_name keys query response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_get_item"
        echo "Get an item from a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -k keys -- Path to json file containing the keys that identify the item
to get."
        echo " [-q query] -- Optional JMESPath query expression."
        echo ""
    }
    query=""
    while getopt "n:k:q:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            q) query="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi
}

```



```
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

if [[ -n "$query" ]]; then
    response=$(aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://"keys" \
        --output text \
        --query "$query")
else
    response=$(
        aws dynamodb get-item \
            --table-name "$table_name" \
            --key file://"keys" \
            --output text
    )
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports get-item operation failed.$response"
    return 1
fi

if [[ -n "$query" ]]; then
    echo "$response" | sed "/^\t/s/\t//1" # Remove initial tab that the JMSEPath
query inserts on some strings.
else
    echo "$response"
fi

return 0
}
```

Fungsi utilitas yang digunakan dalam contoh ini.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi
}
```

```

    return 0
}

```

- Untuk detail API, lihat [GetItem](#) di Referensi AWS CLI Perintah.

## ListTables

Contoh kode berikut menunjukkan cara menggunakan `ListTables`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_list_tables
#
# This function lists all the tables in a DynamoDB.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_list_tables() {
    response=$(aws dynamodb list-tables \
        --output text \
        --query "TableNames")

    local error_code=${?}

    if [[ $error_code -ne 0 ]]; then
        aws_cli_error_log $error_code
        errecho "ERROR: AWS reports batch-write-item operation failed.$response"
        return 1
    fi

    echo "$response" | tr -s "[:space:]" "\n"
}

```

```

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    }
}

```

```

elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Untuk detail API, lihat [ListTables](#) di Referensi AWS CLI Perintah.

## PutItem

Contoh kode berikut menunjukkan cara menggunakan PutItem.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_put_item
#
# This function puts an item into a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -i item -- Path to json file containing the item values.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_put_item() {
    local table_name item response
    local option OPTARG # Required to use getopt command in a function.

#####

```

```
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_put_item"
    echo "Put an item into a DynamoDB table."
    echo " -n table_name -- The name of the table."
    echo " -i item -- Path to json file containing the item values."
    echo ""
}

while getopts "n:i:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        i) item="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho "    item:        $item"
iecho ""
iecho ""
```

```

response=$(aws dynamodb put-item \
  --table-name "$table_name" \
  --item file://"item")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports put-item operation failed.$response"
  return 1
fi

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()

```

```

#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Untuk detail API, lihat [PutItem](#) di Referensi AWS CLI Perintah.

## Query

Contoh kode berikut menunjukkan cara menggunakan Query.



## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```
#####
# function dynamodb_query
#
# This function queries a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k key_condition_expression -- The key condition expression.
#     -a attribute_names -- Path to JSON file containing the attribute names.
#     -v attribute_values -- Path to JSON file containing the attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_query() {
    local table_name key_condition_expression attribute_names attribute_values
    projection_expression response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    # #####
    function usage() {
        echo "function dynamodb_query"
        echo "Query a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -k key_condition_expression -- The key condition expression."
        echo " -a attribute_names -- Path to JSON file containing the attribute names."
        echo " -v attribute_values -- Path to JSON file containing the attribute
values."
```

```
    echo " [-p projection_expression] -- Optional projection expression."
    echo ""
}

while getopts "n:k:a:v:p:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        k) key_condition_expression="${OPTARG}" ;;
        a) attribute_names="${OPTARG}" ;;
        v) attribute_values="${OPTARG}" ;;
        p) projection_expression="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$key_condition_expression" ]]; then
    errecho "ERROR: You must provide a key condition expression with the -k
parameter."
    usage
    return 1
fi

if [[ -z "$attribute_names" ]]; then
    errecho "ERROR: You must provide a attribute names with the -a parameter."
    usage
    return 1
fi

if [[ -z "$attribute_values" ]]; then
```

```

    errecho "ERROR: You must provide a attribute values with the -v parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"${attribute_names}" \
        --expression-attribute-values file://"${attribute_values}")
else
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"${attribute_names}" \
        --expression-attribute-values file://"${attribute_values}" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports query operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```

}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Untuk detail API, lihat [Kueri](#) di Referensi Perintah AWS CLI .

## Scan

Contoh kode berikut menunjukkan cara menggunakan Scan.

### AWS CLI dengan skrip Bash

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```
#####
# function dynamodb_scan
#
# This function scans a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -f filter_expression -- The filter expression.
#     -a expression_attribute_names -- Path to JSON file containing the expression
#     attribute names.
#     -v expression_attribute_values -- Path to JSON file containing the
#     expression attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_scan() {
    local table_name filter_expression expression_attribute_names
    expression_attribute_values projection_expression response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_scan"
        echo "Scan a DynamoDB table."
    }
}
```

```
    echo " -n table_name -- The name of the table."
    echo " -f filter_expression -- The filter expression."
    echo " -a expression_attribute_names -- Path to JSON file containing the
expression attribute names."
    echo " -v expression_attribute_values -- Path to JSON file containing the
expression attribute values."
    echo " [-p projection_expression] -- Optional projection expression."
    echo ""
}

while getopts "n:f:a:v:p:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        f) filter_expression="${OPTARG}" ;;
        a) expression_attribute_names="${OPTARG}" ;;
        v) expression_attribute_values="${OPTARG}" ;;
        p) projection_expression="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$filter_expression" ]]; then
    errecho "ERROR: You must provide a filter expression with the -f parameter."
    usage
    return 1
fi

if [[ -z "$expression_attribute_names" ]]; then
```

```
    errecho "ERROR: You must provide expression attribute names with the -a
parameter."
    usage
    return 1
fi

if [[ -z "$expression_attribute_values" ]]; then
    errecho "ERROR: You must provide expression attribute values with the -v
parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"${expression_attribute_names}" \
        --expression-attribute-values file://"${expression_attribute_values}")
else
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"${expression_attribute_names}" \
        --expression-attribute-values file://"${expression_attribute_values}" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports scan operation failed.$response"
    return 1
fi

echo "$response"

return 0
}
```

Fungsi utilitas yang digunakan dalam contoh ini.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi
}
```



```

    return 0
}

```

- Untuk detail API, lihat [Scan](#) in Referensi Perintah AWS CLI .

## UpdateItem

Contoh kode berikut menunjukkan cara menggunakan UpdateItem.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

```

#####
# function dynamodb_update_item
#
# This function updates an item in a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k keys -- Path to json file containing the keys that identify the item to
#     update.
#     -e update expression -- An expression that defines one or more attributes
#     to be updated.
#     -v values -- Path to json file containing the update values.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_update_item() {
    local table_name keys update_expression values response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation

```

```
#####  
function usage() {  
    echo "function dynamodb_update_item"  
    echo "Update an item in a DynamoDB table."  
    echo " -n table_name -- The name of the table."  
    echo " -k keys -- Path to json file containing the keys that identify the item  
to update."  
    echo " -e update expression -- An expression that defines one or more  
attributes to be updated."  
    echo " -v values -- Path to json file containing the update values."  
    echo ""  
}  
  
while getopts "n:k:e:v:h" option; do  
    case "${option}" in  
        n) table_name="${OPTARG}" ;;  
        k) keys="${OPTARG}" ;;  
        e) update_expression="${OPTARG}" ;;  
        v) values="${OPTARG}" ;;  
        h)  
            usage  
            return 0  
            ;;  
        \?)  
            echo "Invalid parameter"  
            usage  
            return 1  
            ;;  
    esac  
done  
export OPTIND=1  
  
if [[ -z "$table_name" ]]; then  
    errecho "ERROR: You must provide a table name with the -n parameter."  
    usage  
    return 1  
fi  
  
if [[ -z "$keys" ]]; then  
    errecho "ERROR: You must provide a keys json file path the -k parameter."  
    usage  
    return 1  
fi  
if [[ -z "$update_expression" ]]; then
```

```

    errecho "ERROR: You must provide an update expression with the -e parameter."
    usage
    return 1
fi

if [[ -z "$values" ]]; then
    errecho "ERROR: You must provide a values json file path the -v parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:        $keys"
iecho "  update_expression:  $update_expression"
iecho "  values:      $values"

response=$(aws dynamodb update-item \
  --table-name "$table_name" \
  --key file://" $keys" \
  --update-expression "$update_expression" \
  --expression-attribute-values file://" $values")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports update-item operation failed.$response"
    return 1
fi

return 0
}

```

Fungsi utilitas yang digunakan dalam contoh ini.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.

```

```
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho " One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
  fi
}
```

```
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}
```

- Untuk detail API, lihat [UpdateItem](#) di Referensi AWS CLI Perintah.

## Skenario

Memulai tabel, item, dan kueri

Contoh kode berikut ini menunjukkan cara:

- Buat tabel yang dapat menyimpan data film.
- Masukkan, dapatkan, dan perbarui satu film dalam tabel tersebut.
- Tulis data film ke tabel dari file JSON sampel.
- Kueri untuk film yang dirilis pada tahun tertentu.
- Pindai film yang dirilis dalam suatu rentang tahun.
- Hapus film dari tabel, lalu hapus tabel tersebut.

## AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

Skenario mulai DynamoDB.

```
#####
# function dynamodb_getting_started_movies
#
```

```
# Scenario to create an Amazon DynamoDB table and perform a series of operations on
the table.
#
# Returns:
#     0 - If successful.
#     1 - If an error occurred.
#####
function dynamodb_getting_started_movies() {

    source ./dynamodb_operations.sh

    key_schema_json_file="dynamodb_key_schema.json"
    attribute_definitions_json_file="dynamodb_attr_def.json"
    item_json_file="movie_item.json"
    key_json_file="movie_key.json"
    batch_json_file="batch.json"
    attribute_names_json_file="attribute_names.json"
    attributes_values_json_file="attribute_values.json"

    echo_repeat "*" 88
    echo
    echo "Welcome to the Amazon DynamoDB getting started demo."
    echo
    echo_repeat "*" 88
    echo

    local table_name
    echo -n "Enter a name for a new DynamoDB table: "
    get_input
    table_name=$get_input_result

    local provisioned_throughput="ReadCapacityUnits=5,WriteCapacityUnits=5"

    echo '['
    {"AttributeName": "year", "KeyType": "HASH"},
    {"AttributeName": "title", "KeyType": "RANGE"}
    ]' >"$key_schema_json_file"

    echo '['
    {"AttributeName": "year", "AttributeType": "N"},
    {"AttributeName": "title", "AttributeType": "S"}
    ]' >"$attribute_definitions_json_file"

    if dynamodb_create_table -n "$table_name" -a "$attribute_definitions_json_file" \
```

```
-k "$key_schema_json_file" -p "$provisioned_throughput" 1>/dev/null; then
echo "Created a DynamoDB table named $table_name"
else
errecho "The table failed to create. This demo will exit."
clean_up
return 1
fi

echo "Waiting for the table to become active...."

if dynamodb_wait_table_active -n "$table_name"; then
echo "The table is now active."
else
errecho "The table failed to become active. This demo will exit."
cleanup "$table_name"
return 1
fi

echo
echo_repeat "*" 88
echo

echo -n "Enter the title of a movie you want to add to the table: "
get_input
local added_title
added_title=$get_input_result

local added_year
get_int_input "What year was it released? "
added_year=$get_input_result

local rating
get_float_input "On a scale of 1 - 10, how do you rate it? " "1" "10"
rating=$get_input_result

local plot
echo -n "Summarize the plot for me: "
get_input
plot=$get_input_result

echo '{
  "year": {"N" : ""$added_year""},
  "title": {"S" : ""$added_title""},
  "info": {"M" : {"plot": {"S" : ""$plot""}, "rating": {"N" : ""$rating""} } } }
```

```
}' >"$item_json_file"

if dynamodb_put_item -n "$table_name" -i "$item_json_file"; then
  echo "The movie '$added_title' was successfully added to the table
'$table_name'."
else
  errecho "Put item failed. This demo will exit."
  clean_up "$table_name"
  return 1
fi

echo
echo_repeat "*" 88
echo

echo "Let's update your movie '$added_title'."
get_float_input "You rated it $rating, what new rating would you give it? " "1"
"10"
rating=$get_input_result

echo -n "You summarized the plot as '$plot'."
echo "What would you say now? "
get_input
plot=$get_input_result

echo '{
  "year": {"N" : ""$added_year""},
  "title": {"S" : ""$added_title""}
}' >"$key_json_file"

echo '{
  "r": {"N" : ""$rating""},
  "p": {"S" : ""$plot""}
}' >"$item_json_file"

local update_expression="SET info.rating = :r, info.plot = :p"

if dynamodb_update_item -n "$table_name" -k "$key_json_file" -e
"$update_expression" -v "$item_json_file"; then
  echo "Updated '$added_title' with new attributes."
else
  errecho "Update item failed. This demo will exit."
  clean_up "$table_name"
  return 1
```



```
fi

echo
echo_repeat "*" 88
echo

echo "We will now use batch write to upload 150 movie entries into the table."

local batch_json
for batch_json in movie_files/movies_*.json; do
  echo "{ \"\$table_name\" : $(<"$batch_json") }" >"$batch_json_file"
  if dynamodb_batch_write_item -i "$batch_json_file" 1>/dev/null; then
    echo "Entries in $batch_json added to table."
  else
    errecho "Batch write failed. This demo will exit."
    clean_up "$table_name"
    return 1
  fi
done

local title="The Lord of the Rings: The Fellowship of the Ring"
local year="2001"

if get_yes_no_input "Let's move on...do you want to get info about '$title'? (y/n)"; then
  echo '{
"year": {"N" : ""$year""},
"title": {"S" : ""$title""}
}' >"$key_json_file"
  local info
  info=$(dynamodb_get_item -n "$table_name" -k "$key_json_file")

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "Get item failed. This demo will exit."
    clean_up "$table_name"
    return 1
  fi

  echo "Here is what I found:"
  echo "$info"
fi

local ask_for_year=true
```

```

while [[ "$ask_for_year" == true ]]; do
    echo "Let's get a list of movies released in a given year."
    get_int_input "Enter a year between 1972 and 2018: " "1972" "2018"
    year=$get_input_result
    echo '{
"#n": "year"
}' >"$attribute_names_json_file"

    echo '{
":v": {"N" : ""$year""}
}' >"$attributes_values_json_file"

    response=$(dynamodb_query -n "$table_name" -k "#n=:v" -a
"$attribute_names_json_file" -v "$attributes_values_json_file")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "Query table failed. This demo will exit."
        clean_up "$table_name"
        return 1
    fi

    echo "Here is what I found:"
    echo "$response"

    if ! get_yes_no_input "Try another year? (y/n) "; then
        ask_for_year=false
    fi
done

echo "Now let's scan for movies released in a range of years. Enter a year: "
get_int_input "Enter a year between 1972 and 2018: " "1972" "2018"
local start=$get_input_result

get_int_input "Enter another year: " "1972" "2018"
local end=$get_input_result

echo '{
"#n": "year"
}' >"$attribute_names_json_file"

echo '{
":v1": {"N" : ""$start""},
":v2": {"N" : ""$end""}
}

```

```
    }' >"$attributes_values_json_file"

response=$(dynamodb_scan -n "$table_name" -f "#n BETWEEN :v1 AND :v2" -a
"$attribute_names_json_file" -v "$attributes_values_json_file")

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "Scan table failed. This demo will exit."
    clean_up "$table_name"
    return 1
fi

echo "Here is what I found:"
echo "$response"

echo
echo_repeat "*" 88
echo

echo "Let's remove your movie '$added_title' from the table."

if get_yes_no_input "Do you want to remove '$added_title'? (y/n) "; then
    echo '{
"year": {"N" : ""'$added_year'""},
"title": {"S" : ""'$added_title'""}
}' >"$key_json_file"

    if ! dynamodb_delete_item -n "$table_name" -k "$key_json_file"; then
        errecho "Delete item failed. This demo will exit."
        clean_up "$table_name"
        return 1
    fi
fi

if get_yes_no_input "Do you want to delete the table '$table_name'? (y/n) "; then
    if ! clean_up "$table_name"; then
        return 1
    fi
else
    if ! clean_up; then
        return 1
    fi
fi
fi
```

```

return 0
}

```

Fungsi DynamoDB yang digunakan dalam skenario ini.

```

#####
# function dynamodb_create_table
#
# This function creates an Amazon DynamoDB table.
#
# Parameters:
#   -n table_name -- The name of the table to create.
#   -a attribute_definitions -- JSON file path of a list of attributes and their
#   types.
#   -k key_schema -- JSON file path of a list of attributes and their key types.
#   -p provisioned_throughput -- Provisioned throughput settings for the table.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function dynamodb_create_table() {
    local table_name attribute_definitions key_schema provisioned_throughput response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_create_table"
        echo "Creates an Amazon DynamoDB table."
        echo " -n table_name -- The name of the table to create."
        echo " -a attribute_definitions -- JSON file path of a list of attributes and
their types."
        echo " -k key_schema -- JSON file path of a list of attributes and their key
types."
        echo " -p provisioned_throughput -- Provisioned throughput settings for the
table."
        echo ""
    }

    # Retrieve the calling parameters.

```

```
while getopts "n:a:k:p:h" option; do
  case "${option}" in
    n) table_name="${OPTARG}" ;;
    a) attribute_definitions="${OPTARG}" ;;
    k) key_schema="${OPTARG}" ;;
    p) provisioned_throughput="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$attribute_definitions" ]]; then
  errecho "ERROR: You must provide an attribute definitions json file path the -a
parameter."
  usage
  return 1
fi

if [[ -z "$key_schema" ]]; then
  errecho "ERROR: You must provide a key schema json file path the -k parameter."
  usage
  return 1
fi

if [[ -z "$provisioned_throughput" ]]; then
  errecho "ERROR: You must provide a provisioned throughput json file path the -p
parameter."
  usage
  return 1
fi
```

```

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  attribute_definitions:  $attribute_definitions"
iecho "  key_schema:  $key_schema"
iecho "  provisioned_throughput:  $provisioned_throughput"
iecho ""

response=$(aws dynamodb create-table \
  --table-name "$table_name" \
  --attribute-definitions file://"${attribute_definitions}" \
  --key-schema file://"${key_schema}" \
  --provisioned-throughput "${provisioned_throughput}")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-table operation failed.$response"
  return 1
fi

return 0
}

#####
# function dynamodb_describe_table
#
# This function returns the status of a DynamoDB table.
#
# Parameters:
#   -n table_name  -- The name of the table.
#
# Response:
#   - TableStatus:
#     And:
#     0 - Table is active.
#     1 - If it fails.
#####
function dynamodb_describe_table {
  local table_name
  local option OPTARG # Required to use getopt command in a function.

#####

```

```
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_describe_table"
    echo "Describe the status of a DynamoDB table."
    echo "  -n table_name  -- The name of the table."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

local table_status
table_status=$(
    aws dynamodb describe-table \
        --table-name "$table_name" \
        --output text \
        --query 'Table.TableStatus'
)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log "$error_code"
```

```

        errecho "ERROR: AWS reports describe-table operation failed.$table_status"
        return 1
    fi

    echo "$table_status"

    return 0
}

#####
# function dynamodb_put_item
#
# This function puts an item into a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -i item -- Path to json file containing the item values.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_put_item() {
    local table_name item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_put_item"
        echo "Put an item into a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -i item -- Path to json file containing the item values."
        echo ""
    }

    while getopt "n:i:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            i) item="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done

```



```
    ;;
    \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  item:       $item"
iecho ""
iecho ""

response=$(aws dynamodb put-item \
  --table-name "$table_name" \
  --item file://"${item}")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports put-item operation failed.$response"
    return 1
fi

return 0
}
```

```
#####
# function dynamodb_update_item
#
# This function updates an item in a DynamoDB table.
#
#
# Parameters:
#     -n table_name  -- The name of the table.
#     -k keys        -- Path to json file containing the keys that identify the item to
#                       update.
#     -e update expression  -- An expression that defines one or more attributes
#                       to be updated.
#     -v values      -- Path to json file containing the update values.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_update_item() {
    local table_name keys update_expression values response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_update_item"
        echo "Update an item in a DynamoDB table."
        echo " -n table_name  -- The name of the table."
        echo " -k keys        -- Path to json file containing the keys that identify the item
to update."
        echo " -e update expression  -- An expression that defines one or more
attributes to be updated."
        echo " -v values      -- Path to json file containing the update values."
        echo ""
    }

    while getopt "n:k:e:v:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            e) update_expression="${OPTARG}" ;;
            v) values="${OPTARG}" ;;
            h)

```

```
        usage
        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

if [[ -z "$update_expression" ]]; then
    errecho "ERROR: You must provide an update expression with the -e parameter."
    usage
    return 1
fi

if [[ -z "$values" ]]; then
    errecho "ERROR: You must provide a values json file path the -v parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:        $keys"
iecho "  update_expression:  $update_expression"
iecho "  values:     $values"

response=$(aws dynamodb update-item \
  --table-name "$table_name" \
  --key file://" $keys \
```

```

--update-expression "$update_expression" \
--expression-attribute-values file://"values")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports update-item operation failed.$response"
    return 1
fi

return 0

}

#####
# function dynamodb_batch_write_item
#
# This function writes a batch of items into a DynamoDB table.
#
# Parameters:
#     -i item -- Path to json file containing the items to write.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_batch_write_item() {
    local item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_batch_write_item"
        echo "Write a batch of items into a DynamoDB table."
        echo " -i item -- Path to json file containing the items to write."
        echo ""
    }
    while getopt "i:h" option; do
        case "${option}" in
            i) item="${OPTARG}" ;;
            h)

```

```

        usage
        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho "    item:        $item"
iecho ""

response=$(aws dynamodb batch-write-item \
    --request-items file://"${item}")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
fi

return 0
}

#####
# function dynamodb_get_item
#
# This function gets an item from a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.

```

```

# -k keys -- Path to json file containing the keys that identify the item to
get.
# [-q query] -- Optional JMESPath query expression.
#
# Returns:
# The item as text output.
# And:
# 0 - If successful.
# 1 - If it fails.
#####
function dynamodb_get_item() {
    local table_name keys query response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    # #####
    function usage() {
        echo "function dynamodb_get_item"
        echo "Get an item from a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -k keys -- Path to json file containing the keys that identify the item
to get."
        echo " [-q query] -- Optional JMESPath query expression."
        echo ""
    }
    query=""
    while getopt "n:k:q:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            q) query="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

```

```
if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

if [[ -n "$query" ]]; then
    response=$(aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://" $keys" \
        --output text \
        --query "$query")
else
    response=$(
        aws dynamodb get-item \
            --table-name "$table_name" \
            --key file://" $keys" \
            --output text
    )
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports get-item operation failed.$response"
    return 1
fi

if [[ -n "$query" ]]; then
    echo "$response" | sed "/^\t/s/\t//1" # Remove initial tab that the JMSEPath
query inserts on some strings.
else
    echo "$response"
fi

return 0
```

```

}

#####
# function dynamodb_query
#
# This function queries a DynamoDB table.
#
# Parameters:
#   -n table_name -- The name of the table.
#   -k key_condition_expression -- The key condition expression.
#   -a attribute_names -- Path to JSON file containing the attribute names.
#   -v attribute_values -- Path to JSON file containing the attribute values.
#   [-p projection_expression] -- Optional projection expression.
#
# Returns:
#   The items as json output.
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function dynamodb_query() {
    local table_name key_condition_expression attribute_names attribute_values
    projection_expression response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    # #####
    function usage() {
        echo "function dynamodb_query"
        echo "Query a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -k key_condition_expression -- The key condition expression."
        echo " -a attribute_names -- Path to JSON file containing the attribute names."
        echo " -v attribute_values -- Path to JSON file containing the attribute
values."
        echo " [-p projection_expression] -- Optional projection expression."
        echo ""
    }

    while getopt "n:k:a:v:p:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) key_condition_expression="${OPTARG}" ;;

```



```
a) attribute_names="${OPTARG}" ;;
v) attribute_values="${OPTARG}" ;;
p) projection_expression="${OPTARG}" ;;
h)
    usage
    return 0
    ;;
\?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$key_condition_expression" ]]; then
    errecho "ERROR: You must provide a key condition expression with the -k
parameter."
    usage
    return 1
fi

if [[ -z "$attribute_names" ]]; then
    errecho "ERROR: You must provide a attribute names with the -a parameter."
    usage
    return 1
fi

if [[ -z "$attribute_values" ]]; then
    errecho "ERROR: You must provide a attribute values with the -v parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb query \
        --table-name "$table_name" \
```

```

    --key-condition-expression "$key_condition_expression" \
    --expression-attribute-names file://"$attribute_names" \
    --expression-attribute-values file://"$attribute_values")
else
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"$attribute_names" \
        --expression-attribute-values file://"$attribute_values" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports query operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function dynamodb_scan
#
# This function scans a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -f filter_expression -- The filter expression.
#     -a expression_attribute_names -- Path to JSON file containing the expression
#     attribute names.
#     -v expression_attribute_values -- Path to JSON file containing the
#     expression attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
#
# And:
#     0 - If successful.
#     1 - If it fails.

```

```
#####
function dynamodb_scan() {
    local table_name filter_expression expression_attribute_names
    expression_attribute_values projection_expression response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_scan"
        echo "Scan a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -f filter_expression -- The filter expression."
        echo " -a expression_attribute_names -- Path to JSON file containing the
expression attribute names."
        echo " -v expression_attribute_values -- Path to JSON file containing the
expression attribute values."
        echo " [-p projection_expression] -- Optional projection expression."
        echo ""
    }

    while getopt "n:f:a:v:p:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            f) filter_expression="${OPTARG}" ;;
            a) expression_attribute_names="${OPTARG}" ;;
            v) expression_attribute_values="${OPTARG}" ;;
            p) projection_expression="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
    fi
}
#####
```

```
usage
return 1
fi

if [[ -z "$filter_expression" ]]; then
    errecho "ERROR: You must provide a filter expression with the -f parameter."
    usage
    return 1
fi

if [[ -z "$expression_attribute_names" ]]; then
    errecho "ERROR: You must provide expression attribute names with the -a
parameter."
    usage
    return 1
fi

if [[ -z "$expression_attribute_values" ]]; then
    errecho "ERROR: You must provide expression attribute values with the -v
parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"expression_attribute_names" \
        --expression-attribute-values file://"expression_attribute_values")
else
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"expression_attribute_names" \
        --expression-attribute-values file://"expression_attribute_values" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports scan operation failed.$response"
```

```

    return 1
fi

echo "$response"

return 0
}

#####
# function dynamodb_delete_item
#
# This function deletes an item from a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table.
#     -k keys       -- Path to json file containing the keys that identify the item to
delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_delete_item() {
    local table_name keys response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_delete_item"
        echo "Delete an item from a DynamoDB table."
        echo " -n table_name  -- The name of the table."
        echo " -k keys       -- Path to json file containing the keys that identify the item
to delete."
        echo ""
    }
    while getopt "n:k:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}

```

```
    ;;
    \(?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:       $keys"
iecho ""

response=$(aws dynamodb delete-item \
  --table-name "$table_name" \
  --key file://" $keys")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-item operation failed.$response"
    return 1
fi

return 0
}
```

```
#####
```

```

# function dynamodb_delete_table
#
# This function deletes a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_delete_table() {
    local table_name response
    local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function dynamodb_delete_table"
    echo "Deletes an Amazon DynamoDB table."
    echo " -n table_name  -- The name of the table to delete."
    echo ""
}

# Retrieve the calling parameters.
while getopt "n:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1

```

```

fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho ""

response=$(aws dynamodb delete-table \
  --table-name "$table_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-table operation failed.$response"
  return 1
fi

return 0
}

```

Fungsi utilitas yang digunakan dalam skenario ini.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

```



```
#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Untuk detail API, lihat topik berikut di Referensi Perintah AWS CLI .
  - [BatchWriteItem](#)
  - [CreateTable](#)

- [DeleteItem](#)
- [DeleteTable](#)
- [DescribeTable](#)
- [GetItem](#)
- [PutItem](#)
- [Kueri](#)
- [Scan](#)
- [UpdateItem](#)

## HealthImaging contoh menggunakan AWS CLI dengan skrip Bash

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan skrip AWS Command Line Interface with Bash with HealthImaging.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

### **CreateDatastore**

Contoh kode berikut menunjukkan cara menggunakan>CreateDatastore.

AWS CLI dengan skrip Bash

```
#####  
# function errecho  
#
```

```

# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_create_datastore
#
# This function creates an AWS HealthImaging data store for importing DICOM P10
files.
#
# Parameters:
#     -n data_store_name - The name of the data store.
#
# Returns:
#     The datastore ID.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_create_datastore() {
    local datastore_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_create_datastore"
        echo "Creates an AWS HealthImaging data store for importing DICOM P10 files."
        echo "  -n data_store_name - The name of the data store."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) datastore_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage

```

```
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$datastore_name" ]]; then
    errecho "ERROR: You must provide a data store name with the -n parameter."
    usage
    return 1
fi

response=$(aws medical-imaging create-datastore \
    --datastore-name "$datastore_name" \
    --output text \
    --query 'datastoreId')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports medical-imaging create-datastore operation failed.
$response"
    return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [CreateDatastore](#) di Referensi AWS CLI Perintah.

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

## DeleteDatastore

Contoh kode berikut menunjukkan cara menggunakan `DeleteDatastore`.

## AWS CLI dengan skrip Bash

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_delete_datastore
#
# This function deletes an AWS HealthImaging data store.
#
# Parameters:
#     -i datastore_id - The ID of the data store.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_delete_datastore() {
    local datastore_id response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_delete_datastore"
        echo "Deletes an AWS HealthImaging data store."
        echo "  -i datastore_id - The ID of the data store."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopts "i:h" option; do
        case "${option}" in
            i) datastore_id="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)

```

```
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$datastore_id" ]]; then
    errecho "ERROR: You must provide a data store ID with the -i parameter."
    usage
    return 1
fi

response=$(aws medical-imaging delete-datastore \
    --datastore-id "$datastore_id")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports medical-imaging delete-datastore operation failed.
$response"
    return 1
fi

return 0
}
```

- Untuk detail API, lihat [DeleteDatastore](#) di Referensi AWS CLI Perintah.

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

## GetDatastore

Contoh kode berikut menunjukkan cara menggunakan `GetDatastore`.

## AWS CLI dengan skrip Bash

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_get_datastore
#
# Get a data store's properties.
#
# Parameters:
#     -i data_store_id - The ID of the data store.
#
# Returns:
#     [datastore_name, datastore_id, datastore_status, datastore_arn, created_at,
updated_at]
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_get_datastore() {
    local datastore_id option OPTARG # Required to use getopt command in a function.
    local error_code
    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_get_datastore"
        echo "Gets a data store's properties."
        echo "  -i datastore_id - The ID of the data store."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) datastore_id="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```
    ;;
    \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$datastore_id" ]]; then
    errecho "ERROR: You must provide a data store ID with the -i parameter."
    usage
    return 1
fi

local response

response=$(
    aws medical-imaging get-datastore \
        --datastore-id "$datastore_id" \
        --output text \
        --query "[ datastoreProperties.datastoreName,
datastoreProperties.datastoreId, datastoreProperties.datastoreStatus,
datastoreProperties.datastoreArn, datastoreProperties.createdAt,
datastoreProperties.updatedAt]"
)
error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports list-datastores operation failed.$response"
    return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [GetDatastore](#) di Referensi AWS CLI Perintah.



**Note**

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

**ListDatastores**

Contoh kode berikut menunjukkan cara menggunakan `ListDatastores`.

**AWS CLI dengan skrip Bash**

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_list_datastores
#
# List the HealthImaging data stores in the account.
#
# Returns:
#     [[datastore_name, datastore_id, datastore_status]]
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_list_datastores() {
    local option OPTARG # Required to use getopt command in a function.
    local error_code
    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_list_datastores"
        echo "Lists the AWS HealthImaging data stores in the account."
        echo ""
    }

    # Retrieve the calling parameters.
```

```
while getopts "h" option; do
  case "${option}" in
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

local response
response=$(aws medical-imaging list-datastores \
  --output text \
  --query "datastoreSummaries[*][datastoreName, datastoreId, datastoreStatus]")
error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports list-datastores operation failed.$response"
  return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [ListDatastores](#) di Referensi AWS CLI Perintah.

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

## Contoh IAM menggunakan skrip AWS CLI Bash

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan skrip AWS Command Line Interface with Bash dengan IAM.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)
- [Skenario](#)

Tindakan

### **AttachRolePolicy**

Contoh kode berikut menunjukkan cara menggunakan `AttachRolePolicy`.

AWS CLI dengan skrip Bash

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####  
# function errecho  
#  
# This function outputs everything sent to it to STDERR (standard error output).  
#####  
function errecho() {  
    printf "%s\n" "$*" 1>&2
```

```

}

#####
# function iam_attach_role_policy
#
# This function attaches an IAM policy to a role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_attach_role_policy() {
    local role_name policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_attach_role_policy"
        echo "Attaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
        echo "  -n role_name    The name of the IAM role."
        echo "  -p policy_ARN -- The IAM policy document ARN."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            p) policy_arn="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
}

```

```
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam attach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports attach-role-policy operation failed.\n$response"
    return 1
fi

echo "$response"


return 0
}
```

- Untuk detail API, lihat [AttachRolePolicy](#) di Referensi AWS CLI Perintah.

## CreateAccessKey

Contoh kode berikut menunjukkan cara menggunakan `CreateAccessKey`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_user_access_key
#
# This function creates an IAM access key for the specified user.
#
# Parameters:
#     -u user_name -- The name of the IAM user.
#     [-f file_name] -- The optional file name for the access key output.
#
# Returns:
#     [access_key_id access_key_secret]
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_user_access_key() {
    local user_name file_name response
    local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function iam_create_user_access_key"
    echo "Creates an AWS Identity and Access Management (IAM) key pair."
    echo "  -u user_name  The name of the IAM user."
    echo "  [-f file_name] Optional file name for the access key output."
}
```

```
    echo ""
}

# Retrieve the calling parameters.
while getopts "u:f:h" option; do
    case "${option}" in
        u) user_name="${OPTARG}" ;;
        f) file_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

response=$(aws iam create-access-key \
    --user-name "$user_name" \
    --output text)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
fi

if [[ -n "$file_name" ]]; then
    echo "$response" >"$file_name"
fi

local key_id key_secret
```

```
# shellcheck disable=SC2086
key_id=$(echo $response | cut -f 2 -d ' ')
# shellcheck disable=SC2086
key_secret=$(echo $response | cut -f 4 -d ' ')

echo "$key_id $key_secret"

return 0
}
```

- Untuk detail API, lihat [CreateAccessKey](#) di Referensi AWS CLI Perintah.

## CreatePolicy

Contoh kode berikut menunjukkan cara menggunakan `CreatePolicy`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_policy
#
# This function creates an IAM policy.
#
# Parameters:
#     -n policy_name -- The name of the IAM policy.
#     -p policy_json -- The policy document.
```



```

#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_policy() {
    local policy_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_policy"
        echo "Creates an AWS Identity and Access Management (IAM) policy."
        echo "  -n policy_name  The name of the IAM policy."
        echo "  -p policy_json -- The policy document."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) policy_name="${OPTARG}" ;;
            p) policy_document="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$policy_name" ]]; then
        errecho "ERROR: You must provide a policy name with the -n parameter."
        usage
        return 1
    fi

    if [[ -z "$policy_document" ]]; then
        errecho "ERROR: You must provide a policy document with the -p parameter."
    fi
}

```

```

usage
return 1
fi

response=$(aws iam create-policy \
  --policy-name "$policy_name" \
  --policy-document "$policy_document" \
  --output text \
  --query Policy.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-policy operation failed.\n$response"
  return 1
fi

echo "$response"
}

```

- Untuk detail API, lihat [CreatePolicy](#) di Referensi AWS CLI Perintah.

## CreateRole

Contoh kode berikut menunjukkan cara menggunakan `CreateRole`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {

```

```

    printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_role
#
# This function creates an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_json -- The assume role policy document.
#
# Returns:
#     The ARN of the role.
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_role() {
    local role_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user_access_key"
        echo "Creates an AWS Identity and Access Management (IAM) role."
        echo "  -n role_name  The name of the IAM role."
        echo "  -p policy_json -- The assume role policy document."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            p) policy_document="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
        esac
    done
}

```

```
    ;;
    esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
fi

response=$(aws iam create-role \
    --role-name "$role_name" \
    --assume-role-policy-document "$policy_document" \
    --output text \
    --query Role.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [CreateRole](#) di Referensi AWS CLI Perintah.

## CreateUser

Contoh kode berikut menunjukkan cara menggunakan `CreateUser`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_user
#
# This function creates the specified IAM user, unless
# it already exists.
#
# Parameters:
#     -u user_name -- The name of the user to create.
#
# Returns:
#     The ARN of the user.
#     And:
#     0 - If successful.
```

```
# 1 - If it fails.
#####
function iam_create_user() {
    local user_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user"
        echo "Creates an WS Identity and Access Management (IAM) user. You must supply a
username:"
        echo "  -u user_name    The name of the user. It must be unique within the
account."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$user_name" ]]; then
        errecho "ERROR: You must provide a username with the -u parameter."
        usage
        return 1
    fi

    iecho "Parameters:\n"
    iecho "  User name:  $user_name"
    iecho ""

    # If the user already exists, we don't want to try to create it.
```

```

if (iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name already exists in the account."
    return 1
fi

response=$(aws iam create-user --user-name "$user_name" \
    --output text \
    --query 'User.Arn')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-user operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

```

- Untuk detail API, lihat [CreateUser](#) di Referensi AWS CLI Perintah.

## DeleteAccessKey

Contoh kode berikut menunjukkan cara menggunakan `DeleteAccessKey`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####

```

```

function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_access_key
#
# This function deletes an IAM access key for the specified IAM user.
#
# Parameters:
#     -u user_name -- The name of the user.
#     -k access_key -- The access key to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_access_key() {
    local user_name access_key response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_access_key"
        echo "Deletes an WS Identity and Access Management (IAM) access key for the
specified IAM user"
        echo "  -u user_name    The name of the user."
        echo "  -k access_key    The access key to delete."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:k:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            k) access_key="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
        esac
    done
}

```



```
        ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

if [[ -z "$access_key" ]]; then
    errecho "ERROR: You must provide an access key with the -k parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    Username:  $user_name"
iecho "    Access key:  $access_key"
iecho ""

response=$(aws iam delete-access-key \
    --user-name "$user_name" \
    --access-key-id "$access_key")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-access-key operation failed.\n$response"
    return 1
fi

iecho "delete-access-key response:$response"
iecho

return 0
}
```

- Untuk detail API, lihat [DeleteAccessKey](#) di Referensi AWS CLI Perintah.

## DeletePolicy

Contoh kode berikut menunjukkan cara menggunakan DeletePolicy.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_policy
#
# This function deletes an IAM policy.
#
# Parameters:
#     -n policy_arn -- The name of the IAM policy arn.
#
# Returns:
#     0 - If successful.
```

```

#      1 - If it fails.
#####
function iam_delete_policy() {
    local policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_policy"
        echo "Deletes an WS Identity and Access Management (IAM) policy"
        echo "  -n policy_arn -- The name of the IAM policy arn."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) policy_arn="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$policy_arn" ]]; then
        errecho "ERROR: You must provide a policy arn with the -n parameter."
        usage
        return 1
    fi

    iecho "Parameters:\n"
    iecho "  Policy arn: $policy_arn"
    iecho ""

    response=$(aws iam delete-policy \
        --policy-arn "$policy_arn")

```

```

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-policy operation failed.\n$response"
    return 1
fi

iecho "delete-policy response:$response"
iecho

return 0
}

```

- Untuk detail API, lihat [DeletePolicy](#) di Referensi AWS CLI Perintah.

## DeleteRole

Contoh kode berikut menunjukkan cara menggunakan `DeleteRole`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####

```

```

# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_role
#
# This function deletes an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_role() {
    local role_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_role"
        echo "Deletes an WS Identity and Access Management (IAM) role"
        echo "  -n role_name -- The name of the IAM role."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
        esac
    done
}

```

```
    ;;
    esac
done
export OPTIND=1

echo "role_name:$role_name"
if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    Role name: $role_name"
iecho ""

response=$(aws iam delete-role \
    --role-name "$role_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-role operation failed.\n$response"
    return 1
fi

iecho "delete-role response:$response"
iecho

return 0
}
```

- Untuk detail API, lihat [DeleteRole](#) di Referensi AWS CLI Perintah.

## DeleteUser

Contoh kode berikut menunjukkan cara menggunakan `DeleteUser`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_user
#
# This function deletes the specified IAM user.
#
# Parameters:
#     -u user_name -- The name of the user to create.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_user() {
```

```
local user_name response
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function iam_delete_user"
    echo "Deletes an WS Identity and Access Management (IAM) user. You must supply a
username:"
    echo "  -u user_name    The name of the user."
    echo ""
}

# Retrieve the calling parameters.
while getopt "u:h" option; do
    case "${option}" in
        u) user_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  User name:  $user_name"
iecho ""

# If the user does not exist, we don't want to try to delete it.
if (! iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name does not exist in the account."
    return 1
fi
```



```

response=$(aws iam delete-user \
  --user-name "$user_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-user operation failed.$response"
  return 1
fi

iecho "delete-user response:$response"
iecho

return 0
}

```

- Untuk detail API, lihat [DeleteUser](#) di Referensi AWS CLI Perintah.

## DetachRolePolicy

Contoh kode berikut menunjukkan cara menggunakan `DetachRolePolicy`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

```

```
#####  
# function iam_detach_role_policy  
#  
# This function detaches an IAM policy to a role.  
#  
# Parameters:  
#   -n role_name -- The name of the IAM role.  
#   -p policy_ARN -- The IAM policy document ARN..  
#  
# Returns:  
#   0 - If successful.  
#   1 - If it fails.  
#####  
function iam_detach_role_policy() {  
    local role_name policy_arn response  
    local option OPTARG # Required to use getopt command in a function.  
  
    # bashsupport disable=BP5008  
    function usage() {  
        echo "function iam_detach_role_policy"  
        echo "Detaches an AWS Identity and Access Management (IAM) policy to an IAM  
role."  
        echo "  -n role_name  The name of the IAM role."  
        echo "  -p policy_ARN -- The IAM policy document ARN."  
        echo ""  
    }  
  
    # Retrieve the calling parameters.  
    while getopt "n:p:h" option; do  
        case "${option}" in  
            n) role_name="${OPTARG}" ;;  
            p) policy_arn="${OPTARG}" ;;  
            h)  
                usage  
                return 0  
                ;;  
            \?)  
                echo "Invalid parameter"  
                usage  
                return 1  
                ;;  
        esac  
    done  
    export OPTIND=1
```

```
if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam detach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports detach-role-policy operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [DetachRolePolicy](#) di Referensi AWS CLI Perintah.

## GetUser

Contoh kode berikut menunjukkan cara menggunakan `GetUser`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_user_exists
#
# This function checks to see if the specified AWS Identity and Access Management
# (IAM) user already exists.
#
# Parameters:
#     $1 - The name of the IAM user to check.
#
# Returns:
#     0 - If the user already exists.
#     1 - If the user doesn't exist.
#####
function iam_user_exists() {
    local user_name
    user_name=$1

    # Check whether the IAM user already exists.
    # We suppress all output - we're interested only in the return code.

    local errors
    errors=$(aws iam get-user \
        --user-name "$user_name" 2>&1 >/dev/null)

    local error_code=${?}
```

```

if [[ $error_code -eq 0 ]]; then
    return 0 # 0 in Bash script means true.
else
    if [[ $errors != *"error"*(NoSuchEntity)* ]]; then
        aws_cli_error_log $error_code
        errecho "Error calling iam get-user $errors"
    fi

    return 1 # 1 in Bash script means false.
fi
}

```

- Untuk detail API, lihat [GetUser](#) di Referensi AWS CLI Perintah.

## ListAccessKeys

Contoh kode berikut menunjukkan cara menggunakan `ListAccessKeys`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_list_access_keys
#
# This function lists the access keys for the specified user.
#

```

```

# Parameters:
#     -u user_name -- The name of the IAM user.
#
# Returns:
#     access_key_ids
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_list_access_keys() {

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_list_access_keys"
        echo "Lists the AWS Identity and Access Management (IAM) access key IDs for the
specified user."
        echo "  -u user_name  The name of the IAM user."
        echo ""
    }

    local user_name response
    local option OPTARG # Required to use getopt command in a function.
    # Retrieve the calling parameters.
    while getopt "u:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$user_name" ]]; then
        errecho "ERROR: You must provide a username with the -u parameter."
        usage
        return 1
    fi
}

```

```

response=$(aws iam list-access-keys \
  --user-name "$user_name" \
  --output text \
  --query 'AccessKeyMetadata[].AccessKeyId')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports list-access-keys operation failed.$response"
  return 1
fi

echo "$response"

return 0
}

```

- Untuk detail API, lihat [ListAccessKeys](#) di Referensi AWS CLI Perintah.

## ListUsers

Contoh kode berikut menunjukkan cara menggunakan `ListUsers`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

```

```
#####
# function iam_list_users
#
# List the IAM users in the account.
#
# Returns:
#     The list of users names
# And:
#     0 - If the user already exists.
#     1 - If the user doesn't exist.
#####
function iam_list_users() {
    local option OPTARG # Required to use getopt command in a function.
    local error_code
    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_list_users"
        echo "Lists the AWS Identity and Access Management (IAM) user in the account."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "h" option; do
        case "${option}" in
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    local response

    response=$(aws iam list-users \
        --output text \
        --query "Users[].UserName")
    error_code=${?}
}
```



```
if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports list-users operation failed.$response"
  return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [ListUsers](#) di Referensi AWS CLI Perintah.

## Skenario

Buat pengguna dan ambil peran

Contoh kode berikut menunjukkan cara membuat pengguna dan mengambil peran.

### Warning

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti [AWS IAM Identity Center](#).

- Buat pengguna tanpa izin.
- Buat peran yang memberikan izin untuk mencantumkan bucket Amazon S3 untuk akun tersebut.
- Tambahkan kebijakan agar pengguna dapat mengambil peran tersebut.
- Asumsikan peran dan daftar bucket S3 menggunakan kredensial sementara, lalu bersihkan sumber daya.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function iam_create_user_assume_role
#
# Scenario to create an IAM user, create an IAM role, and apply the role to the
# user.
#
# "IAM access" permissions are needed to run this code.
# "STS assume role" permissions are needed to run this code. (Note: It might be
# necessary to
# create a custom policy).
#
# Returns:
# 0 - If successful.
# 1 - If an error occurred.
#####
function iam_create_user_assume_role() {
{
    if [ "$IAM_OPERATIONS_SOURCED" != "True" ]; then

        source ./iam_operations.sh
    fi
}

echo_repeat "*" 88
echo "Welcome to the IAM create user and assume role demo."
echo
echo "This demo will create an IAM user, create an IAM role, and apply the role to
the user."
echo_repeat "*" 88
echo

echo -n "Enter a name for a new IAM user: "
get_input
user_name=$get_input_result
```

```
local user_arn
user_arn=$(iam_create_user -u "$user_name")

# shellcheck disable=SC2181
if [[ ${?} == 0 ]]; then
    echo "Created demo IAM user named $user_name"
else
    errecho "$user_arn"
    errecho "The user failed to create. This demo will exit."
    return 1
fi

local access_key_response
access_key_response=$(iam_create_user_access_key -u "$user_name")
# shellcheck disable=SC2181
if [[ ${?} != 0 ]]; then
    errecho "The access key failed to create. This demo will exit."
    clean_up "$user_name"
    return 1
fi

IFS=$'\t ' read -r -a access_key_values <<<"$access_key_response"
local key_name=${access_key_values[0]}
local key_secret=${access_key_values[1]}

echo "Created access key named $key_name"

echo "Wait 10 seconds for the user to be ready."
sleep 10
echo_repeat "*" 88
echo

local iam_role_name
iam_role_name=$(generate_random_name "test-role")
echo "Creating a role named $iam_role_name with user $user_name as the principal."

local assume_role_policy_document="{
  \"Version\": \"2012-10-17\",
  \"Statement\": [{
    \"Effect\": \"Allow\",
    \"Principal\": {\"AWS\": \"$user_arn\"},
    \"Action\": \"sts:AssumeRole\"
  }]
}
```

```
}"  
  
local role_arn  
role_arn=$(iam_create_role -n "$iam_role_name" -p "$assume_role_policy_document")  
  
# shellcheck disable=SC2181  
if [ ${?} == 0 ]; then  
    echo "Created IAM role named $iam_role_name"  
else  
    errecho "The role failed to create. This demo will exit."  
    clean_up "$user_name" "$key_name"  
    return 1  
fi  
  
local policy_name  
policy_name=$(generate_random_name "test-policy")  
local policy_document="{  
    \"Version\": \"2012-10-17\",  
    \"Statement\": [{  
        \"Effect\": \"Allow\",  
        \"Action\": \"s3:ListAllMyBuckets\",  
        \"Resource\": \"arn:aws:s3::*\"}]}"  
  
local policy_arn  
policy_arn=$(iam_create_policy -n "$policy_name" -p "$policy_document")  
# shellcheck disable=SC2181  
if [[ ${?} == 0 ]]; then  
    echo "Created IAM policy named $policy_name"  
else  
    errecho "The policy failed to create."  
    clean_up "$user_name" "$key_name" "$iam_role_name"  
    return 1  
fi  
  
if (iam_attach_role_policy -n "$iam_role_name" -p "$policy_arn"); then  
    echo "Attached policy $policy_arn to role $iam_role_name"  
else  
    errecho "The policy failed to attach."  
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn"  
    return 1  
fi  
  
local assume_role_policy_document="{  
    \"Version\": \"2012-10-17\",
```

```
        \Statement\": [{
            \Effect\: \"Allow\",
            \Action\: \"sts:AssumeRole\",
            \Resource\: \"\${role_arn}\"}]}"

local assume_role_policy_name
assume_role_policy_name=$(generate_random_name "test-assume-role-")

# shellcheck disable=SC2181
local assume_role_policy_arn
assume_role_policy_arn=$(iam_create_policy -n "\${assume_role_policy_name}" -p
"\${assume_role_policy_document}")
# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    echo "Created IAM policy named \${assume_role_policy_name} for sts assume role"
else
    errecho "The policy failed to create."
    clean_up "\${user_name}" "\${key_name}" "\${iam_role_name}" "\${policy_arn}" "\${policy_arn}"
    return 1
fi

echo "Wait 10 seconds to give AWS time to propagate these new resources and
connections."
sleep 10
echo_repeat "*" 88
echo

echo "Try to list buckets without the new user assuming the role."
echo_repeat "*" 88
echo

# Set the environment variables for the created user.
# bashsupport disable=BP2001
export AWS_ACCESS_KEY_ID=\${key_name}
# bashsupport disable=BP2001
export AWS_SECRET_ACCESS_KEY=\${key_secret}

local buckets
buckets=$(s3_list_buckets)

# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    local bucket_count
    bucket_count=$(echo "\${buckets}" | wc -w | xargs)
```

```
    echo "There are $bucket_count buckets in the account. This should not have
happened."
    else
        errecho "Because the role with permissions has not been assumed, listing buckets
failed."
    fi

    echo
    echo_repeat "*" 88
    echo "Now assume the role $iam_role_name and list the buckets."
    echo_repeat "*" 88
    echo

local credentials

credentials=$(sts_assume_role -r "$role_arn" -n "AssumeRoleDemoSession")
# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    echo "Assumed role $iam_role_name"
else
    errecho "Failed to assume role."
    export AWS_ACCESS_KEY_ID=""
    export AWS_SECRET_ACCESS_KEY=""
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"
    return 1
fi

IFS=$'\t ' read -r -a credentials <<<"$credentials"

export AWS_ACCESS_KEY_ID=${credentials[0]}
export AWS_SECRET_ACCESS_KEY=${credentials[1]}
# bashsupport disable=BP2001
export AWS_SESSION_TOKEN=${credentials[2]}

buckets=$(s3_list_buckets)

# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    local bucket_count
    bucket_count=$(echo "$buckets" | wc -w | xargs)
    echo "There are $bucket_count buckets in the account. Listing buckets succeeded
because of "
    echo "the assumed role."
```

```

else
  errecho "Failed to list buckets. This should not happen."
  export AWS_ACCESS_KEY_ID=""
  export AWS_SECRET_ACCESS_KEY=""
  export AWS_SESSION_TOKEN=""
  clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"
  return 1
fi

local result=0
export AWS_ACCESS_KEY_ID=""
export AWS_SECRET_ACCESS_KEY=""

echo
echo_repeat "*" 88
echo "The created resources will now be deleted."
echo_repeat "*" 88
echo

clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
  result=1
fi

return $result
}

```

Fungsi IAM yang digunakan dalam skenario ini.

```

#####
# function iam_user_exists
#
# This function checks to see if the specified AWS Identity and Access Management
(IAM) user already exists.
#
# Parameters:
#     $1 - The name of the IAM user to check.
#

```

```

# Returns:
#     0 - If the user already exists.
#     1 - If the user doesn't exist.
#####
function iam_user_exists() {
    local user_name
    user_name=$1

    # Check whether the IAM user already exists.
    # We suppress all output - we're interested only in the return code.

    local errors
    errors=$(aws iam get-user \
        --user-name "$user_name" 2>&1 >/dev/null)

    local error_code=${?}

    if [[ $error_code -eq 0 ]]; then
        return 0 # 0 in Bash script means true.
    else
        if [[ $errors != *"error"*(NoSuchEntity)* ]]; then
            aws_cli_error_log $error_code
            errecho "Error calling iam get-user $errors"
        fi

        return 1 # 1 in Bash script means false.
    fi
}

#####
# function iam_create_user
#
# This function creates the specified IAM user, unless
# it already exists.
#
# Parameters:
#     -u user_name -- The name of the user to create.
#
# Returns:
#     The ARN of the user.
# And:
#     0 - If successful.
#     1 - If it fails.
#####

```



```
function iam_create_user() {
    local user_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user"
        echo "Creates an WS Identity and Access Management (IAM) user. You must supply a
username:"
        echo "  -u user_name    The name of the user. It must be unique within the
account."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$user_name" ]]; then
        errecho "ERROR: You must provide a username with the -u parameter."
        usage
        return 1
    fi

    iecho "Parameters:\n"
    iecho "  User name:  $user_name"
    iecho ""

    # If the user already exists, we don't want to try to create it.
    if (iam_user_exists "$user_name"); then
        errecho "ERROR: A user with that name already exists in the account."
    fi
}
```

```

    return 1
fi

response=$(aws iam create-user --user-name "$user_name" \
  --output text \
  --query 'User.Arn')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-user operation failed.$response"
  return 1
fi

echo "$response"

return 0
}

#####
# function iam_create_user_access_key
#
# This function creates an IAM access key for the specified user.
#
# Parameters:
#   -u user_name -- The name of the IAM user.
#   [-f file_name] -- The optional file name for the access key output.
#
# Returns:
#   [access_key_id access_key_secret]
#   And:
#   0 - If successful.
#   1 - If it fails.
#####
function iam_create_user_access_key() {
  local user_name file_name response
  local option OPTARG # Required to use getopt command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user_access_key"
    echo "Creates an AWS Identity and Access Management (IAM) key pair."
    echo "  -u user_name  The name of the IAM user."
  }

```

```
    echo " [-f file_name] Optional file name for the access key output."
    echo ""
}

# Retrieve the calling parameters.
while getopts "u:f:h" option; do
    case "${option}" in
        u) user_name="${OPTARG}" ;;
        f) file_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

response=$(aws iam create-access-key \
    --user-name "$user_name" \
    --output text)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
fi

if [[ -n "$file_name" ]]; then
    echo "$response" >"$file_name"
fi
```

```

local key_id key_secret
# shellcheck disable=SC2086
key_id=$(echo $response | cut -f 2 -d ' ')
# shellcheck disable=SC2086
key_secret=$(echo $response | cut -f 4 -d ' ')

echo "$key_id $key_secret"

return 0
}

#####
# function iam_create_role
#
# This function creates an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_json -- The assume role policy document.
#
# Returns:
#     The ARN of the role.
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_role() {
    local role_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user_access_key"
        echo "Creates an AWS Identity and Access Management (IAM) role."
        echo "  -n role_name    The name of the IAM role."
        echo "  -p policy_json -- The assume role policy document."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            p) policy_document="${OPTARG}" ;;
        esac
    done

```

```
h)
  usage
  return 0
  ;;
\?)
  echo "Invalid parameter"
  usage
  return 1
  ;;
esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
  errecho "ERROR: You must provide a role name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$policy_document" ]]; then
  errecho "ERROR: You must provide a policy document with the -p parameter."
  usage
  return 1
fi

response=$(aws iam create-role \
  --role-name "$role_name" \
  --assume-role-policy-document "$policy_document" \
  --output text \
  --query Role.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-role operation failed.\n$response"
  return 1
fi

echo "$response"

return 0
}
```

```
#####
# function iam_create_policy
#
# This function creates an IAM policy.
#
# Parameters:
#   -n policy_name -- The name of the IAM policy.
#   -p policy_json -- The policy document.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function iam_create_policy() {
    local policy_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_policy"
        echo "Creates an AWS Identity and Access Management (IAM) policy."
        echo "  -n policy_name  The name of the IAM policy."
        echo "  -p policy_json -- The policy document."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) policy_name="${OPTARG}" ;;
            p) policy_document="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1
}
```

```

if [[ -z "$policy_name" ]]; then
    errecho "ERROR: You must provide a policy name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
fi

response=$(aws iam create-policy \
    --policy-name "$policy_name" \
    --policy-document "$policy_document" \
    --output text \
    --query Policy.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-policy operation failed.\n$response"
    return 1
fi

echo "$response"
}

#####
# function iam_attach_role_policy
#
# This function attaches an IAM policy to a role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_attach_role_policy() {
    local role_name policy_arn response

```

```
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function iam_attach_role_policy"
    echo "Attaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
    echo "  -n role_name    The name of the IAM role."
    echo "  -p policy_ARN -- The IAM policy document ARN."
    echo ""
}

# Retrieve the calling parameters.
while getopt "n:p:h" option; do
    case "${option}" in
        n) role_name="${OPTARG}" ;;
        p) policy_arn="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam attach-role-policy \
    --role-name "$role_name" \
```



```

    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports attach-role-policy operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function iam_detach_role_policy
#
# This function detaches an IAM policy to a role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_detach_role_policy() {
    local role_name policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_detach_role_policy"
        echo "Detaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
        echo "  -n role_name    The name of the IAM role."
        echo "  -p policy_ARN -- The IAM policy document ARN."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do

```

```
case "${option}" in
  n) role_name="${OPTARG}" ;;
  p) policy_arn="${OPTARG}" ;;
  h)
    usage
    return 0
    ;;
  \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
  errecho "ERROR: You must provide a role name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$policy_arn" ]]; then
  errecho "ERROR: You must provide a policy ARN with the -p parameter."
  usage
  return 1
fi

response=$(aws iam detach-role-policy \
  --role-name "$role_name" \
  --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports detach-role-policy operation failed.\n$response"
  return 1
fi

echo "$response"

return 0
}
```

```
#####
# function iam_delete_policy
#
# This function deletes an IAM policy.
#
# Parameters:
#     -n policy_arn -- The name of the IAM policy arn.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_policy() {
    local policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_policy"
        echo "Deletes an WS Identity and Access Management (IAM) policy"
        echo "  -n policy_arn -- The name of the IAM policy arn."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) policy_arn="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$policy_arn" ]]; then
        errecho "ERROR: You must provide a policy arn with the -n parameter."
    fi
}

```

```

    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    Policy arn: $policy_arn"
iecho ""

response=$(aws iam delete-policy \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-policy operation failed.\n$response"
    return 1
fi

iecho "delete-policy response:$response"
iecho

return 0
}

#####
# function iam_delete_role
#
# This function deletes an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_role() {
    local role_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_role"
    }

```

```
    echo "Deletes an WS Identity and Access Management (IAM) role"
    echo "  -n role_name -- The name of the IAM role."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:h" option; do
    case "${option}" in
        n) role_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

echo "role_name:$role_name"
if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  Role name: $role_name"
iecho ""

response=$(aws iam delete-role \
  --role-name "$role_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-role operation failed.\n$response"
    return 1
fi
```

```

iecho "delete-role response:$response"
iecho

return 0
}

#####
# function iam_delete_access_key
#
# This function deletes an IAM access key for the specified IAM user.
#
# Parameters:
#     -u user_name  -- The name of the user.
#     -k access_key -- The access key to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_access_key() {
    local user_name access_key response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_access_key"
        echo "Deletes an WS Identity and Access Management (IAM) access key for the
specified IAM user"
        echo "  -u user_name    The name of the user."
        echo "  -k access_key    The access key to delete."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:k:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            k) access_key="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"

```

```

        usage
        return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

if [[ -z "$access_key" ]]; then
    errecho "ERROR: You must provide an access key with the -k parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    Username:  $user_name"
iecho "    Access key: $access_key"
iecho ""

response=$(aws iam delete-access-key \
    --user-name "$user_name" \
    --access-key-id "$access_key")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-access-key operation failed.\n$response"
    return 1
fi

iecho "delete-access-key response:$response"
iecho

return 0
}

#####
# function iam_delete_user

```

```

#
# This function deletes the specified IAM user.
#
# Parameters:
#     -u user_name  -- The name of the user to create.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_user() {
    local user_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_user"
        echo "Deletes an WS Identity and Access Management (IAM) user. You must supply a
username:"
        echo "  -u user_name    The name of the user."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$user_name" ]]; then
        errecho "ERROR: You must provide a username with the -u parameter."
        usage
        return 1
    fi
}

```



```
fi

iecho "Parameters:\n"
iecho "    User name:  $user_name"
iecho ""

# If the user does not exist, we don't want to try to delete it.
if (! iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name does not exist in the account."
    return 1
fi

response=$(aws iam delete-user \
    --user-name "$user_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-user operation failed.$response"
    return 1
fi

iecho "delete-user response:$response"
iecho

return 0
}
```

- Untuk detail API, lihat topik berikut di Referensi Perintah AWS CLI .
  - [AttachRolePolicy](#)
  - [CreateAccessKey](#)
  - [CreatePolicy](#)
  - [CreateRole](#)
  - [CreateUser](#)
  - [DeleteAccessKey](#)
  - [DeletePolicy](#)
  - [DeleteRole](#)
  - [DeleteUser](#)

- [DeleteUserPolicy](#)
- [DetachRolePolicy](#)
- [PutUserPolicy](#)

## Contoh Amazon S3 menggunakan skrip AWS CLI Bash

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan skrip AWS Command Line Interface with Bash dengan Amazon S3.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)
- [Skenario](#)

Tindakan

### CopyObject

Contoh kode berikut menunjukkan cara menggunakan CopyObject.

AWS CLI dengan skrip Bash

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
```

```

# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function copy_item_in_bucket
#
# This function creates a copy of the specified file in the same bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file from and to.
#     $2 - The key of the source file to copy.
#     $3 - The key of the destination file.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_item_in_bucket() {
    local bucket_name=$1
    local source_key=$2
    local destination_key=$3
    local response

    response=$(aws s3api copy-object \
        --bucket "$bucket_name" \
        --copy-source "$bucket_name/$source_key" \
        --key "$destination_key")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api copy-object operation failed.\n$response"
        return 1
    fi
}

```

- Untuk detail API, lihat [CopyObject](#) di Referensi AWS CLI Perintah.

## CreateBucket

Contoh kode berikut menunjukkan cara menggunakan `CreateBucket`.

### AWS CLI dengan skrip Bash

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function create-bucket
#
# This function creates the specified bucket in the specified AWS Region, unless
# it already exists.
#
# Parameters:
#     -b bucket_name -- The name of the bucket to create.
#     -r region_code -- The code for an AWS Region in which to
#                       create the bucket.
```

```
#
# Returns:
#     The URL of the bucket that was created.
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function create_bucket() {
    local bucket_name region_code response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function create_bucket"
        echo "Creates an Amazon S3 bucket. You must supply a bucket name:"
        echo "  -b bucket_name    The name of the bucket. It must be globally unique."
        echo "  [-r region_code]   The code for an AWS Region in which the bucket is
created."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "b:r:h" option; do
        case "${option}" in
            b) bucket_name="${OPTARG}" ;;
            r) region_code="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done

    if [[ -z "$bucket_name" ]]; then
        errecho "ERROR: You must provide a bucket name with the -b parameter."
        usage
        return 1
    fi
}
```

```
local bucket_config_arg
# A location constraint for "us-east-1" returns an error.
if [[ -n "$region_code" ]] && [[ "$region_code" != "us-east-1" ]]; then
    bucket_config_arg="--create-bucket-configuration LocationConstraint=
$region_code"
fi

iecho "Parameters:\n"
iecho "    Bucket name:    $bucket_name"
iecho "    Region code:    $region_code"
iecho ""

# If the bucket already exists, we don't want to try to create it.
if (bucket_exists "$bucket_name"); then
    errecho "ERROR: A bucket with that name already exists. Try again."
    return 1
fi

# shellcheck disable=SC2086
response=$(aws s3api create-bucket \
    --bucket "$bucket_name" \
    $bucket_config_arg)

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports create-bucket operation failed.\n$response"
    return 1
fi
}
```

- Untuk detail API, lihat [CreateBucket](#) di Referensi AWS CLI Perintah.

## DeleteBucket

Contoh kode berikut menunjukkan cara menggunakan `DeleteBucket`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function delete_bucket
#
# This function deletes the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api delete-bucket \
        --bucket "$bucket_name")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-bucket failed.\n$response"
        return 1
    fi
}
}
```

- Untuk detail API, lihat [DeleteBucket](#) di Referensi AWS CLI Perintah.

## DeleteObject

Contoh kode berikut menunjukkan cara menggunakan `DeleteObject`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function delete_item_in_bucket
#
# This function deletes the specified file from the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.
#     $2 - The key (file name) in the bucket to delete.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_item_in_bucket() {
    local bucket_name=$1
    local key=$2
```



```

local response

response=$(aws s3api delete-object \
  --bucket "$bucket_name" \
  --key "$key")

# shellcheck disable=SC2181
if [[ $? -ne 0 ]]; then
  errecho "ERROR: AWS reports s3api delete-object operation failed.\n$response"
  return 1
fi
}

```

- Untuk detail API, lihat [DeleteObject](#) di Referensi AWS CLI Perintah.

## DeleteObjects

Contoh kode berikut menunjukkan cara menggunakan `DeleteObjects`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function delete_items_in_bucket
#
# This function deletes the specified list of keys from the specified bucket.

```

```

#
# Parameters:
#     $1 - The name of the bucket.
#     $2 - A list of keys in the bucket to delete.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_items_in_bucket() {
    local bucket_name=$1
    local keys=$2
    local response

    # Create the JSON for the items to delete.
    local delete_items
    delete_items="{\"Objects\":["
    for key in $keys; do
        delete_items="$delete_items{\"Key\": \"$key\"},"
    done
    delete_items=${delete_items%?} # Remove the final comma.
    delete_items="$delete_items]"

    response=$(aws s3api delete-objects \
        --bucket "$bucket_name" \
        --delete "$delete_items")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-object operation failed.\n$response"
        return 1
    fi
}

```

- Untuk detail API, lihat [DeleteObjects](#) di Referensi AWS CLI Perintah.

## GetObject

Contoh kode berikut menunjukkan cara menggunakan `GetObject`.

## AWS CLI dengan skrip Bash

 Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function download_object_from_bucket
#
# This function downloads an object in a bucket to a file.
#
# Parameters:
#     $1 - The name of the bucket to download the object from.
#     $2 - The path and file name to store the downloaded bucket.
#     $3 - The key (name) of the object in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function download_object_from_bucket() {
    local bucket_name=$1
    local destination_file_name=$2
    local object_name=$3
    local response

    response=$(aws s3api get-object \
        --bucket "$bucket_name" \
        --key "$object_name" \
        "$destination_file_name")
}
```

```
# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports put-object operation failed.\n$response"
    return 1
fi
}
```

- Untuk detail API, lihat [GetObject](#) di Referensi AWS CLI Perintah.

## HeadBucket

Contoh kode berikut menunjukkan cara menggunakan HeadBucket.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function bucket_exists
#
# This function checks to see if the specified bucket already exists.
#
# Parameters:
#     $1 - The name of the bucket to check.
#
# Returns:
#     0 - If the bucket already exists.
#     1 - If the bucket doesn't exist.
#####
function bucket_exists() {
    local bucket_name
    bucket_name=$1

    # Check whether the bucket already exists.
    # We suppress all output - we're interested only in the return code.
```

```

if aws s3api head-bucket \
  --bucket "$bucket_name" \
  >/dev/null 2>&1; then
  return 0 # 0 in Bash script means true.
else
  return 1 # 1 in Bash script means false.
fi
}

```

- Untuk detail API, lihat [HeadBucket](#) di Referensi AWS CLI Perintah.

## ListObjectsV2

Contoh kode berikut menunjukkan cara menggunakan `ListObjectsV2`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function list_items_in_bucket
#
# This function displays a list of the files in the bucket with each file's
# size. The function uses the --query parameter to retrieve only the key and
# size fields from the Contents collection.
#
# Parameters:
#   $1 - The name of the bucket.

```

```

#
# Returns:
#     The list of files in text format.
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function list_items_in_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api list-objects \
        --bucket "$bucket_name" \
        --output text \
        --query 'Contents[].{Key: Key, Size: Size}')

    # shellcheck disable=SC2181
    if [[ ${?} -eq 0 ]]; then
        echo "$response"
    else
        errecho "ERROR: AWS reports s3api list-objects operation failed.\n$response"
        return 1
    fi
}

```

- Untuk detail API, lihat [ListObjectsV2](#) di Referensi AWS CLI Perintah.

## PutObject

Contoh kode berikut menunjukkan cara menggunakan PutObject.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
```

```

# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function copy_file_to_bucket
#
# This function creates a file in the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file to.
#     $2 - The path and file name of the local file to copy to the bucket.
#     $3 - The key (name) to call the copy of the file in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_file_to_bucket() {
    local response bucket_name source_file destination_file_name
    bucket_name=$1
    source_file=$2
    destination_file_name=$3

    response=$(aws s3api put-object \
        --bucket "$bucket_name" \
        --body "$source_file" \
        --key "$destination_file_name")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "ERROR: AWS reports put-object operation failed.\n$response"
        return 1
    fi
}

```

- Untuk detail API, lihat [PutObject](#) di Referensi AWS CLI Perintah.

## Skenario

### Memulai bucket dan objek

Contoh kode berikut ini menunjukkan cara:

- Membuat bucket dan mengunggah file ke dalamnya.
- Mengunduh objek dari bucket.
- Menyalin objek ke subfolder di bucket.
- Membuat daftar objek dalam bucket.
- Menghapus objek bucket dan bucket tersebut.

### AWS CLI dengan skrip Bash

#### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####  
# function s3_getting_started  
#  
# This function creates, copies, and deletes S3 buckets and objects.  
#  
# Returns:  
#     0 - If successful.  
#     1 - If an error occurred.  
#####  
function s3_getting_started() {  
  {  
    if [ "$BUCKET_OPERATIONS_SOURCED" != "True" ]; then  
      cd bucket-lifecycle-operations || exit  
  
      source ./bucket_operations.sh  
      cd ..  
    fi  
  }  
  
  echo_repeat "*" 88
```



```
echo "Welcome to the Amazon S3 getting started demo."
echo_repeat "*" 88

local bucket_name
bucket_name=$(generate_random_name "doc-example-bucket")

local region_code
region_code=$(aws configure get region)

if create_bucket -b "$bucket_name" -r "$region_code"; then
    echo "Created demo bucket named $bucket_name"
else
    errecho "The bucket failed to create. This demo will exit."
    return 1
fi

local file_name
while [ -z "$file_name" ]; do
    echo -n "Enter a file you want to upload to your bucket: "
    get_input
    file_name=$get_input_result

    if [ ! -f "$file_name" ]; then
        echo "Could not find file $file_name. Are you sure it exists?"
        file_name=""
    fi
done

local key
key="$(basename "$file_name")"

local result=0
if copy_file_to_bucket "$bucket_name" "$file_name" "$key"; then
    echo "Uploaded file $file_name into bucket $bucket_name with key $key."
else
    result=1
fi

local destination_file
destination_file="$file_name.download"
if yes_no_input "Would you like to download $key to the file $destination_file?
(y/n) "; then
    if download_object_from_bucket "$bucket_name" "$destination_file" "$key"; then
```

```
    echo "Downloaded $key in the bucket $bucket_name to the file
$destination_file."
    else
        result=1
    fi
fi

if yes_no_input "Would you like to copy $key a new object key in your bucket? (y/
n) "; then
    local to_key
    to_key="demo/$key"
    if copy_item_in_bucket "$bucket_name" "$key" "$to_key"; then
        echo "Copied $key in the bucket $bucket_name to the $to_key."
    else
        result=1
    fi
fi

local bucket_items
bucket_items=$(list_items_in_bucket "$bucket_name")

# shellcheck disable=SC2181
if [[ $? -ne 0 ]]; then
    result=1
fi

echo "Your bucket contains the following items."
echo -e "Name\t\tSize"
echo "$bucket_items"

if yes_no_input "Delete the bucket, $bucket_name, as well as the objects in it?
(y/n) "; then
    bucket_items=$(echo "$bucket_items" | cut -f 1)

    if delete_items_in_bucket "$bucket_name" "$bucket_items"; then
        echo "The following items were deleted from the bucket $bucket_name"
        echo "$bucket_items"
    else
        result=1
    fi

    if delete_bucket "$bucket_name"; then
        echo "Deleted the bucket $bucket_name"
    else
```

```

    result=1
    fi
fi

return $result
}

```

Fungsi Amazon S3 yang digunakan dalam skenario ini.

```

#####
# function create-bucket
#
# This function creates the specified bucket in the specified AWS Region, unless
# it already exists.
#
# Parameters:
#     -b bucket_name  -- The name of the bucket to create.
#     -r region_code  -- The code for an AWS Region in which to
#                       create the bucket.
#
# Returns:
#     The URL of the bucket that was created.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function create_bucket() {
    local bucket_name region_code response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function create_bucket"
        echo "Creates an Amazon S3 bucket. You must supply a bucket name:"
        echo "  -b bucket_name    The name of the bucket. It must be globally unique."
        echo "  [-r region_code]  The code for an AWS Region in which the bucket is
created."
        echo ""
    }
}

# Retrieve the calling parameters.
while getopt "b:r:h" option; do

```

```
case "${option}" in
  b) bucket_name="${OPTARG}" ;;
  r) region_code="${OPTARG}" ;;
  h)
    usage
    return 0
    ;;
  \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done

if [[ -z "$bucket_name" ]]; then
  errecho "ERROR: You must provide a bucket name with the -b parameter."
  usage
  return 1
fi

local bucket_config_arg
# A location constraint for "us-east-1" returns an error.
if [[ -n "$region_code" ]] && [[ "$region_code" != "us-east-1" ]]; then
  bucket_config_arg="--create-bucket-configuration LocationConstraint=
$region_code"
fi

iecho "Parameters:\n"
iecho "    Bucket name:  $bucket_name"
iecho "    Region code:  $region_code"
iecho ""

# If the bucket already exists, we don't want to try to create it.
if (bucket_exists "$bucket_name"); then
  errecho "ERROR: A bucket with that name already exists. Try again."
  return 1
fi

# shellcheck disable=SC2086
response=$(aws s3api create-bucket \
  --bucket "$bucket_name" \
  $bucket_config_arg)
```

```

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports create-bucket operation failed.\n$response"
    return 1
fi
}

#####
# function copy_file_to_bucket
#
# This function creates a file in the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file to.
#     $2 - The path and file name of the local file to copy to the bucket.
#     $3 - The key (name) to call the copy of the file in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_file_to_bucket() {
    local response bucket_name source_file destination_file_name
    bucket_name=$1
    source_file=$2
    destination_file_name=$3

    response=$(aws s3api put-object \
        --bucket "$bucket_name" \
        --body "$source_file" \
        --key "$destination_file_name")

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports put-object operation failed.\n$response"
    return 1
fi
}

#####
# function download_object_from_bucket
#
# This function downloads an object in a bucket to a file.
#

```

```

# Parameters:
#     $1 - The name of the bucket to download the object from.
#     $2 - The path and file name to store the downloaded bucket.
#     $3 - The key (name) of the object in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function download_object_from_bucket() {
    local bucket_name=$1
    local destination_file_name=$2
    local object_name=$3
    local response

    response=$(aws s3api get-object \
        --bucket "$bucket_name" \
        --key "$object_name" \
        "$destination_file_name")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "ERROR: AWS reports put-object operation failed.\n$response"
        return 1
    fi
}

#####
# function copy_item_in_bucket
#
# This function creates a copy of the specified file in the same bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file from and to.
#     $2 - The key of the source file to copy.
#     $3 - The key of the destination file.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_item_in_bucket() {
    local bucket_name=$1
    local source_key=$2

```

```

local destination_key=$3
local response

response=$(aws s3api copy-object \
  --bucket "$bucket_name" \
  --copy-source "$bucket_name/$source_key" \
  --key "$destination_key")

# shellcheck disable=SC2181
if [[ $? -ne 0 ]]; then
  errecho "ERROR: AWS reports s3api copy-object operation failed.\n$response"
  return 1
fi
}

#####
# function list_items_in_bucket
#
# This function displays a list of the files in the bucket with each file's
# size. The function uses the --query parameter to retrieve only the key and
# size fields from the Contents collection.
#
# Parameters:
#   $1 - The name of the bucket.
#
# Returns:
#   The list of files in text format.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function list_items_in_bucket() {
  local bucket_name=$1
  local response

  response=$(aws s3api list-objects \
    --bucket "$bucket_name" \
    --output text \
    --query 'Contents[].{Key: Key, Size: Size}')

  # shellcheck disable=SC2181
  if [[ ${?} -eq 0 ]]; then
    echo "$response"
  else

```

```

    errecho "ERROR: AWS reports s3api list-objects operation failed.\n$response"
    return 1
fi
}

#####
# function delete_items_in_bucket
#
# This function deletes the specified list of keys from the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.
#     $2 - A list of keys in the bucket to delete.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_items_in_bucket() {
    local bucket_name=$1
    local keys=$2
    local response

    # Create the JSON for the items to delete.
    local delete_items
    delete_items="{\"Objects\":["
    for key in $keys; do
        delete_items="$delete_items{\"Key\": \"$key\"},"
    done
    delete_items=${delete_items%?} # Remove the final comma.
    delete_items="$delete_items]"

    response=$(aws s3api delete-objects \
        --bucket "$bucket_name" \
        --delete "$delete_items")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-object operation failed.\n$response"
        return 1
    fi
}

#####

```



```
# function delete_bucket
#
# This function deletes the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api delete-bucket \
        --bucket "$bucket_name")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-bucket failed.\n$response"
        return 1
    fi
}
```

- Untuk detail API, lihat topik berikut di Referensi Perintah AWS CLI .
  - [CopyObject](#)
  - [CreateBucket](#)
  - [DeleteBucket](#)
  - [DeleteObjects](#)
  - [GetObject](#)
  - [ListObjectsV2](#)
  - [PutObject](#)

## AWS STS contoh menggunakan AWS CLI dengan skrip Bash

Contoh kode berikut menunjukkan cara melakukan tindakan dan mengimplementasikan skenario umum dengan menggunakan skrip AWS Command Line Interface with Bash with AWS STS.

Tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Meskipun tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks pada skenario terkait dan contoh lintas layanan.

Skenario adalah contoh kode yang menunjukkan cara menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan yang sama.

Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan petunjuk tentang cara mengatur dan menjalankan kode dalam konteks.

Topik

- [Tindakan](#)

Tindakan

## AssumeRole

Contoh kode berikut menunjukkan cara menggunakan `AssumeRole`.

AWS CLI dengan skrip Bash

### Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [AWS Repositori Contoh Kode](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

#####
# function errecho
```

```

#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function sts_assume_role
#
# This function assumes a role in the AWS account and returns the temporary
# credentials.
#
# Parameters:
#     -n role_session_name -- The name of the session.
#     -r role_arn -- The ARN of the role to assume.
#
# Returns:
#     [access_key_id, secret_access_key, session_token]
#     And:
#     0 - If successful.
#     1 - If an error occurred.
#####
function sts_assume_role() {
    local role_session_name role_arn response
    local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function sts_assume_role"
    echo "Assumes a role in the AWS account and returns the temporary credentials:"
    echo "  -n role_session_name -- The name of the session."
    echo "  -r role_arn -- The ARN of the role to assume."
    echo ""
}

while getopt n:r:h option; do
    case "${option}" in
        n) role_session_name=${OPTARG} ;;
        r) role_arn=${OPTARG} ;;
        h)
            usage
            return 0
            ;;
    esac
done

```

```
\?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done

response=$(aws sts assume-role \
  --role-session-name "$role_session_name" \
  --role-arn "$role_arn" \
  --output text \
  --query "Credentials.[AccessKeyId, SecretAccessKey, SessionToken]")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}
```

- Untuk detail API, lihat [AssumeRole](#) di Referensi AWS CLI Perintah.

# Keamanan di AWS Command Line Interface

Keamanan cloud di AWS merupakan prioritas tertinggi. Sebagai seorang pelanggan AWS, Anda mendapatkan manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan dari organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara AWS dan Anda. [Model tanggung jawab bersama](#) model menjelaskan hal ini sebagai keamanan dari cloud dan keamanan dalam cloud:

- Keamanan cloud – AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan layanan AWS di AWS Cloud. AWS juga menyediakan layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga menguji dan memverifikasi secara berkala efektivitas keamanan kami sebagai bagian dari [Program Kepatuhan AWS](#). Untuk mempelajari tentang program kepatuhan yang berlaku AWS Command Line Interface, lihat [AWS Layanan dalam Lingkup oleh AWS Layanan Program Kepatuhan](#).
- Keamanan di cloud – Tanggung jawab Anda ditentukan oleh layanan AWS yang digunakan. Anda juga bertanggung jawab atas faktor lain termasuk sensitivitas data Anda, persyaratan perusahaan Anda, serta hukum dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AWS Command Line Interface (AWS CLI). Topik berikut menunjukkan kepada Anda cara mengonfigurasi AWS CLI untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS CLI untuk membantu Anda memantau dan mengamankan sumber daya AWS Anda.

## Topik

- [Perlindungan data di AWS CLI](#)
- [Manajemen Identitas dan Akses](#)
- [Validasi Kepatuhan untuk AWS Produk atau Layanan ini](#)
- [Ketahanan untuk AWS Produk atau Layanan ini](#)
- [Keamanan Infrastruktur untuk AWS Produk atau Layanan ini](#)
- [Menerapkan versi minimum TLS](#)

# Perlindungan data di AWS CLI

[Model tanggung jawab bersama](#) AWS diterapkan untuk perlindungan data AWS Command Line Interface. Sebagaimana dijelaskan dalam model ini, AWS bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda harus bertanggung jawab untuk memelihara kendali terhadap konten yang di-hosting pada infrastruktur ini. Anda juga bertanggung jawab atas konfigurasi keamanan dan tugas manajemen untuk berbagai layanan Layanan AWS yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, silakan lihat [Pertanyaan Umum Privasi Data](#). Untuk informasi tentang perlindungan data di Eropa, silakan lihat postingan blog [Model Tanggung Jawab Bersama AWS dan GDPR](#) di Blog Keamanan AWS.

Untuk tujuan perlindungan data, sebaiknya Anda melindungi kredensial Akun AWS dan menyiapkan akun pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara seperti itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugas mereka. Kami juga merekomendasikan agar Anda mengamankan data Anda dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk melakukan komunikasi dengan sumber daya AWS. Kami mensyaratkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Siapkan API dan log aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi enkripsi AWS, bersama dengan semua kontrol keamanan default dalam layanan Layanan AWS.
- Gunakan layanan keamanan terkelola lanjutan seperti Amazon Macie, yang membantu menemukan dan mengamankan data pribadi yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-2 ketika mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Untuk informasi selengkapnya tentang titik akhir FIPS yang tersedia, silakan lihat [Standar Pemrosesan Informasi Federal \(FIPS\) 140-2](#).

Sebaiknya Anda tidak memasukkan informasi rahasia atau sensitif, seperti alamat email pelanggan, ke dalam tanda atau bidang teks bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan AWS CLI atau lainnya Layanan AWS menggunakan konsol, APIAWS CLI, atau AWS SDK. Data apa pun yang Anda masukkan ke dalam tanda atau bidang teks bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau diagnostik. Saat Anda memberikan URL ke server eksternal, sebaiknya jangan menyertakan informasi kredensial di URL untuk memvalidasi permintaan Anda ke server tersebut.

## Enkripsi data

Fitur utama dari setiap layanan aman adalah bahwa informasi dienkripsi ketika tidak aktif digunakan.

### Enkripsi saat tidak aktif

AWS CLI sendiri tidak menyimpan data konsumen selain kredensialnya yang dibutuhkan untuk berinteraksi dengan layanan AWS atas nama pengguna.

Jika Anda menggunakan AWS CLI untuk membuka sebuah layanan AWS yang mentransmisikan data pelanggan ke komputer lokal Anda untuk penyimpanan, kemudian lihat bagian Keamanan & Kepatuhan dalam Panduan Pengguna layanan tersebut untuk informasi tentang bagaimana data tersebut disimpan, dilindungi, dan dienkripsi.

### Enkripsi dalam transit

Secara default, semua data yang dikirimkan dari komputer klien yang menjalankan titik akhir AWS CLI dan AWS layanan dienkripsi dengan mengirimkan semuanya melalui koneksi HTTPS/TLS.

Anda tidak perlu melakukan apapun untuk mengaktifkan penggunaan HTTPS/TLS. Itu selalu diaktifkan kecuali Anda secara eksplisit menonaktifkannya untuk perintah individual dengan menggunakan opsi baris `--no-verify-ssl` perintah.

## Manajemen Identitas dan Akses

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. AWS IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

### Topik

- [Audiens](#)
- [Mengautentikasi dengan identitas](#)
- [Mengelola akses menggunakan kebijakan](#)
- [Bagaimana Layanan AWS bekerja dengan IAM](#)
- [Memecahkan masalah AWS identitas dan akses](#)

## Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan. AWS

**Pengguna layanan** — Jika Anda menggunakan Layanan AWS untuk melakukan pekerjaan Anda, maka administrator Anda memberi Anda kredensial dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak AWS fitur untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur AWS, lihat [Memecahkan masalah AWS identitas dan akses](#) atau panduan pengguna yang Layanan AWS Anda gunakan.

**Administrator layanan** — Jika Anda bertanggung jawab atas AWS sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke AWS. Tugas Anda adalah menentukan AWS fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep Basic IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM AWS, lihat panduan pengguna yang Layanan AWS Anda gunakan.

**Administrator IAM** – Jika Anda adalah administrator IAM, Anda mungkin ingin belajar dengan lebih detail tentang cara Anda menulis kebijakan untuk mengelola akses ke AWS. Untuk melihat contoh kebijakan AWS berbasis identitas yang dapat Anda gunakan di IAM, lihat panduan pengguna yang Anda gunakan. Layanan AWS

## Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensial identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensial yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (IAM Identity Center), autentikasi masuk tunggal perusahaan Anda, dan kredensial Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas terfederasi, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.



Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat [Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS](#).

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang penggunaan metode yang disarankan untuk menandatangani permintaan sendiri, lihat [Menandatangani permintaan AWS API](#) di Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat [Autentikasi multi-faktor](#) dalam Panduan Pengguna AWS IAM Identity Center dan [Menggunakan autentikasi multi-faktor \(MFA\) dalam AWS](#) dalam Panduan Pengguna IAM.

## Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat [Tugas yang memerlukan kredensial pengguna root](#) dalam Panduan Pengguna IAM.

## Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, direktori Pusat Identitas AWS Directory Service, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensial yang disediakan melalui sumber identitas. Ketika identitas federasi mengakses Akun AWS, mereka mengambil peran, dan peran memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat Identitas IAM, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua aplikasi Akun AWS dan aplikasi Anda. Untuk informasi tentang Pusat Identitas IAM, lihat [Apakah itu Pusat Identitas IAM?](#) dalam Panduan Pengguna AWS IAM Identity Center .

## Pengguna dan grup IAM

[Pengguna IAM](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, kami merekomendasikan untuk mengandalkan kredensial sementara, bukan membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan tertentu yang memerlukan kredensial jangka panjang dengan pengguna IAM, kami merekomendasikan Anda merotasi kunci akses. Untuk informasi selengkapnya, lihat [Merotasi kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang](#) dalam Panduan Pengguna IAM.

[Grup IAM](#) adalah identitas yang menentukan sekumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat memiliki grup yang bernama IAMAdmins dan memberikan izin ke grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, lihat [Kapan harus membuat pengguna IAM \(bukan peran\)](#) dalam Panduan Pengguna IAM.

## Peran IAM

[Peran IAM](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil peran IAM untuk sementara AWS Management Console dengan [beralih peran](#). Anda dapat mengambil peran dengan memanggil operasi AWS CLI atau AWS API atau dengan menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, lihat [Menggunakan peran IAM](#) dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat [Membuat peran untuk Penyedia Identitas pihak ketiga](#) dalam Panduan Pengguna IAM. Jika menggunakan Pusat Identitas IAM, Anda harus mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM akan mengorelasikan set izin ke peran dalam IAM. Untuk informasi tentang set izin, lihat [Set izin](#) dalam Panduan Pengguna AWS IAM Identity Center .
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (prinsipal tepercaya) di akun lain untuk mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat [Bagaimana peran IAM berbeda dari kebijakan berbasis sumber daya](#) dalam Panduan Pengguna IAM.
- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Sebagai contoh, ketika Anda memanggil suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
- Sesi akses teruskan (FAS) — Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat [Sesi akses maju](#).
- Peran layanan – Peran layanan adalah [peran IAM](#) yang dijalankan oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat sebuah peran untuk mendelegasikan izin ke Layanan AWS](#) dalam Panduan pengguna IAM.

- Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan peran IAM untuk mengelola kredensi sementara untuk aplikasi yang berjalan pada instans EC2 dan membuat atau permintaan API. AWS CLI AWS Cara ini lebih dianjurkan daripada menyimpan kunci akses dalam instans EC2. Untuk menetapkan AWS peran ke instans EC2 dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat [Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan dalam instans Amazon EC2](#) dalam Panduan Pengguna IAM.

Untuk mempelajari apakah kita harus menggunakan peran IAM atau pengguna IAM, lihat [Kapan harus membuat peran IAM \(bukan pengguna\)](#) dalam Panduan Pengguna IAM.

## Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan isi dokumen kebijakan JSON, lihat [Gambaran umum kebijakan JSON](#) dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasinya. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan

tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut bisa mendapatkan informasi peran dari AWS Management Console, API AWS CLI, atau AWS API.

## Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat [Membuat kebijakan IAM](#) dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam Akun AWS. Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan yang dikelola atau kebijakan inline, lihat [Memilih antara kebijakan yang dikelola dan kebijakan inline](#) dalam Panduan Pengguna IAM.

## Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan prinsipal](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau Layanan AWS.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

## Daftar kontrol akses (ACL)

Daftar kontrol akses (ACL) mengendalikan prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACL serupa dengan kebijakan berbasis sumber daya, meskipun kebijakan tersebut tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung ACL. Untuk mempelajari ACL selengkapnya, lihat [Gambaran umum daftar kontrol akses \(ACL\)](#) dalam Panduan Developer Amazon Simple Storage Service.

## Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- **Batasan izin** – Batasan izin adalah fitur lanjutan tempat Anda mengatur izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas ke entitas IAM (pengguna IAM atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya tentang batasan izin, lihat [Batasan izin untuk entitas IAM](#) dalam Panduan Pengguna IAM.
- **Kebijakan kontrol layanan (SCP)** — SCP adalah kebijakan JSON yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur di organisasi, Anda dapat menerapkan kebijakan kontrol layanan (SCP) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas di akun anggota, termasuk masing-masing. Pengguna root akun AWS Untuk informasi selengkapnya tentang Organisasi dan SCP, lihat [Cara kerja SCP](#) dalam Panduan Pengguna AWS Organizations .
- **Kebijakan sesi** – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya, lihat [Kebijakan sesi](#) dalam Panduan Pengguna IAM.

## Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat [Logika evaluasi kebijakan](#) di Panduan Pengguna IAM.

## Bagaimana Layanan AWS bekerja dengan IAM

Untuk mendapatkan tampilan tingkat tinggi tentang cara Layanan AWS bekerja dengan sebagian besar fitur IAM, lihat [AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM](#).

Untuk mempelajari cara menggunakan spesifik Layanan AWS dengan IAM, lihat bagian keamanan dari Panduan Pengguna layanan yang relevan.

## Memecahkan masalah AWS identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS dan IAM.

### Topik

- [Saya tidak berwenang untuk melakukan tindakan di AWS](#)
- [Saya tidak berwenang untuk melakukan iam: PassRole](#)
- [Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS sumber daya saya](#)

## Saya tidak berwenang untuk melakukan tindakan di AWS

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya `my-example-widget` rekaan, tetapi tidak memiliki izin `aws:GetWidget` rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:  
aws:GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya `my-example-widget` dengan menggunakan tindakan `aws:GetWidget`.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

## Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran AWS.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di AWS. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

## Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACL), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mempelajari apakah AWS mendukung fitur ini, lihat [Bagaimana Layanan AWS bekerja dengan IAM](#).
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat [Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki](#) di Panduan Pengguna IAM.



- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat [Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga](#) dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat [Menyediakan akses ke pengguna terautentikasi eksternal \(federasi identitas\)](#) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara penggunaan kebijakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat [Bagaimana peran IAM berbeda dari kebijakan berbasis sumber daya](#) dalam Panduan Pengguna IAM.

## Validasi Kepatuhan untuk AWS Produk atau Layanan ini

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat [Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS](#) dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat [Program AWS Kepatuhan Program AWS](#) .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat [Mengunduh Laporan di AWS Artifact](#) .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan Memulai Cepat Keamanan dan Kepatuhan — Panduan](#) penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar AWS yang berfokus pada keamanan dan kepatuhan.
- [Arsitektur untuk Keamanan dan Kepatuhan HIPAA di Amazon Web Services](#) — Whitepaper ini menjelaskan bagaimana perusahaan dapat menggunakan AWS untuk membuat aplikasi yang memenuhi syarat HIPAA.

### Note

Tidak semua memenuhi Layanan AWS syarat HIPAA. Untuk informasi selengkapnya, lihat [Referensi Layanan yang Memenuhi Syarat HIPAA](#).

- [AWS Sumber Daya AWS](#) — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.

- [AWS Panduan Kepatuhan Pelanggan](#) - Memahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan ini merangkum praktik terbaik untuk mengamankan Layanan AWS dan memetakan panduan untuk kontrol keamanan di berbagai kerangka kerja (termasuk Institut Standar dan Teknologi Nasional (NIST), Dewan Standar Keamanan Industri Kartu Pembayaran (PCI), dan Organisasi Internasional untuk Standardisasi (ISO)).
- [Mengevaluasi Sumber Daya dengan Aturan](#) dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- [AWS Security Hub](#)— Ini Layanan AWS memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS. Security Hub menggunakan kontrol keamanan untuk sumber daya AWS Anda serta untuk memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat [Referensi kontrol Security Hub](#).
- [Amazon GuardDuty](#) — Ini Layanan AWS mendeteksi potensi ancaman terhadap beban kerja Akun AWS, kontainer, dan data Anda dengan memantau lingkungan Anda untuk aktivitas mencurigakan dan berbahaya. GuardDuty dapat membantu Anda mengatasi berbagai persyaratan kepatuhan, seperti PCI DSS, dengan memenuhi persyaratan deteksi intrusi yang diamanatkan oleh kerangka kerja kepatuhan tertentu.
- [AWS Audit Manager](#)Ini Layanan AWS membantu Anda terus mengaudit AWS penggunaan Anda untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap peraturan dan standar industri.

AWS Produk atau layanan ini mengikuti [model tanggung jawab bersama](#) melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat [halaman dokumentasi keamanan AWS layanan](#) dan [AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan](#).

## Ketahanan untuk AWS Produk atau Layanan ini

Infrastruktur AWS global dibangun di sekitar Wilayah AWS dan Availability Zones.

Wilayah AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan.

Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan

memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat [Infrastruktur AWS Global](#).

AWS Produk atau layanan ini mengikuti [model tanggung jawab bersama](#) melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat [halaman dokumentasi keamanan AWS layanan](#) dan [AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan](#).

## Keamanan Infrastruktur untuk AWS Produk atau Layanan ini

AWS Produk atau layanan ini menggunakan layanan terkelola, dan karenanya dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat [Keamanan AWS Cloud](#). Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat [Perlindungan Infrastruktur dalam Kerangka Kerja](#) yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses AWS Produk atau Layanan ini melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan prinsipal IAM. Atau Anda dapat menggunakan [AWS Security Token Service](#) (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

AWS Produk atau layanan ini mengikuti [model tanggung jawab bersama](#) melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat [halaman dokumentasi keamanan AWS layanan](#) dan [AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan](#).

## Menerapkan versi minimum TLS

Untuk menambahkan peningkatan keamanan saat berkomunikasi dengan AWS layanan, Anda harus menggunakan TLS 1.2 atau yang lebih baru. Saat Anda menggunakan AWS CLI, Python digunakan untuk mengatur versi TLS.

AWS CLI versi 2 menggunakan skrip Python internal yang dikompilasi untuk menggunakan minimal TLS 1.2 saat layanan yang diajak bicara mendukungnya. Selama Anda menggunakan versi 2 AWS CLI, tidak ada langkah lebih lanjut yang diperlukan untuk menegakkan minimum ini.

# Memecahkan masalah kesalahan AWS CLI

Bagian ini mencakup kesalahan umum dan langkah-langkah pemecahan masalah yang harus diikuti untuk menyelesaikan masalah Anda. Kami menyarankan untuk mengikuti [pemecahan masalah umum](#) terlebih dahulu.

## Daftar Isi

- [Pemecahan masalah umum untuk dicoba terlebih dahulu](#)
  - [Periksa pemformatan AWS CLI perintah Anda](#)
  - [Periksa AWS CLI perintah yang Wilayah AWS Anda gunakan](#)
  - [Konfirmasikan bahwa Anda menjalankan versi terbaru AWS CLI](#)
  - [Gunakan --debug opsi](#)
  - [Aktifkan dan tinjau log riwayat AWS CLI perintah](#)
  - [Konfirmasikan bahwa Anda telah AWS CLI dikonfigurasi](#)
- [Perintah tidak ditemukan kesalahan](#)
- [Perintah aws --version "" mengembalikan versi yang berbeda dari yang Anda instal](#)
- [Perintah aws --version "" mengembalikan versi setelah mencopot pemasangan AWS CLI](#)
- [AWS CLI Diproses perintah dengan nama parameter yang tidak lengkap](#)
- [Akses kesalahan ditolak](#)
- [Kredensial dan kesalahan kunci tidak valid](#)
- [Tanda tangan tidak cocok dengan kesalahan](#)
- [Kesalahan sertifikat SSL](#)
- [Kesalahan JSON tidak valid](#)
- [Sumber daya tambahan](#)

## Pemecahan masalah umum untuk dicoba terlebih dahulu

Jika Anda menerima kesalahan atau mengalami masalah dengan AWS CLI, kami menyarankan tips umum berikut untuk membantu Anda memecahkan masalah.

[Kembali ke atas](#)

## Periksa pemformatan AWS CLI perintah Anda

Jika Anda menerima kesalahan yang menunjukkan bahwa perintah tidak ada, atau tidak mengenali parameter (`Parameter validation failed`) yang menurut dokumentasi tersedia, maka perintah Anda mungkin salah diformat. Kami menyarankan Anda memeriksa hal-hal berikut:

- Periksa perintah Anda untuk kesalahan ejaan dan pemformatan.
- Konfirmasikan semua [tanda kutip dan pelarian yang sesuai untuk terminal Anda](#) benar dalam perintah Anda.
- Hasilkan [AWS CLI kerangka](#) untuk mengkonfirmasi struktur perintah Anda.
- Untuk JSON, lihat [pemecahan masalah tambahan untuk](#) nilai JSON. Jika Anda mengalami masalah dengan terminal Anda memproses pemformatan JSON, kami sarankan untuk melewati aturan kutipan terminal dengan menggunakan [Blobs untuk meneruskan data JSON langsung ke file](#). AWS CLI

Untuk informasi selengkapnya tentang bagaimana perintah tertentu harus disusun, lihat [versi 2 panduan referensi](#).

[Kembali ke atas](#)

## Periksa AWS CLI perintah yang Wilayah AWS Anda gunakan

### Note

Anda harus menentukan Wilayah AWS saat menggunakan AWS CLI, baik secara eksplisit atau dengan menyetel Wilayah default. Untuk daftar semua Wilayah AWS yang dapat Anda tentukan, lihat [AWS Wilayah dan Titik Akhir](#) di Referensi Umum Amazon Web Services. Wilayah AWS Penunjuk yang digunakan oleh AWS CLI adalah nama yang sama dengan yang Anda lihat di AWS Management Console URL dan titik akhir layanan.

Kesalahan atau hasil yang tidak terduga dapat terjadi jika Layanan AWS tidak tersedia untuk yang Anda tentukan Wilayah AWS atau sumber daya Anda berada di tempat yang berbeda Wilayah AWS. Dalam urutan prioritas, diatur dengan Wilayah AWS cara-cara berikut:

- Opsi baris `--region` perintah.
- Variabel [AWS\\_REGION](#) lingkungan yang kompatibel dengan SDK.

- Variabel [AWS\\_DEFAULT\\_REGION](#) lingkungan.
- Pengaturan [region](#) profil.

Konfirmasikan bahwa Anda menggunakan yang benar Wilayah AWS untuk sumber daya Anda.

[Kembali ke atas](#)

## Konfirmasikan bahwa Anda menjalankan versi terbaru AWS CLI

Jika Anda menerima kesalahan yang menunjukkan bahwa perintah tidak ada, atau tidak mengenali parameter yang menurut panduan referensi referensi [AWS CLI versi 2](#) tersedia, pertama-tama konfirmasikan bahwa perintah Anda diformat dengan benar. Jika pemformatannya benar, maka kami sarankan Anda meningkatkan ke versi terbaru dari file. AWS CLI Versi terbaru dari AWS CLI dirilis hampir setiap hari kerja. AWS Layanan, fitur, dan parameter baru diperkenalkan dalam versi baru AWS CLI. Satu-satunya cara untuk mendapatkan akses ke layanan, fitur, atau parameter baru tersebut adalah dengan meningkatkan ke versi yang dirilis setelah elemen itu pertama kali diperkenalkan.

Bagaimana Anda memperbarui versi Anda AWS CLI tergantung pada bagaimana Anda awalnya menginstalnya seperti yang dijelaskan dalam [the section called “Instal/Perbarui”](#).

Jika Anda menggunakan salah satu installer yang dibundel, Anda mungkin perlu menghapus instalasi yang ada sebelum mengunduh dan menginstal versi terbaru untuk sistem operasi Anda.

[Kembali ke atas](#)

## Gunakan **--debug** opsi

Ketika AWS CLI melaporkan kesalahan yang tidak langsung Anda pahami, atau menghasilkan hasil yang tidak Anda harapkan, Anda bisa mendapatkan detail lebih lanjut tentang kesalahan dengan menjalankan perintah lagi dengan **--debug** opsi. Dengan opsi ini, AWS CLI output merinci tentang setiap langkah yang diperlukan untuk memproses perintah Anda. Detail dalam output dapat membantu Anda menentukan kapan kesalahan terjadi dan memberikan petunjuk tentang di mana itu dimulai.

Anda dapat mengirim output ke file teks untuk ditinjau nanti, atau untuk dikirim AWS Support ketika diminta.

Saat Anda menyertakan **--debug** opsi, beberapa detailnya meliputi:

- Mencari kredensial
- Mengurai parameter yang disediakan
- Membangun permintaan yang dikirim ke server AWS
- Isi permintaan yang dikirim ke AWS
- Isi respon mentah
- Output yang diformat

Berikut adalah contoh perintah yang dijalankan dengan dan tanpa `--debug` opsi.

```
$ aws iam list-groups --profile MyTestProfile
{
  "Groups": [
    {
      "Path": "/",
      "GroupName": "MyTestGroup",
      "GroupId": "AGPA0123456789EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/MyTestGroup",
      "CreateDate": "2019-08-12T19:34:04Z"
    }
  ]
}
```

```
$ aws iam list-groups --profile MyTestProfile --debug
2019-08-12 12:36:18,305 - MainThread - awscli.clidriver - DEBUG - CLI version: aws-
cli/1.16.215 Python/3.7.3 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205
2019-08-12 12:36:18,305 - MainThread - awscli.clidriver - DEBUG - Arguments entered to
CLI: ['iam', 'list-groups', '--debug']
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function add_scalar_parsers at 0x7fdf173161e0>
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function register_uri_param_handler at 0x7fdf17dec400>
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function inject_assume_role_provider_cache at
0x7fdf17da9378>
2019-08-12 12:36:18,307 - MainThread - botocore.credentials - DEBUG - Skipping
environment variable credential check because profile name was explicitly set.
2019-08-12 12:36:18,307 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function attach_history_handler at 0x7fdf173ed9d8>
```



```
2019-08-12 12:36:18,308 - MainThread - botocore.loaders - DEBUG - Loading JSON
file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
service-2.json
2019-08-12 12:36:18,317 - MainThread - botocore.hooks - DEBUG - Event building-command-
table.iam: calling handler <function add_waiters at 0x7fdf1731a840>
2019-08-12 12:36:18,320 - MainThread - botocore.loaders - DEBUG - Loading JSON
file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
waiters-2.json
2019-08-12 12:36:18,321 - MainThread - awscli.clidriver - DEBUG - OrderedDict([('path-
prefix', <awscli.arguments.CLIArgument object at 0x7fdf171ac780>), ('marker',
<awscli.arguments.CLIArgument object at 0x7fdf171b09e8>), ('max-items',
<awscli.arguments.CLIArgument object at 0x7fdf171b09b0>)])
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_streaming_output_arg at
0x7fdf17316510>
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_cli_input_json at
0x7fdf17da9d90>
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function unify_paging_params at
0x7fdf17328048>
2019-08-12 12:36:18,326 - MainThread - botocore.loaders - DEBUG - Loading JSON
file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
paginator-1.json
2019-08-12 12:36:18,326 - MainThread - awscli.customizations.paginate - DEBUG -
Modifying paging parameters for operation: ListGroups
2019-08-12 12:36:18,326 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_generate_skeleton at
0x7fdf1737eae8>
2019-08-12 12:36:18,326 - MainThread - botocore.hooks - DEBUG - Event
before-building-argument-table-parser.iam.list-groups: calling handler
<bound method OverrideRequiredArgsArgument.override_required_args of
<awscli.customizations.cliinputjson.CliInputJSONArgument object at 0x7fdf171b0a58>>
2019-08-12 12:36:18,327 - MainThread - botocore.hooks - DEBUG - Event
before-building-argument-table-parser.iam.list-groups: calling handler
<bound method GenerateCliSkeletonArgument.override_required_args of
<awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
0x7fdf171c5978>>
2019-08-12 12:36:18,327 - MainThread - botocore.hooks - DEBUG - Event operation-
args-parsed.iam.list-groups: calling handler functools.partial(<function
check_should_enable_pagination at 0x7fdf17328158>, ['marker', 'max-items'], {'max-
items': <awscli.arguments.CLIArgument object at 0x7fdf171b09b0>}, OrderedDict([('path-
prefix', <awscli.arguments.CLIArgument object at 0x7fdf171ac780>), ('marker',
<awscli.arguments.CLIArgument object at 0x7fdf171b09e8>), ('max-items',
```

```
<awscli.customizations.paginate.PageArgument object at 0x7fdf171c58d0>), ('cli-  
input-json', <awscli.customizations.cliinputjson.CliInputJSONArgument object at  
0x7fdf171b0a58>), ('starting-token', <awscli.customizations.paginate.PageArgument  
object at 0x7fdf171b0a20>), ('page-size', <awscli.customizations.paginate.PageArgument  
object at 0x7fdf171c5828>), ('generate-cli-skeleton',  
<awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at  
0x7fdf171c5978>]]))  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-  
arg.iam.list-groups.path-prefix: calling handler <awscli.paramfile.URIArgumentHandler  
object at 0x7fdf1725c978>  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-  
arg.iam.list-groups.marker: calling handler <awscli.paramfile.URIArgumentHandler object  
at 0x7fdf1725c978>  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-  
arg.iam.list-groups.max-items: calling handler <awscli.paramfile.URIArgumentHandler  
object at 0x7fdf1725c978>  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG -  
Event load-cli-arg.iam.list-groups.cli-input-json: calling handler  
<awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG -  
Event load-cli-arg.iam.list-groups.starting-token: calling handler  
<awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-  
arg.iam.list-groups.page-size: calling handler <awscli.paramfile.URIArgumentHandler  
object at 0x7fdf1725c978>  
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event  
load-cli-arg.iam.list-groups.generate-cli-skeleton: calling handler  
<awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>  
2019-08-12 12:36:18,329 - MainThread - botocore.hooks - DEBUG  
- Event calling-command.iam.list-groups: calling handler  
<bound method CliInputJSONArgument.add_to_call_parameters of  
<awscli.customizations.cliinputjson.CliInputJSONArgument object at 0x7fdf171b0a58>>  
2019-08-12 12:36:18,329 - MainThread - botocore.hooks - DEBUG -  
Event calling-command.iam.list-groups: calling handler <bound  
method GenerateCliSkeletonArgument.generate_json_skeleton of  
<awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at  
0x7fdf171c5978>>  
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for  
credentials via: assume-role  
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for  
credentials via: assume-role-with-web-identity  
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for  
credentials via: shared-credentials-file
```

```
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - INFO - Found credentials
in shared credentials file: ~/.aws/credentials
2019-08-12 12:36:18,330 - MainThread - botocore.loaders - DEBUG - Loading JSON file: /
home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/endpoints.json
2019-08-12 12:36:18,334 - MainThread - botocore.hooks - DEBUG - Event choose-service-
name: calling handler <function handle_service_name_alias at 0x7fdf1898eb70>
2019-08-12 12:36:18,337 - MainThread - botocore.hooks - DEBUG - Event creating-client-
class.iam: calling handler <function add_generate_presigned_url at 0x7fdf18a028c8>
2019-08-12 12:36:18,337 - MainThread - botocore.regions - DEBUG - Using partition
endpoint for iam, us-west-2: aws-global
2019-08-12 12:36:18,337 - MainThread - botocore.args - DEBUG - The s3 config key is not
a dictionary type, ignoring its value of: None
2019-08-12 12:36:18,340 - MainThread - botocore.endpoint - DEBUG - Setting iam timeout
as (60, 60)
2019-08-12 12:36:18,341 - MainThread - botocore.loaders - DEBUG - Loading JSON file: /
home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/_retry.json
2019-08-12 12:36:18,341 - MainThread - botocore.client - DEBUG - Registering retry
handlers for service: iam
2019-08-12 12:36:18,342 - MainThread - botocore.hooks - DEBUG - Event before-
parameter-build.iam.ListGroups: calling handler <function generate_idempotent_uuid at
0x7fdf189b10d0>
2019-08-12 12:36:18,342 - MainThread - botocore.hooks - DEBUG - Event before-
call.iam.ListGroups: calling handler <function inject_api_version_header_if_needed at
0x7fdf189b2a60>
2019-08-12 12:36:18,343 - MainThread - botocore.endpoint - DEBUG - Making
request for OperationModel(name=ListGroups) with params: {'url_path': '/',
'query_string': '', 'method': 'POST', 'headers': {'Content-Type': 'application/x-
www-form-urlencoded; charset=utf-8', 'User-Agent': 'aws-cli/1.16.215 Python/3.7.3
Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205'}, 'body': {'Action':
'ListGroups', 'Version': '2010-05-08'}, 'url': 'https://iam.amazonaws.com/',
'context': {'client_region': 'aws-global', 'client_config': <botoconfig.Config
object at 0x7fdf16e9a4a8>, 'has_streaming_input': False, 'auth_type': None}}
2019-08-12 12:36:18,343 - MainThread - botocore.hooks - DEBUG - Event request-
created.iam.ListGroups: calling handler <bound method RequestSigner.handler of
<botoconfig.signers.RequestSigner object at 0x7fdf16e9a470>>
2019-08-12 12:36:18,343 - MainThread - botocore.hooks - DEBUG - Event choose-
signer.iam.ListGroups: calling handler <function set_operation_specific_signer at
0x7fdf18996f28>
2019-08-12 12:36:18,343 - MainThread - botocore.auth - DEBUG - Calculating signature
using v4 auth.
2019-08-12 12:36:18,343 - MainThread - botocore.auth - DEBUG - CanonicalRequest:
POST
/
```

```

content-type:application/x-www-form-urlencoded; charset=utf-8
host:iam.amazonaws.com
x-amz-date:20190812T193618Z

content-type;host;x-amz-date
5f776d91EXAMPLE9b8cb5eb5d6d4a787a33ae41c8cd6eEXAMPLEEca69080e1e1f
2019-08-12 12:36:18,344 - MainThread - botocore.auth - DEBUG - StringToSign:
AWS4-HMAC-SHA256
20190812T193618Z
20190812/us-east-1/iam/aws4_request
ab7e367eEXAMPLE2769f178ea509978cf8bfa054874b3EXAMPLE8d043fab6cc9
2019-08-12 12:36:18,344 - MainThread - botocore.auth - DEBUG - Signature:
d85a0EXAMPLEeb40164f2f539cdc76d4f294fe822EXAMPLE18ad1ddf58a1a3ce7
2019-08-12 12:36:18,344 - MainThread - botocore.endpoint - DEBUG - Sending
http request: <AWSPreparedRequest stream_output=False, method=POST,
url=https://iam.amazonaws.com/, headers={'Content-Type': b'application/
x-www-form-urlencoded; charset=utf-8', 'User-Agent': b'aws-cli/1.16.215
Python/3.7.3 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205',
'X-Amz-Date': b'20190812T193618Z', 'Authorization': b'AWS4-HMAC-SHA256
Credential=AKIA01234567890EXAMPLE-east-1/iam/aws4_request, SignedHeaders=content-
type;host;x-amz-date, Signature=d85a07692aceb401EXAMPLEa1b18ad1ddf58a1a3ce7EXAMPLE',
'Content-Length': '36'}>
2019-08-12 12:36:18,344 - MainThread - urllib3.util.retry - DEBUG - Converted retries
value: False -> Retry(total=False, connect=None, read=None, redirect=0, status=None)
2019-08-12 12:36:18,344 - MainThread - urllib3.connectionpool - DEBUG - Starting new
HTTPS connection (1): iam.amazonaws.com:443
2019-08-12 12:36:18,664 - MainThread - urllib3.connectionpool - DEBUG - https://
iam.amazonaws.com:443 "POST / HTTP/1.1" 200 570
2019-08-12 12:36:18,664 - MainThread - botocore.parsers - DEBUG - Response headers:
{'x-amzn-RequestId': '74c11606-bd38-11e9-9c82-559da0adb349', 'Content-Type': 'text/
xml', 'Content-Length': '570', 'Date': 'Mon, 12 Aug 2019 19:36:18 GMT'}
2019-08-12 12:36:18,664 - MainThread - botocore.parsers - DEBUG - Response body:
b'<ListGroupResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">\n
<ListGroupResult>\n  <IsTruncated>>false</IsTruncated>\n  <Groups>\n
  <member>\n    <Path></Path>\n    <GroupName>MyTestGroup</GroupName>
\n    <Arn>arn:aws:iam::123456789012:group/MyTestGroup</Arn>\n
  <GroupId>AGPA1234567890EXAMPLE</GroupId>\n    <CreateDate>2019-08-12T19:34:04Z</
CreateDate>\n  </member>\n  </Groups>\n </ListGroupResult>\n
  <ResponseMetadata>\n    <RequestId>74c11606-bd38-11e9-9c82-559da0adb349</RequestId>\n
  </ResponseMetadata>\n</ListGroupResponse>\n'
2019-08-12 12:36:18,665 - MainThread - botocore.hooks - DEBUG - Event needs-
retry.iam.ListGroups: calling handler <botocore.retryhandler.RetryHandler object at
0x7fdf16e9a780>
2019-08-12 12:36:18,665 - MainThread - botocore.retryhandler - DEBUG - No retry needed.

```

```
2019-08-12 12:36:18,665 - MainThread - botocore.hooks - DEBUG - Event after-
call.iam.ListGroups: calling handler <function json_decode_policies at 0x7fdf189b1d90>
{
  "Groups": [
    {
      "Path": "/",
      "GroupName": "MyTestGroup",
      "GroupId": "AGPA123456789012EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/MyTestGroup",
      "CreateDate": "2019-08-12T19:34:04Z"
    }
  ]
}
```

[Kembali ke atas](#)

## Aktifkan dan tinjau log riwayat AWS CLI perintah

Anda dapat mengaktifkan log riwayat AWS CLI perintah menggunakan pengaturan [cli\\_history](#) file. Setelah mengaktifkan pengaturan ini, AWS CLI catatan sejarah aws perintah.

Anda dapat mencantumkan riwayat Anda menggunakan `aws history list` perintah, dan menggunakan `command_ids` yang dihasilkan dalam `aws history show` perintah untuk detailnya. Untuk informasi lebih lanjut lihat di panduan AWS CLI referensi.

Saat Anda menyertakan `--debug` opsi, beberapa detailnya meliputi:

- Panggilan API dilakukan ke botocore
- Kode status
- Tanggapan HTTP
- Header
- Kode pengembalian

Anda dapat menggunakan informasi ini untuk mengonfirmasi data paramater dan panggilan API berperilaku seperti yang Anda harapkan, dan kemudian dapat menyimpulkan pada langkah apa dalam proses perintah Anda gagal.

[Kembali ke atas](#)

## Konfirmasikan bahwa Anda telah AWS CLI dikonfigurasi

Berbagai kesalahan dapat terjadi jika `credentials` file config dan Anda atau pengguna IAM atau peran Anda tidak dikonfigurasi dengan benar. Untuk informasi selengkapnya tentang menyelesaikan kesalahan dengan config dan `credentials` file atau pengguna atau peran IAM Anda, lihat [the section called “Akses kesalahan ditolak”](#) dan [the section called “Kredensyal dan kesalahan kunci tidak valid”](#)

[Kembali ke atas](#)

## Perintah tidak ditemukan kesalahan

Kesalahan ini berarti bahwa sistem operasi tidak dapat menemukan AWS CLI perintah. Instalasi mungkin tidak lengkap atau memerlukan pembaruan.

Kemungkinan penyebabnya: Anda mencoba menggunakan AWS CLI fitur yang lebih baru dari versi yang diinstal, atau memiliki format yang salah

Contoh teks kesalahan:

```
$ aws s3 copy
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help
aws: error: argument subcommand: Invalid choice, valid choices are:

ls                | website
cp                | mv
.....
```

Berbagai kesalahan dapat terjadi jika perintah Anda diformat secara tidak benar atau Anda menggunakan versi sebelumnya dari sebelum fitur dirilis. Untuk informasi lebih lanjut tentang menyelesaikan kesalahan di sekitar dua masalah ini, lihat [the section called “Periksa pemformatan AWS CLI perintah Anda”](#) dan [the section called “Konfirmasikan bahwa Anda menjalankan versi terbaru AWS CLI”](#).

[Kembali ke atas](#)

Kemungkinan penyebabnya: Terminal perlu dihidupkan ulang setelah instalasi

Contoh teks kesalahan:

```
$ aws --version
command not found: aws
```

Jika `aws` perintah tidak dapat ditemukan setelah pertama kali menginstal atau memperbarui AWS CLI, Anda mungkin perlu me-restart terminal Anda untuk mengenali PATH pembaruan apa pun.

[Kembali ke atas](#)

Kemungkinan penyebabnya: AWS CLI Tidak sepenuhnya diinstal

Contoh teks kesalahan:

```
$ aws --version
command not found: aws
```

Jika `aws` perintah tidak dapat ditemukan setelah pertama kali menginstal atau memperbarui AWS CLI, itu mungkin belum sepenuhnya diinstal. Coba instal ulang dengan mengikuti langkah-langkah untuk platform Anda. [the section called “Instal/Perbarui”](#)

[Kembali ke atas](#)

Kemungkinan penyebabnya: AWS CLI Tidak memiliki izin (Linux)

Jika `aws` perintah tidak dapat ditemukan setelah pertama kali menginstal atau memperbarui AWS CLI di Linux, itu mungkin tidak memiliki `execute` izin untuk folder yang diinstal. Jalankan perintah berikut dengan PATH ke AWS CLI instalasi Anda, untuk memberikan [chmod](#) izin ke AWS CLI:

```
$ sudo chmod -R 755 /usr/local/aws-cli/
```

[Kembali ke atas](#)

Kemungkinan penyebabnya: Sistem operasi tidak **PATH** diperbarui selama instalasi

Contoh teks kesalahan:

```
$ aws --version
```

```
command not found: aws
```

Anda mungkin perlu menambahkan `aws` executable ke variabel `PATH` lingkungan sistem operasi Anda. Untuk menambahkan AWS CLI ke `PATH`, gunakan instruksi berikut untuk sistem operasi Anda.

### Linux and macOS

1. Temukan skrip profil shell Anda di direktori pengguna Anda. Jika Anda tidak yakin shell mana yang Anda miliki, jalankan `echo $SHELL`.

```
$ ls -a ~  
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash – `.bash_profile`, `.profile`, atau `.bash_login`
  - Zsh – `.zshrc`
  - Tcsh – `.tcshrc`, `.cshrc`, atau `.login`
2. Tambahkan perintah ekspor ke penulisan profil Anda. Perintah berikut menambahkan bin lokal Anda ke `PATH` variabel saat ini.

```
export PATH=/usr/local/bin:$PATH
```

3. Muat ulang profil yang diperbarui ke sesi Anda saat ini.

```
$ source ~/.bash_profile
```

### Windows

1. Dalam Prompt Perintah Windows, gunakan `where` perintah dengan `/R path` parameter untuk menemukan lokasi `aws` file. Hasilnya mengembalikan semua folder yang berisi `aws`.

```
C:\> where /R c:\ aws  
c:\Program Files\Amazon\AWSCLIV2\aws.exe  
...
```

Secara default, AWS CLI versi 2 terletak di:

```
c:\Program Files\Amazon\AWSCLIV2\aws.exe
```



2. Tekan tombol Windows dan masukkan **environment variables**.
3. Dari daftar saran, pilih Edit variabel lingkungan untuk akun Anda.
4. Pilih PATH, lalu pilih Edit.
5. Tambahkan jalur yang Anda temukan di langkah pertama ke bidang Nilai variabel, misalnya, **C:\Program Files\Amazon\AWSCLIV2\aws.exe**.
6. Pilih OK dua kali untuk menggunakan pengaturan baru.
7. Tutup semua command prompt yang sedang berjalan dan buka kembali jendela command prompt.

[Kembali ke atas](#)

## Perintah `aws --version ""` mengembalikan versi yang berbeda dari yang Anda instal

Terminal Anda mungkin mengembalikan yang berbeda PATH AWS CLI dari yang Anda harapkan.

Kemungkinan penyebabnya: Terminal perlu dimulai ulang setelah menginstal

Jika `aws` perintah menunjukkan versi yang salah, Anda mungkin perlu me-restart terminal Anda agar dapat mengenali PATH pembaruan apa pun. Semua terminal terbuka harus ditutup, bukan hanya terminal aktif Anda.

[Kembali ke atas](#)

Kemungkinan penyebabnya: Sistem perlu dimulai ulang setelah menginstal

Jika `aws` perintah menunjukkan versi yang salah dan memulai ulang terminal tidak berfungsi, Anda mungkin perlu me-restart sistem Anda agar dapat mengenali pembaruan AndaPATH.


[Kembali ke atas](#)

Kemungkinan penyebabnya: Anda memiliki beberapa versi AWS CLI

Jika Anda memperbarui AWS CLI dan menggunakan metode penginstalan yang berbeda dari instalasi yang sudah ada sebelumnya, itu mungkin menyebabkan beberapa versi diinstal. Misalnya, jika di Linux atau macOS Anda gunakan `pip` untuk instalasi Anda saat ini, tetapi mencoba memperbarui menggunakan file `.pkg` instal, ini dapat menyebabkan beberapa konflik terutama dengan Anda PATH menunjuk ke versi lama.

Untuk mengatasi ini, [hapus semua versi AWS CLI](#) dan lakukan instalasi bersih.

Setelah menghapus semua versi, ikuti petunjuk yang sesuai untuk sistem operasi Anda untuk menginstal versi yang Anda inginkan dari [AWS CLI versi 1](#) atau [AWS CLI versi 2](#).

 Note

Jika ini terjadi setelah Anda menginstal AWS CLI versi 2 dengan penginstalan AWS CLI versi 1 yang sudah ada sebelumnya, ikuti petunjuk migrasi di Petunjuk 1. [the section called "Instruksi migrasi"](#)

[Kembali ke atas](#)

## Perintah `aws --version ""` mengembalikan versi setelah mencopot pemasangan AWS CLI

Ini sering terjadi ketika masih ada yang AWS CLI diinstal di suatu tempat di sistem Anda.

Kemungkinan penyebabnya: Terminal perlu dimulai ulang setelah mencopot pemasangan

Jika `aws --version` perintah masih berfungsi, Anda mungkin perlu me-restart terminal Anda agar dapat mengenali pembaruan terminal apa pun.

[Kembali ke atas](#)

Kemungkinan penyebabnya: Anda memiliki beberapa versi AWS CLI pada sistem Anda, atau tidak menggunakan metode `uninstall` yang sama dengan yang Anda gunakan untuk menginstal AWS CLI

AWS CLI Mungkin tidak dihapus dengan benar jika Anda menghapus instalasi AWS CLI menggunakan metode yang berbeda dari yang Anda gunakan untuk menginstalnya, atau jika Anda menginstal beberapa versi. Misalnya, jika Anda menggunakan `pip` untuk instalasi Anda saat ini, Anda harus menggunakan `pip` untuk menghapus instalannya. Untuk mengatasi ini, hapus instalasi AWS CLI menggunakan metode yang sama yang Anda gunakan untuk menginstalnya.

1. 4. Ikuti instruksi yang sesuai untuk sistem operasi Anda dan metode instalasi asli Anda untuk menghapus instalasi [AWS CLI versi 1](#) dan [AWS CLI versi 2](#).
2. Tutup semua terminal yang telah Anda buka.

3. Buka terminal pilihan Anda, masukkan perintah berikut dan konfirmasi bahwa tidak ada versi yang dikembalikan.

```
$ aws --version
command not found: aws
```

Jika Anda masih memiliki versi yang tercantum dalam output, kemungkinan besar diinstal menggunakan metode yang berbeda atau ada beberapa versi. AWS CLI Jika Anda tidak tahu metode mana yang Anda instal AWS CLI, ikuti petunjuk untuk setiap metode uninstall untuk [AWS CLI versi 1](#) dan [AWS CLI versi 2](#) yang sesuai dengan sistem operasi Anda sampai tidak ada output versi yang diterima.

#### Note

Jika Anda menggunakan manajer paket untuk menginstal AWS CLI (pip,, aptbrew, dll.), Anda harus menggunakan manajer paket yang sama untuk menghapusnya. Pastikan untuk mengikuti instruksi yang diberikan oleh manajer paket tentang cara menghapus semua versi paket.

[Kembali ke atas](#)

## AWS CLI Diproses perintah dengan nama parameter yang tidak lengkap

Kemungkinan penyebabnya: Anda menggunakan singkatan AWS CLI parameter yang diakui

Karena AWS CLI dibangun menggunakan Python, menggunakan `argparse` pustaka Python, termasuk argumennya. AWS CLI [allow\\_abbrev](#) Singkatan parameter dikenali oleh AWS CLI dan diproses.

Contoh perintah berikut mengubah nama CloudFormation stack. Parameter `--change-set-n` diakui sebagai singkatan dari `--change-set-name`, dan AWS CLI memproses perintah.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set-n my-change-set
```

Ketika singkatan Anda bisa berupa beberapa perintah, parameter tidak akan dikenali sebagai singkatan.

Contoh perintah berikut mengubah nama CloudFormation stack. Parameter tidak **--change-set-** dikenali sebagai singkatan, karena ada beberapa parameter yang bisa menjadi singkatan dari, seperti **--change-set-name** dan **--change-set-type**. Oleh karena itu AWS CLI tidak memproses perintah.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set- my-change-set
```

### Warning

Jangan sengaja menggunakan singkatan parameter. Mereka tidak dapat diandalkan dan tidak kompatibel ke belakang. Jika ada parameter baru yang ditambahkan ke perintah yang membingungkan singkatan Anda, itu akan merusak perintah Anda.

Selain itu, jika parameter adalah argumen nilai tunggal, itu dapat menyebabkan perilaku tak terduga dengan perintah Anda. Jika beberapa contoh argumen nilai tunggal diteruskan, hanya contoh terakhir yang akan berjalan. Dalam contoh berikut, parameter **--filters** adalah argumen bernilai tunggal. Parameter **--filters** dan **--filter** ditentukan. **--filterParameter** adalah singkatan dari **--filters**. Ini menyebabkan dua contoh **--filters** diterapkan dan hanya **--filter** argumen terakhir yang berlaku.

```
$ aws ec2 describe-vpc-peering-connections \  
  --filters Name=tag:TagName,Values=VpcPeeringConnection \  
  --filter Name=status-code,Values=active
```

Konfirmasikan bahwa Anda menggunakan parameter yang valid sebelum menjalankan perintah untuk mencegah perilaku yang tidak terduga.

[Kembali ke atas](#)

## Akses kesalahan ditolak

Kemungkinan penyebabnya: File AWS CLI program tidak memiliki izin “jalankan”

Di Linux atau macOS, pastikan bahwa aws program telah menjalankan izin untuk pengguna yang menelepon. Biasanya, izin diatur ke755.

Untuk menambahkan izin jalankan bagi pengguna Anda, jalankan perintah berikut, ganti `~/local/bin/aws` dengan jalur ke program di komputer Anda.

```
$ chmod +x ~/local/bin/aws
```

[Kembali ke atas](#)

Kemungkinan penyebabnya: Identitas IAM Anda tidak memiliki izin untuk melakukan operasi

Contoh teks kesalahan:

```
$ aws s3 ls
An error occurred (AccessDenied) when calling the ListBuckets operation: Access
denied.
```

Saat Anda menjalankan AWS CLI perintah, AWS operasi dilakukan atas nama Anda, menggunakan kredensyal yang mengaitkan Anda dengan akun atau peran IAM. Kebijakan yang dilampirkan harus memberi Anda izin untuk memanggil tindakan API yang sesuai dengan perintah yang Anda jalankan dengan AWS CLI.

Sebagian besar perintah memanggil satu tindakan dengan nama yang cocok dengan nama perintah. Namun, perintah khusus seperti `aws s3 sync` memanggil beberapa API. Anda dapat melihat API mana yang dipanggil perintah dengan menggunakan `--debug` opsi.

Jika Anda yakin bahwa pengguna atau peran memiliki izin yang tepat yang ditetapkan oleh kebijakan, pastikan bahwa AWS CLI perintah Anda menggunakan kredensyal yang Anda harapkan. Lihat [bagian selanjutnya tentang kredensyal](#) untuk memverifikasi bahwa kredensyal yang digunakan AWS CLI adalah yang Anda harapkan.

Untuk informasi tentang menetapkan izin IAM, lihat [Ikhtisar Manajemen Akses: Izin dan Kebijakan dalam Panduan Pengguna IAM](#).

[Kembali ke atas](#)

## Kredensial dan kesalahan kunci tidak valid

Contoh teks kesalahan:

```
$ aws s3 ls
An error occurred (InvalidAccessKeyId) when calling the ListBuckets operation: The AWS
Access Key Id
you provided does not exist in our records.
```

```
$ aws s3 ls
An error occurred (InvalidClientTokenId) when calling the ListBuckets operation: The
security token
included in the request is invalid.
```

Kemungkinan penyebabnya: Membaca kredensial yang salah atau dari lokasi yang tidak terduga AWS CLI

AWS CLI Mungkin membaca kredensial dari lokasi yang berbeda dari yang Anda harapkan, atau informasi key pair Anda salah. Anda dapat menjalankan `aws configure list` untuk mengonfirmasi kredensial mana yang digunakan.

Contoh berikut menunjukkan cara memeriksa kredensial yang digunakan untuk profil default.

```
$ aws configure list
      Name                Value                Type    Location
      ----                -
      profile              <not set>           None    None
      access_key           *****XYVA         shared-credentials-file
      secret_key           *****ZAGY         shared-credentials-file
      region               us-west-2           config-file  ~/.aws/config
```

Contoh berikut menunjukkan cara memeriksa kredensial profil bernama.

```
$ aws configure list --profile saanvi
      Name                Value                Type    Location
      ----                -
      profile              saanvi              manual  --profile
      access_key           *****             shared-credentials-file
      secret_key           *****             shared-credentials-file
      region               us-west-2           config-file  ~/.aws/config
```

Untuk mengonfirmasi detail key pair Anda, periksa `credentials` file config dan file Anda. Untuk informasi lebih lanjut tentang config dan `credentials` file, lihat [the section called “Konfigurasi dan pengaturan file kredensi”](#). Untuk informasi selengkapnya tentang kredensial dan otentikasi, termasuk prioritas kredensial, lihat [Otentikasi dan akses kredensial](#)

### [Kembali ke atas](#)

Kemungkinan penyebabnya: Jam komputer Anda tidak sinkron

Jika Anda menggunakan kredensial yang valid, jam Anda mungkin tidak sinkron. Di Linux atau macOS, jalankan `date` untuk memeriksa waktu.

```
$ date
```

Jika jam sistem Anda tidak benar dalam beberapa menit, gunakan `ntpd` untuk menyinkronkannya.

```
$ sudo service ntpd stop
$ sudo ntpdate time.nist.gov
$ sudo service ntpd start
$ ntpstat
```

Di Windows, gunakan opsi tanggal dan waktu di Control Panel untuk mengonfigurasi jam sistem Anda.

### [Kembali ke atas](#)

## Tanda tangan tidak cocok dengan kesalahan

Contoh teks kesalahan:

```
$ aws s3 ls
An error occurred (SignatureDoesNotMatch) when calling the ListBuckets operation: The request signature we calculated does not match the signature you provided. Check your key and signing method.
```

Ketika AWS CLI menjalankan perintah, ia mengirimkan permintaan terenkripsi ke AWS server untuk melakukan operasi AWS layanan yang sesuai. Kredensial Anda (kunci akses dan kunci rahasia) terlibat dalam enkripsi dan memungkinkan AWS untuk mengautentikasi orang yang membuat

permintaan. Ada beberapa hal yang dapat mengganggu operasi yang benar dari proses ini, sebagai berikut.

Kemungkinan penyebabnya: Jam Anda tidak sinkron dengan AWS server

Untuk membantu melindungi dari [serangan replay](#), waktu saat ini dapat digunakan selama proses enkripsi/dekripsi. Jika waktu klien dan server tidak setuju dengan lebih dari jumlah yang diizinkan, proses dapat gagal dan permintaan ditolak. Ini juga dapat terjadi ketika Anda menjalankan perintah di mesin virtual yang jamnya tidak sinkron dengan jam mesin host. Salah satu kemungkinan penyebabnya adalah ketika mesin virtual hibernasi dan membutuhkan waktu setelah bangun untuk menyinkronkan jam dengan mesin host.

Di Linux atau macOS, jalankan `date` untuk memeriksa waktu.

```
$ date
```

Jika jam sistem Anda tidak benar dalam beberapa menit, gunakan `ntpd` untuk menyinkronkannya.

```
$ sudo service ntpd stop
$ sudo ntpdate time.nist.gov
$ sudo service ntpd start
$ ntpstat
```

Di Windows, gunakan opsi tanggal dan waktu di Control Panel untuk mengonfigurasi jam sistem Anda.

### [Kembali ke atas](#)

Kemungkinan penyebabnya: Sistem operasi Anda salah menangani AWS kunci yang berisi karakter khusus tertentu

Jika AWS kunci Anda menyertakan karakter khusus tertentu, seperti `-`, `+`, atau `/`, beberapa varian sistem operasi memproses string secara tidak benar dan menyebabkan string kunci ditafsirkan secara tidak benar.

Jika Anda memproses kunci Anda menggunakan alat atau skrip lain, seperti alat yang membangun file kredensial pada instance baru sebagai bagian dari pembuatannya, alat dan skrip tersebut mungkin memiliki penanganan karakter khusus sendiri yang menyebabkannya diubah menjadi sesuatu yang AWS tidak lagi dikenali.



Kami menyarankan untuk meregenerasi kunci rahasia untuk mendapatkan kunci yang tidak menyertakan karakter khusus yang menyebabkan masalah.

[Kembali ke atas](#)

## Kesalahan sertifikat SSL

Kemungkinan penyebabnya: AWS CLI Tidak mempercayai sertifikat proxy Anda

Contoh teks kesalahan:

```
$ aws s3 ls
[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed
```

Saat Anda menggunakan AWS CLI perintah, Anda menerima pesan [SSL: CERTIFICATE\_VERIFY\_FAILED] certificate verify failed kesalahan. Hal ini disebabkan oleh AWS CLI tidak mempercayai sertifikat proxy Anda karena faktor-faktor seperti sertifikat proxy Anda ditandatangani sendiri, dengan perusahaan Anda ditetapkan sebagai Otoritas Sertifikasi (CA). Ini mencegah AWS CLI dari menemukan sertifikat akar CA perusahaan Anda di registri CA lokal.

Untuk memperbaikinya, instruksikan AWS CLI di mana menemukan .pem file perusahaan Anda menggunakan pengaturan file [ca\\_bundle](#) konfigurasi, opsi baris [--ca-bundle](#) perintah, atau variabel [AWS\\_CA\\_BUNDLE](#) lingkungan.

[Kembali ke atas](#)

Kemungkinan penyebabnya: Konfigurasi Anda tidak menunjuk ke lokasi sertifikat root CA yang benar

Contoh teks kesalahan:

```
$ aws s3 ls
SSL validation failed for regionname [Errno 2] No such file or directory
```

Hal ini disebabkan oleh lokasi file bundel Otoritas Sertifikasi (CA) Anda dikonfigurasi secara tidak benar di file AWS CLI. Untuk memperbaikinya, konfirmasi di mana .pem file perusahaan Anda berada dan perbarui AWS CLI [ca\\_bundle](#) konfigurasi dengan menggunakan pengaturan file konfigurasi, opsi baris [--ca-bundle](#) perintah, atau variabel [AWS\\_CA\\_BUNDLE](#) lingkungan.

[Kembali ke atas](#)

Kemungkinan penyebabnya: Konfigurasi Anda tidak menggunakan yang benar Wilayah AWS

Contoh teks kesalahan:

```
$ aws s3 ls
[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed
```

Kesalahan atau hasil yang tidak terduga dapat terjadi jika Layanan AWS tidak tersedia untuk yang Anda tentukan Wilayah AWS atau sumber daya Anda berada di tempat yang berbeda Wilayah AWS. Untuk langkah-langkah pemecahan masalah, lihat [the section called “Periksa AWS CLI perintah yang Wilayah AWS Anda gunakan”](#).

[Kembali ke atas](#)

Kemungkinan penyebabnya: Versi TLS Anda perlu diperbarui

Contoh teks kesalahan:

```
$ aws s3 ls
[SSL: UNSAFE_LEGACY_RENEGOTIATION_DISABLED] unsafe legacy renegotiation disabled
```

Layanan AWS Ini menggunakan versi TLS yang tidak kompatibel dengan versi TLS perangkat Anda. Untuk mengatasi masalah ini, perbarui ke versi TLS yang didukung. Untuk informasi selengkapnya, lihat [the section called “Menegakkan versi TLS minimum”](#).

[Kembali ke atas](#)

## Kesalahan JSON tidak valid

Contoh teks kesalahan:

```
$ aws dynamodb update-table \
  --provisioned-throughput '{"ReadCapacityUnits':15,WriteCapacityUnits':10}' \
  --table-name MyDDBTable
Error parsing parameter '--provisioned-throughput': Invalid JSON: Expecting property
name enclosed in
double quotes: line 1 column 25 (char 24)
JSON received: {"ReadCapacityUnits":15,WriteCapacityUnits":10}
```

Saat Anda menggunakan AWS CLI perintah, Anda menerima pesan kesalahan Invalid JSON "". Ini biasanya merupakan kesalahan yang terlihat ketika Anda memasukkan perintah dengan format JSON yang diharapkan dan AWS CLI tidak dapat membaca JSON Anda dengan benar.

Kemungkinan penyebabnya: Anda tidak memasukkan JSON yang valid untuk AWS CLI digunakan

Konfirmasikan bahwa Anda telah memasukkan JSON yang valid untuk perintah Anda. Kami menyarankan menggunakan validator JSON untuk JSON Anda mengalami masalah dalam memformat.

Untuk penggunaan JSON yang lebih maju di baris perintah, pertimbangkan untuk menggunakan prosesor JSON baris perintah, seperti `jq`, untuk membuat string JSON. Untuk informasi lebih lanjut tentang `jq`, lihat [repositori jq](#) di GitHub

### [Kembali ke atas](#)

Kemungkinan penyebabnya: Aturan kutipan terminal Anda mencegah JSON yang valid dikirim ke AWS CLI

Sebelum AWS CLI menerima sesuatu dari perintah, terminal Anda memproses perintah menggunakan aturan kutipan dan pelolosan itu sendiri. Karena aturan pemformatan terminal, beberapa konten JSON Anda mungkin dilucuti sebelum perintah diteruskan ke file. AWS CLI Saat merumuskan perintah, pastikan untuk menggunakan aturan [kutipan terminal Anda](#).

Untuk memecahkan masalah, gunakan `echo` perintah untuk melihat bagaimana shell menangani parameter Anda:

```
$ echo {"ReadCapacityUnits":15,"WriteCapacityUnits":10}
ReadCapacityUnits:15 WriteCapacityUnits:10
```

```
$ echo '{"ReadCapacityUnits":15,"WriteCapacityUnits":10}'
{"ReadCapacityUnits":15,"WriteCapacityUnits":10}
```

Ubah perintah Anda hingga JSON Anda yang valid dikembalikan.

Untuk pemecahan masalah yang lebih mendalam, gunakan `--debug` parameter untuk melihat log debug karena akan menampilkan persis apa yang diteruskan ke: AWS CLI

```
$ aws dynamodb update-table \
  --provisioned-throughput '{"ReadCapacityUnits":15,WriteCapacityUnits":10}' \
```

```
--table-name MyDDBTable \  
--debug  
2022-07-19 22:25:07,741 - MainThread - awscli.clidriver - DEBUG - CLI version: aws-  
cli/1.18.147  
Python/2.7.18 Linux/5.4.196-119.356.amzn2int.x86_64 botocore/1.18.6  
2022-07-19 22:25:07,741 - MainThread - awscli.clidriver - DEBUG - Arguments entered  
to CLI:  
['dynamodb', 'update-table', '--provisioned-throughput',  
 '{"ReadCapacityUnits":15,WriteCapacityUnits":10}',  
 '--table-name', 'MyDDBTable', '--debug']
```

Gunakan aturan kutipan terminal Anda untuk memperbaiki masalah apa pun yang dimiliki input JSON Anda saat dikirim ke file. AWS CLI Untuk informasi lebih lanjut tentang aturan kutipan, lihat [the section called “Kutipan dengan Strings”](#).

#### Note

Jika Anda mengalami masalah dengan mendapatkan JSON yang valid ke AWS CLI, kami sarankan untuk melewati aturan kutipan terminal untuk input data JSON dengan menggunakan Blobs untuk meneruskan data JSON Anda langsung ke. AWS CLI Untuk informasi lebih lanjut tentang Blobs, lihat [blob](#).

[Kembali ke atas](#)

## Sumber daya tambahan

Untuk bantuan tambahan AWS CLI terkait masalah Anda, kunjungi [AWS CLI komunitas](#) di GitHub atau [AWS re:Post komunitas](#).

[Kembali ke atas](#)

# Migrasi dari AWS CLI versi 1 ke versi 2

Bagian ini berisi instruksi untuk memperbarui AWS CLI versi 1 ke AWS CLI versi 2. AWS CLIVersi 2 dibangun di atas AWS CLI versi 1 dan mencakup fitur dan penyempurnaan berdasarkan umpan balik komunitas.

Sebelum Anda bermigrasi ke versi 2, [pelajari tentang perbedaan antara versi](#). AWS CLIVersi 2 mencakup fitur dan perubahan baru yang mungkin mengharuskan Anda memperbarui skrip atau perintah untuk kompatibilitas mundur.

AWS CLIVersi 1 dan 2 menggunakan nama `aws` perintah yang sama. Jika Anda memiliki kedua versi yang diinstal, komputer Anda menggunakan yang pertama ditemukan di jalur pencarian Anda.

Jika sebelumnya Anda menginstal AWS CLI versi 1, ikuti [petunjuk migrasi kami untuk mulai menggunakan versi 2](#).

Jika Anda belum menginstal AWS CLI versi 1 sebelumnya, ikuti instruksi di [Memulai](#).

Topik

- [Fitur baru dan perubahan dalam AWS CLI versi 2](#)
- [AWS CLInstruksi migrasi versi 2](#)

## Fitur baru dan perubahan dalam AWS CLI versi 2

Topik ini menjelaskan fitur baru dan perubahan perilaku antara AWS CLI versi 1 dan AWS CLI versi 2. Perubahan ini mungkin mengharuskan Anda memperbarui skrip atau perintah untuk mendapatkan perilaku yang sama di versi 2 seperti yang Anda lakukan di versi 1.

Topik

- [AWS CLIVersi 2 fitur baru](#)
- [Memecahkan perubahan antara AWS CLI versi 1 dan AWS CLI versi 2](#)

## AWS CLIVersi 2 fitur baru

AWS CLIVersi 2 adalah versi utama terbaru AWS CLI dan mendukung semua fitur terbaru. Beberapa fitur yang diperkenalkan di versi 2 tidak di-backport ke versi 1 dan Anda harus meningkatkan untuk mengakses fitur-fitur tersebut. Fitur-fitur ini mencakup sebagai berikut:

## Penerjemah Python tidak diperlukan

AWS CLIVersi 2 tidak memerlukan instalasi Python yang terpisah. Ini termasuk versi tertanam.

### [Penyihir](#)

Anda dapat menggunakan wizard dengan AWS CLI versi 2. Wizard memandu Anda melalui pembuatan perintah tertentu.

### [Autentikasi Pusat Identitas IAM](#)

Jika organisasi Anda menggunakan AWS IAM Identity Center (Pusat Identitas IAM), pengguna Anda dapat masuk ke Active Directory, direktori Pusat Identitas IAM bawaan, atau [iDP lain yang terhubung ke Pusat Identitas](#) IAM. Kemudian, mereka dipetakan ke peran AWS Identity and Access Management (IAM) yang memungkinkan Anda menjalankan AWS CLI perintah.

### [Prompt otomatis](#)

Saat diaktifkan, AWS CLI versi 2 dapat meminta Anda untuk perintah, parameter, dan sumber daya saat Anda menjalankan aws perintah.

### [Jalankan AWS CLI dari gambar Amazon ECR Public atau Docker resmi](#)

Gambar Docker resmi untuk AWS CLI menyediakan isolasi, portabilitas, dan keamanan yang AWS secara langsung mendukung dan memelihara. Dengan cara ini, Anda dapat menggunakan AWS CLI versi 2 di lingkungan berbasis wadah tanpa harus mengelola instalasi sendiri.

### [Pager sisi klien](#)

AWS CLIVersi 2 menyediakan penggunaan program pager sisi klien untuk output. Secara default, fitur ini diaktifkan dan mengembalikan semua output melalui program pager default sistem operasi Anda.

### [aws configure import](#)

Impor .csv kredensial yang dihasilkan dari file. AWS Management Console .csvFile diimpor dengan nama profil yang cocok dengan nama pengguna IAM.

### [aws configure list-profiles](#)

Daftar nama semua profil yang telah Anda konfigurasi.

### [the section called “Format keluaran aliran YAMG”](#)

`yaml-streamFormat` `yaml` dan memanfaatkan format [YAML](#) sambil memberikan tampilan kumpulan data besar yang lebih responsif dengan mengalirkan data kepada Anda. Anda dapat mulai melihat dan menggunakan data YAMB sebelum seluruh unduhan kueri.

## [ddb](#)Perintah tingkat tinggi baru untuk DynamoDB

AWS CLIVersi 2 memiliki perintah Amazon [ddb putddb select](#)DynamoDB tingkat tinggi dan. Perintah ini menyediakan antarmuka yang disederhanakan untuk menempatkan item dalam tabel DynamoDB dan mencari di tabel atau indeks DynamoDB.

## [aws logs tail](#)

AWS CLIVersi 2 memiliki `aws logs tail` perintah khusus yang mengekor log untuk grup Amazon CloudWatch Logs. Secara default, perintah mengembalikan log dari semua aliran CloudWatch Log terkait selama sepuluh menit terakhir.

## [Menambahkan dukungan metadata untuk perintah tingkat tinggi s3](#)

AWS CLIVersi 2 menambahkan `--copy-props` parameter ke `s3` perintah tingkat tinggi. Dengan parameter ini, Anda dapat mengonfigurasi metadata dan tag tambahan untuk Amazon Simple Storage Service (Amazon S3).

## [AWS\\_REGION](#)

AWS CLIVersi 2 memiliki variabel lingkungan yang AWS kompatibel dengan SDK yang disebut `AWS_REGION`. Variabel ini menentukan Wilayah AWS untuk mengirim permintaan ke. Ini mengesampingkan variabel `AWS_DEFAULT_REGION` lingkungan, yang hanya berlaku di. AWS CLI

# Memecahkan perubahan antara AWS CLI versi 1 dan AWS CLI versi 2

Bagian ini menjelaskan semua perubahan perilaku antara AWS CLI versi 1 dan AWS CLI versi 2. Perubahan ini mungkin mengharuskan Anda memperbarui skrip atau perintah untuk mendapatkan perilaku yang sama di versi 2 seperti yang Anda lakukan di versi 1.

## Topik

- [Variabel lingkungan ditambahkan untuk mengatur pengkodean file teks](#)
- [Parameter biner dilewatkan sebagai string yang dikodekan base64 secara default](#)
- [Peningkatan penanganan Amazon S3 dari properti file dan tag untuk salinan multipart](#)
- [Tidak ada pengambilan otomatis http:// atau https:// URL untuk parameter](#)
- [Pager digunakan untuk semua output secara default](#)
- [Nilai keluaran stempel waktu distandarisasi ke format ISO 8601](#)
- [Peningkatan penanganan CloudFormation penerapan yang tidak menghasilkan perubahan](#)
- [Mengubah perilaku default untuk titik akhir Amazon S3 Regional untuk Wilayah us-east-1](#)

- [Mengubah perilaku default untuk titik AWS STS akhir Regional](#)
- [ecr get-logindihapus dan diganti dengan ecr get-login-password](#)
- [AWS CLI dukungan versi 2 untuk plugin berubah](#)
- [Dukungan alias tersembunyi dihapus](#)
- [Pengaturan berkas api\\_versions konfigurasi tidak didukung](#)
- [AWS CLI versi 2 hanya menggunakan Signature v4 untuk mengautentikasi permintaan Amazon S3](#)
- [AWS CLI versi 2 lebih konsisten dengan parameter paging](#)
- [AWS CLI versi 2 menyediakan kode pengembalian yang lebih konsisten di semua perintah](#)

## Variabel lingkungan ditambahkan untuk mengatur pengkodean file teks

Secara default, file teks untuk [the section called “blob”](#) menggunakan pengkodean yang sama dengan lokal yang diinstal. Karena AWS CLI versi 2 menggunakan versi Python yang disematkan, variabel PYTHONUTF8 dan PYTHONIOENCODING lingkungan tidak didukung. Untuk mengatur pengkodean file teks agar berbeda dari lokal, gunakan variabel AWS\_CLI\_FILE\_ENCODING lingkungan. Contoh berikut menetapkan AWS CLI untuk membuka file teks menggunakan UTF-8 pada Windows.

```
AWS_CLI_FILE_ENCODING=UTF-8
```

Untuk informasi selengkapnya, lihat [Variabel lingkungan untuk mengkonfigurasi AWS CLI](#).

## Parameter biner dilewatkan sebagai string yang dikodekan base64 secara default

Dalam AWS CLI, beberapa perintah memerlukan string [base64](#) -encoded, sementara yang lain memerlukan string byte yang dikodekan UTF-8. Dalam AWS CLI versi 1, meneruskan data antara dua jenis string yang dikodekan sering memerlukan beberapa pemrosesan perantara. AWS CLI Versi 2 membuat penanganan parameter biner lebih konsisten, yang membantu meneruskan nilai dari satu perintah ke perintah lainnya dengan lebih andal.

Secara default, AWS CLI versi 2 melewati semua parameter input biner dan output biner sebagai string yang dikodekan base64 blobs (objek besar biner). Untuk informasi selengkapnya, lihat [the section called “blob”](#).

Untuk kembali ke perilaku AWS CLI versi 1, gunakan konfigurasi [cli\\_binary\\_format](#) file atau `--cli-binary-format` parameter.



## Peningkatan penanganan Amazon S3 dari properti file dan tag untuk salinan multipart

Saat Anda menggunakan perintah AWS CLI versi 1 di `aws s3` namespace untuk menyalin file dari satu lokasi bucket S3 ke lokasi bucket lainnya, dan operasi tersebut menggunakan [salinan multipart](#), tidak ada properti file dari objek sumber yang disalin ke objek tujuan.

Secara default, perintah yang sesuai di AWS CLI versi 2 mentransfer semua tag dan beberapa properti dari sumber ke salinan tujuan. Dibandingkan dengan AWS CLI versi 1, ini dapat menghasilkan lebih banyak panggilan AWS API yang dilakukan ke titik akhir Amazon S3. Untuk mengubah perilaku default untuk `s3` perintah di AWS CLI versi 2, gunakan `--copy-props` parameter.

Untuk informasi selengkapnya, lihat [the section called “Properti file dan tag dalam salinan multipart”](#).

## Tidak ada pengambilan otomatis `http://` atau `https://` URL untuk parameter

AWS CLIVersi 2 tidak melakukan GET operasi ketika nilai parameter dimulai dengan `http://` atau `https://`, dan tidak menggunakan konten yang dikembalikan sebagai nilai parameter. Akibatnya, opsi baris perintah terkait `cli_follow_url` param dihapus dari AWS CLI versi 2.

Jika Anda perlu mengambil URL dan meneruskan konten URL ke nilai parameter, kami sarankan Anda menggunakan `curl` atau alat serupa untuk mengunduh konten URL ke file lokal. Kemudian, gunakan `file://` sintaks untuk membaca isi file itu dan menggunakannya sebagai nilai parameter.

Misalnya, perintah berikut tidak lagi mencoba untuk mengambil isi halaman yang ditemukan di `http://www.example.com` dan meneruskan konten tersebut sebagai parameter. Sebaliknya, ia melewati string teks literal `https://example.com` sebagai parameter.

```
$ aws ssm put-parameter \  
  --value http://www.example.com \  
  --name prod.microservice1.db.secret \  
  --type String 2
```

Jika Anda perlu mengambil dan menggunakan konten URL web sebagai parameter, Anda dapat melakukan hal berikut di versi 2.

```
$ curl https://my.example.com/mypolicyfile.json -o mypolicyfile.json  
$ aws iam put-role-policy \  
  --policy-document file://./mypolicyfile.json \  
  --role-name MyRole \  
  --
```

```
--policy-name MyReadOnlyPolicy
```

Pada contoh sebelumnya, `-o` parameter memberitahu `curl` untuk menyimpan file di folder saat ini dengan nama yang sama dengan file sumber. Perintah kedua mengambil konten dari file yang diunduh dan meneruskan konten sebagai nilai. `--policy-document`

## Pager digunakan untuk semua output secara default

Secara default, AWS CLI versi 2 mengembalikan semua output melalui program pager default sistem operasi Anda. Program ini adalah [less](#) program di Linux atau macOS, dan [more](#) program di Windows. Ini dapat membantu Anda menavigasi sejumlah besar output dari layanan dengan menampilkan output itu satu halaman pada satu waktu.

Anda dapat mengonfigurasi AWS CLI versi 2 untuk menggunakan program paging yang berbeda atau tidak sama sekali. Untuk informasi selengkapnya, lihat [the section called “Pager sisi klien”](#).

## Nilai keluaran stempel waktu distandarisasi ke format ISO 8601

Secara default, AWS CLI versi 2 mengembalikan semua nilai respons stempel waktu dalam format [ISO 8601](#). Di AWS CLI versi 1, perintah mengembalikan nilai stempel waktu dalam format apa pun yang dikembalikan oleh respons HTTP API, yang dapat bervariasi dari layanan ke layanan.

Untuk melihat stempel waktu dalam format yang ditampilkan oleh respons HTTP API, gunakan `wire` nilai dalam file `Andaconfig`. Untuk informasi selengkapnya, lihat [cli\\_timestamp\\_format](#).

## Peningkatan penanganan CloudFormation penerapan yang tidak menghasilkan perubahan

Secara default di AWS CLI versi 1, jika Anda menerapkan AWS CloudFormation template yang tidak menghasilkan perubahan, kode kesalahan akan AWS CLI dikembalikan. Ini menyebabkan masalah jika Anda tidak menganggap itu sebagai kesalahan dan Anda ingin skrip Anda berlanjut. Anda dapat mengatasi ini di AWS CLI versi 1 dengan menambahkan bendera `--no-fail-on-empty-changeset`, yang kembali `0`.

Karena ini adalah kasus penggunaan umum, AWS CLI versi 2 default mengembalikan kode keluar yang berhasil `0` ketika tidak ada perubahan yang disebabkan oleh penerapan dan operasi mengembalikan set perubahan kosong.

Untuk kembali ke perilaku asli, tambahkan bendera `--fail-on-empty-changeset`.

## Mengubah perilaku default untuk titik akhir Amazon S3 Regional untuk Wilayah **us-east-1**

Saat Anda mengonfigurasi AWS CLI versi 1 untuk menggunakan `us-east-1` Wilayah, maka akan AWS CLI menggunakan `s3.amazonaws.com` titik akhir global yang dihosting secara fisik di `us-east-1` Wilayah. AWS CLI Versi 2 menggunakan titik akhir Regional yang sebenarnya `s3.us-east-1.amazonaws.com` ketika Region tersebut ditentukan. Untuk memaksa AWS CLI versi 2 menggunakan titik akhir global, Anda dapat mengatur Region untuk `aws-global` perintah.

## Mengubah perilaku default untuk titik AWS STS akhir Regional

Secara default, AWS CLI versi 2 mengirimkan semua AWS Security Token Service (AWS STS) permintaan API ke titik akhir Regional untuk yang saat ini dikonfigurasi Wilayah AWS.

Secara default, AWS CLI versi 1 mengirimkan AWS STS permintaan ke AWS STS titik akhir global. Anda dapat mengontrol perilaku default ini di versi 1 dengan menggunakan [sts\\_regional\\_endpoints](#) pengaturan.

## **aws ecr get-login** dihapus dan diganti dengan **aws ecr get-login-password**

AWS CLI Versi 2 menggantikan perintah `aws ecr get-login` dengan `aws ecr get-login-password` perintah yang meningkatkan integrasi otomatis dengan otentikasi kontainer.

`aws ecr get-login-password` Perintah mengurangi risiko mengekspos kredensial Anda dalam daftar proses, riwayat shell, atau file log lainnya. Ini juga meningkatkan kompatibilitas dengan `docker login` perintah untuk otomatisasi yang lebih baik.

`aws ecr get-login-password` Perintah ini tersedia dalam AWS CLI versi 1.17.10 dan yang lebih baru, dan AWS CLI versi 2. `aws ecr get-login` Perintah sebelumnya masih tersedia di AWS CLI versi 1 untuk kompatibilitas mundur.

Dengan `aws ecr get-login-password` perintah tersebut, Anda dapat mengganti kode berikut yang mengambil kata sandi.

```
$ (aws ecr get-login --no-include-email)
```

Untuk mengurangi risiko mengekspos kata sandi ke riwayat shell atau log, gunakan perintah contoh berikut sebagai gantinya. Dalam contoh ini, kata sandi disalurkan langsung ke `docker login` perintah, di mana ia ditetapkan ke parameter kata sandi dengan `--password-stdin` opsi.

```
$ aws ecr get-login-password | docker login --username AWS --password-stdin MY-REGISTRY-URL
```

Untuk informasi selengkapnya, lihat [aws ecr get-login-password](#) di Panduan Referensi AWS CLI versi 2.

## AWS CLI dukungan versi 2 untuk plugin berubah

Dukungan plugin di AWS CLI versi 2 sepenuhnya sementara dan dimaksudkan untuk membantu pengguna bermigrasi dari AWS CLI versi 1 hingga antarmuka plugin yang stabil dan diperbarui dirilis. Tidak ada jaminan bahwa plugin tertentu atau bahkan antarmuka AWS CLI plugin akan didukung di versi masa depan AWS CLI versi 2. Jika Anda mengandalkan plugin, pastikan untuk mengunci ke versi tertentu dari AWS CLI dan menguji fungsionalitas plugin Anda ketika Anda melakukan upgrade.

Untuk mengaktifkan dukungan plugin, buat [plugins] bagian di bagian Anda `~/.aws/config`.

```
[plugins]
cli_legacy_plugin_path = <path-to-plugins>/python3.7/site-packages
<plugin-name> = <plugin-module>
```

Di [plugins] bagian ini, tentukan `cli_legacy_plugin_path` variabel dan atur nilainya ke jalur paket situs Python tempat modul plugin Anda berada. Kemudian, Anda dapat mengkonfigurasi plugin dengan memberikan nama untuk plugin (`plugin-name`) dan nama file modul Python (`plugin-module`) yang berisi kode sumber untuk plugin Anda. AWS CLI Memuat setiap plugin dengan mengimpor `plugin-module` dan memanggil `awscli_initialize` fungsinya.

## Dukungan alias tersembunyi dihapus

AWS CLI versi 2 tidak lagi mendukung alias tersembunyi berikut yang didukung di versi 1.

Pada tabel berikut, kolom pertama menampilkan layanan, perintah, dan parameter yang berfungsi di semua versi, termasuk AWS CLI versi 2. Kolom kedua menampilkan alias yang tidak lagi berfungsi di AWS CLI versi 2.

Layanan kerja, perintah, dan parameter	Alias usang
kognito-identitas -arns create-identity-pool open-id-connect-provider	open-id-connect-provider-arns

Layanan kerja, perintah, dan parameter	Alias usang
storagegateway mendeskripsikan kaset-kaset-arns	tape-ar-ns
Storagegateway. describe-tape-archives.tape-arns	tape-ar-ns
Storagegateway. describe-vtl-devices. vtl-device-arns	vtl-device-ar-ns
Storagegateway. describe-cached-iscsi-volumes.volume-arns	volume-ar-ns
Storagegateway. describe-stored-iscsi-volumes.volume-arns	volume-ar-ns
route53domains.view-billing.start-time	start
menyebarkan. create-deployment-group.ec2-tag-set	ec-2-tag-set
menyebarkan. list-application-revisions.s3-ember	s-3-ember
menyebarkan. list-application-revisions.s3-kunci-awalan	awalan s-3-kunci-
menyebarkan. update-deployment-group.ec2-tag-set	ec-2-tag-set
saya. enable-mfa-device.otentikasi-kode1	otentikasi-kode-1
saya. enable-mfa-device.otentikasi-kode2	otentikasi-kode-2
saya. resync-mfa-device.otentikasi-kode1	otentikasi-kode-1
saya. resync-mfa-device.otentikasi-kode2	otentikasi-kode-2
impor ekspor. get-shipping-label.jalan1	jalan-1
impor ekspor. get-shipping-label.jalan2	jalan-2
impor ekspor. get-shipping-label.jalan3	jalan-3
lambda.publish-version.code-sha256	kode-sha-256
lightsail. import-key-pair. public-key-base64	public-key-base-64
opsworks.register-volume.ec2-volume-id	ec-2-volume id

## Pengaturan berkas `api_versions` konfigurasi tidak didukung

AWS CLI versi 2 tidak mendukung pemanggilan versi AWS layanan API sebelumnya dengan menggunakan setelan file `api_versions` konfigurasi. Semua AWS CLI perintah sekarang memanggil versi terbaru dari API layanan yang saat ini didukung oleh endpoint.

## AWS CLI versi 2 hanya menggunakan Signature v4 untuk mengautentikasi permintaan Amazon S3

AWS CLI versi 2 tidak mendukung algoritme tanda tangan sebelumnya untuk mengautentikasi permintaan layanan secara kriptografis yang dikirim ke titik akhir Amazon S3. Penandatanganan ini terjadi secara otomatis dengan setiap permintaan Amazon S3 dan hanya [Proses Penandatanganan Versi Tanda Tangan 4](#) yang didukung. Anda tidak dapat mengonfigurasi versi tanda tangan. Semua URL presigned bucket Amazon S3 sekarang hanya menggunakan SigV4 dan memiliki durasi kedaluwarsa maksimum satu minggu.

## AWS CLI versi 2 lebih konsisten dengan parameter paging

Dalam AWS CLI versi 1, jika Anda menentukan parameter pagination pada baris perintah, maka pagination otomatis dimatikan seperti yang diharapkan. Namun, ketika Anda menentukan parameter pagination dengan menggunakan file dengan `--cli-input-json` parameter, pagination otomatis tidak dimatikan, yang dapat menghasilkan output yang tidak terduga. AWS CLI versi 2 mematikan pagination otomatis terlepas dari bagaimana Anda memberikan parameter.

## AWS CLI versi 2 menyediakan kode pengembalian yang lebih konsisten di semua perintah

AWS CLI versi 2 lebih konsisten di semua perintah dan mengembalikan kode keluar yang sesuai dengan AWS CLI versi 1 dengan benar. Kami juga menambahkan kode keluar 252, 253, dan 254. Untuk informasi selengkapnya tentang kode keluar, lihat [the section called "Kode Pengembalian"](#).

Jika Anda memiliki ketergantungan pada bagaimana AWS CLI versi 1 menggunakan nilai kode pengembalian, kami sarankan untuk memeriksa kode keluar untuk memastikan bahwa Anda mendapatkan nilai yang Anda harapkan.

## AWS CLI instruksi migrasi versi 2

Topik ini memberikan petunjuk untuk bermigrasi dari AWS CLI versi 1 ke AWS CLI versi 2.

AWS CLI versi 1 dan 2 menggunakan nama `aws` perintah yang sama. Jika Anda memiliki kedua versi yang diinstal, komputer Anda menggunakan yang pertama ditemukan di jalur pencarian Anda. Jika sebelumnya Anda menginstal AWS CLI versi 1, kami sarankan Anda melakukan salah satu hal berikut untuk menggunakan AWS CLI versi 2:

- Direkomendasikan - [Copot pemasangan AWS CLI versi 1 dan gunakan hanya AWS CLI versi 2](#).
- [Untuk menginstal kedua versi](#), gunakan kemampuan sistem operasi Anda untuk membuat tautan simbolik (symlink) atau alias dengan nama yang berbeda untuk salah satu dari dua perintah. `aws`

Untuk informasi tentang melanggar perubahan antara versi 1 dan versi 2, lihat [the section called "Fitur dan perubahan baru"](#).

## Mengganti versi 1 dengan versi 2

Lakukan langkah-langkah berikut untuk mengganti AWS CLI versi 1 dengan AWS CLI versi 2.

Untuk mengganti AWS CLI versi 1 dengan AWS CLI versi 2

1. Siapkan skrip yang ada yang Anda miliki untuk migrasi dengan mengonfirmasi setiap perubahan yang melanggar antara versi 1 dan versi 2 di [the section called "Fitur dan perubahan baru"](#).
2. Copot pemasangan AWS CLI versi 1 dengan mengikuti petunjuk `uninstall` untuk sistem operasi Anda di [Menginstal, memperbarui, dan menghapus instalasi](#) versi 1. AWS CLI
3. Konfirmasikan bahwa AWS CLI sudah benar-benar dihapus dengan menggunakan perintah berikut.

```
$ aws --version
```

Lengkapi salah satu dari berikut ini berdasarkan output:

- Tidak ada versi yang dikembalikan: Anda telah berhasil menghapus AWS CLI versi 1 dan dapat melanjutkan ke langkah berikutnya.
  - Versi dikembalikan: Anda masih memiliki instalasi AWS CLI versi 1. Untuk langkah-langkah pemecahan masalah, lihat [the section called "Perintah `aws --version`" mengembalikan versi setelah mencopot pemasangan AWS CLI"](#). Lakukan langkah pemecahan masalah hingga tidak ada keluaran versi yang diterima.
4. Instal AWS CLI versi 2 dengan mengikuti petunjuk penginstalan yang sesuai untuk sistem operasi Anda di [Instal atau perbarui ke versi terbaru AWS CLI](#).

## Side-by-side instal

Untuk menginstal kedua versi, gunakan kemampuan sistem operasi Anda untuk membuat tautan simbolik (symlink) atau alias dengan nama yang berbeda untuk salah satu dari dua perintah. `aws`

1. Instal AWS CLI versi 2 dengan mengikuti petunjuk penginstalan yang sesuai untuk sistem operasi Anda di [instal atau perbarui ke versi terbaru AWS CLI](#).
2. Gunakan kemampuan sistem operasi Anda untuk membuat symlink atau alias dengan nama yang berbeda untuk salah satu dari dua `aws` perintah, seperti menggunakan `aws2` untuk AWS CLI versi 2. Berikut ini adalah contoh symlink untuk AWS CLI versi 2. Ganti `PATH` dengan lokasi penginstalan Anda.

### Linux and macOS

Anda dapat menggunakan [tautan simbolis](#) atau [alias](#) di Linux dan macOS.

```
$ alias aws2='PATH'
```

### Windows command prompt

[DOSKEY](#) di Windows.

```
C:\> doskey aws2=PATH
```



# Copot pemasangan AWS CLI versi 2

Topik ini menjelaskan cara menghapus instalasi AWS Command Line Interface versi 2 (AWS CLI versi 2).

AWS CLI instruksi penghapusan instalasi versi 2:

## Linux

Untuk menghapus instalasi AWS CLI versi 2, jalankan perintah berikut.

1. Temukan symlink dan instal jalur.

- Gunakan `which` perintah untuk menemukan symlink. Ini menunjukkan jalur yang Anda gunakan dengan `--bin-dir` parameter.

```
$ which aws
/usr/local/bin/aws
```

- Gunakan `ls` perintah untuk menemukan direktori yang ditunjuk symlink. Ini memberi Anda jalur yang Anda gunakan dengan `--install-dir` parameter.

```
$ ls -l /usr/local/bin/aws
lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/local/
aws-cli/v2/current/bin/aws
```

2. Hapus dua symlink di `--bin-dir` direktori. Jika pengguna Anda memiliki izin menulis ke direktori ini, Anda tidak perlu menggunakannyaudo.

```
$ sudo rm /usr/local/bin/aws
$ sudo rm /usr/local/bin/aws_completer
```

3. Hapus `--install-dir` direktori. Jika pengguna Anda memiliki izin menulis ke direktori ini, Anda tidak perlu menggunakannyaudo.

```
$ sudo rm -rf /usr/local/aws-cli
```

4. (Opsional) Hapus AWS SDK bersama dan informasi AWS CLI pengaturan di `.aws` folder.

**⚠ Warning**

Pengaturan konfigurasi dan kredensial ini dibagikan di semua AWS SDK dan AWS CLI. Jika Anda menghapus folder ini, mereka tidak dapat diakses oleh AWS SDK apa pun yang masih ada di sistem Anda.

Lokasi default `.aws` folder berbeda antar platform, secara default folder terletak di `~/.aws/`. Jika pengguna Anda memiliki izin menulis ke direktori ini, Anda tidak perlu menggunakannya sudo.

```
$ sudo rm -rf ~/.aws/
```

## macOS

Untuk menghapus instalasi AWS CLI versi 2, jalankan perintah berikut, ganti jalur yang Anda gunakan untuk menginstal. Perintah contoh menggunakan jalur instalasi default.

1. Temukan folder yang berisi symlink ke program utama dan pelengkap.

```
$ which aws
/usr/local/bin/aws
```

2. Dengan menggunakan informasi itu, jalankan perintah berikut untuk menemukan folder instalasi yang ditunjuk symlink.

```
$ ls -l /usr/local/bin/aws
lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/local/
aws-cli/aws
```

3. Hapus dua symlink di folder pertama. Jika pengguna Anda sudah memiliki izin menulis ke folder ini, Anda tidak perlu menggunakannya sudo.

```
$ sudo rm /usr/local/bin/aws
$ sudo rm /usr/local/bin/aws_completer
```

4. Hapus folder instalasi utama. Gunakan sudo untuk mendapatkan akses tulis ke `/usr/local` folder.

```
$ sudo rm -rf /usr/local/aws-cli
```

5. (Opsional) Hapus AWS SDK bersama dan informasi AWS CLI pengaturan di `.aws` folder.

#### Warning

Pengaturan konfigurasi dan kredensial ini dibagikan di semua AWS SDK dan AWS CLI. Jika Anda menghapus folder ini, mereka tidak dapat diakses oleh AWS SDK apa pun yang masih ada di sistem Anda.

Lokasi default `.aws` folder berbeda antar platform, secara default folder terletak di `~/.aws/`. Jika pengguna Anda memiliki izin menulis ke direktori ini, Anda tidak perlu menggunakannya sudo.

```
$ sudo rm -rf ~/.aws/
```

## Windows

1. Buka Program dan Fitur dengan melakukan salah satu hal berikut:
  - Buka Control Panel, lalu pilih Programs and Features.
  - Buka prompt perintah, lalu masukkan perintah berikut.

```
C:\> appwiz.cpl
```

2. Pilih entri bernama AWS Command Line Interface, lalu pilih Uninstall untuk meluncurkan uninstaller.
3. Konfirmasikan bahwa Anda ingin menghapus instalasi AWS CLI.
4. (Opsional) Hapus AWS SDK bersama dan informasi AWS CLI pengaturan di `.aws` folder.

#### Warning

Pengaturan konfigurasi dan kredensial ini dibagikan di semua AWS SDK dan AWS CLI. Jika Anda menghapus folder ini, mereka tidak dapat diakses oleh AWS SDK apa pun yang masih ada di sistem Anda.

Lokasi default `.aws` folder berbeda antar platform, secara default folder tersebut terletak di `%UserProfile%\ .aws`.

```
$ rmdir %UserProfile%\ .aws
```

## Memecahkan masalah kesalahan AWS CLI pemasangan dan uninstall

Jika Anda menemukan masalah setelah menginstal atau menghapus instalasi AWS CLI, lihat [Memecahkan masalah kesalahan](#) untuk langkah-langkah pemecahan masalah. Untuk langkah pemecahan masalah yang paling relevan, lihat, [the section called “Perintah tidak ditemukan kesalahan”](#) [the section called “Perintah aws --version ”” mengembalikan versi yang berbeda dari yang Anda instal”](#), dan [the section called “Perintah aws --version ”” mengembalikan versi setelah mencopot pemasangan AWS CLI”](#)

# AWS CLI riwayat dokumen panduan pengguna

Tabel berikut menjelaskan penambahan penting pada Panduan AWS Command Line Interface Pengguna, dimulai pada Januari 2019. Untuk notifikasi tentang pembaruan dokumentasi ini, Anda dapat berlangganan ke umpan RSS.

Perubahan	Deskripsi	Tanggal
<a href="#">Informasi kredensi dan otentikasi yang diperbarui.</a>	Instruksi dan contoh metode kredensi dan otentikasi yang diperbarui. Ini termasuk memperbarui halaman Memulai dan halaman konfigurasi yang relevan. Untuk mengakomodasi peningkatan dokumentasi ini, topik kredensi yang relevan dipindahkan ke bagian Otentikasi dan akses kredensi baru <a href="#">Otentikasi dan akses kredensi</a>	Maret 31, 2023
<a href="#">Konfigurasi penyedia token dengan penyegaran otentikasi otomatis untuk ditambahkan AWS IAM Identity Center</a>	Proses baru untuk mengonfigurasi untuk mengotentikasi pengguna dengan AWS IAM Identity Center (IAM Identity Center) menggunakan konfigurasi penyedia token SSO, yang dapat secara otomatis mengambil token otentikasi yang diperbarui. AWS CLI	Desember 7, 2022
<a href="#">Gambar Publik Amazon ECR resmi untuk AWS CLI versi 2 dirilis</a>	Gambar Publik Amazon ECR yang didukung resmi untuk	November 18, 2022

---

	AWS CLI versi 2 dirilis untuk Linux, macOS, dan Windows.	
<a href="#">Memperbarui panduan untuk bermigrasi dari AWS CLI V1 ke V2</a>	Memperluas panduan perubahan yang melanggar untuk menyertakan instruksi migrasi untuk beralih dari AWS CLI versi 1 ke AWS CLI versi 2. Termasuk pembaruan ke halaman Pemecahan Masalah untuk membantu masalah penginstalan.	Mei 13, 2022
<a href="#">Proses baru untuk membangun AWS CLI installer dari sumber.</a>	Proses baru untuk menginstal atau memperbarui dari sumber ke rilis terbaru dari AWS CLI sistem operasi yang didukung.	Februari 17, 2022
<a href="#">Konten untuk AWS CLI V1 dan V2 sekarang dipisahkan ke dalam panduan masing-masing</a>	Untuk kejelasan dan kemudahan, konten AWS CLI versi 1 dan AWS CLI versi 2 sekarang dipisahkan menjadi panduan mereka sendiri. Untuk AWS CLI versi 1, lihat <a href="#">Panduan Pengguna AWS CLI versi 1</a> .	November 2, 2021
<a href="#">Menambahkan AWS CLI informasi alias</a>	Menambahkan informasi AWS CLI alias. Alias adalah pintasan yang dapat Anda buat di AWS Command Line Interface (AWS CLI) untuk mempersingkat perintah atau skrip yang sering Anda gunakan.	11 Maret 2021

---

<a href="#">Informasi keluaran filter yang diperbarui</a>	Memperbarui informasi untuk filter dan dipindahkan ke halaman mereka sendiri.	1 Februari 2021
<a href="#">Menambahkan informasi untuk Wizards</a>	Ditambahkan AWS CLI versi 2 informasi wizard.	20 November 2020
<a href="#">Prompt otomatis yang diperbarui</a>	Memperbarui informasi prompt otomatis AWS CLI versi 2 dengan fitur saat ini.	10 November 2020
<a href="#">Ditambahkan Amazon S3 contoh scripting</a>	Menambahkan contoh skrip siklus hidup Amazon S3.	15 Oktober 2020
<a href="#">Ditambahkan Amazon EC2 contoh scripting</a>	Menambahkan contoh skrip jenis instans Amazon EC2.	15 Oktober 2020
<a href="#">Menambahkan informasi percobaan ulang</a>	Menambahkan halaman percobaan ulang untuk fitur dan perilaku percobaan ulang di. AWS CLI	17 September 2020
<a href="#">Halaman pagination sisi server dan sisi klien</a>	Informasi pagination diperbarui dan terpusat pada satu halaman.	17 Agustus 2020
<a href="#">Halaman perintah s3 yang diperbarui</a>	Memperbarui halaman perintah s3 tingkat tinggi dengan contoh dan sumber daya baru.	30 Juli 2020
<a href="#">Informasi instalasi yang diperbarui</a>	Informasi instal, perbarui, dan hapus instalasi untuk Linux, macOS, dan Windows diperbarui.	19 Mei 2020

<a href="#">Menambahkan informasi untuk pengkodean file teks pada AWS CLI versi 2</a>	Secara default, AWS CLI versi 2 menggunakan pengkodean file teks yang sama dengan lokal. Anda sekarang dapat menggunakan variabel lingkungan untuk mengatur pengkodean file teks.	14 Mei 2020
<a href="#">Gambar Docker resmi untuk AWS CLI versi 2 dirilis</a>	Dukungan resmi gambar Docker untuk AWS CLI versi 2 dirilis untuk semua Linux, macOS, dan Windows.	31 Maret 2020
<a href="#">Menambahkan informasi mengenai pager sisi klien untuk versi 2 AWS CLI</a>	Secara default, AWS CLI versi 2 menggunakan program pager <code>less</code> untuk semua output sisi klien.	19 Februari 2020
<a href="#">AWS Command Line Interface (AWS CLI) Versi 2 secara resmi dirilis</a>	AWS CLIVersi 2 umumnya tersedia dan merupakan versi yang direkomendasikan bagi pelanggan untuk menginstal.	10 Februari 2020
<a href="#">Penginstal macOS untuk AWS CLI versi 2 sekarang menjadi file penginstal Paket Apple. .pkg</a>	Penginstal macOS untuk AWS CLI versi 2 telah diperbarui dari .zip file dengan skrip shell ke paket Penginstal macOS lengkap. Ini menyederhanakan instalasi dan membuatnya kompatibel dengan rilis macOS terbaru.	Februari 3, 2020



<a href="#">Menambahkan konten untuk penanganan default S3 dan STS Regional endpoint AWS CLI versi 2 yang ditingkatkan</a>	Secara default, AWS CLI versi 2 sekarang mengarahkan permintaan untuk Amazon S3 AWS STS dan layanan ke titik akhir Regional yang saat ini dikonfigurasi, bukan titik akhir global.	13 Januari 2020
<a href="#">Rilis pratinjau pengembang untuk AWS CLI versi 2</a>	Mengumumkan rilis pratinjau AWS CLI versi 2. Menambahkan instruksi tentang menginstall versi 2. Tambahkan topik Migrasi untuk membahas perbedaan antara versi 1 dan 2.	7 November 2019
<a href="#">Ditambahkan dukungan AWS IAM Identity Center untuk profil AWS CLI bernama</a>	AWS CLI versi 2 menambahkan dukungan untuk membuat profil bernama yang dapat langsung masuk ke IAM Identity Center dan mendapatkan kredensi AWS sementara untuk digunakan dalam perintah berikutnya. AWS CLI	7 November 2019
<a href="#">Bagian MFA baru</a>	Menambahkan bagian baru yang menjelaskan cara mengakses CLI menggunakan otentikasi dan peran multi-faktor.	3 Mei 2019
<a href="#">Perbarui ke bagian “Menggunakan CLI”</a>	Perbaiki dan penambahan besar pada instruksi dan prosedur penggunaan.	7 Maret 2019

[Perbarui ke bagian “Menginstal CLI”](#)

Perbaikan dan penambahan besar pada instruksi dan prosedur AWS CLI instalasi.

7 Maret 2019

[Perbarui ke bagian “Mengkonfigurasi CLI”](#)

Perbaikan dan penambahan besar pada instruksi dan prosedur AWS CLI konfigurasi.

7 Maret 2019

# AWS Glosarium

Untuk AWS terminologi terbaru, lihat [AWS glosarium di Referensi](#).Glosarium AWS